QucsStudio

Technical Documentation

Michael Margraf
Stefan Jahn
Vincent Habchi

Contents

Chapter 1

DC Analysis

1.1 Modified Nodal Analysis

Many different kinds of network element are encountered in network analysis. For circuit analysis it is necessary to formulate equations for circuits containing as many different types of network elements as possible. There are various methods for equation formulation for a circuit. These are based on three types of equations found in circuit theory:

- equations based on Kirchhoff's voltage law (KVL)
- equations based on Kirchhoff's current law (KCL)
- branch constitutive equations

The equations have to be formulated (represented in a computer program) automatically in a simple, comprehensive manner. Once formulated, the system of equations has to be solved. There are two main aspects to be considered when choosing algorithms for this purpose: accuracy and speed. The MNA, briefly for **M**odified **N**odal **A**nalysis, has been proved to accomplish these tasks. MNA applied to a circuit with passive elements, independent current and voltage sources and active elements results in a matrix equation of the form:

$$[A] \cdot [x] = [z] \tag{1.1}$$

For a circuit with N nodes and M independent voltage sources:

- The A matrix
  - is (N+M)×(N+M) in size, and consists only of known quantities
  - the N×N part of the matrix in the upper left:
    - has only passive elements
    - elements connected to ground appear only on the diagonal
    - elements not connected to ground are both on the diagonal and off-diagonal terms
  - the rest of the A matrix (not included in the N×N upper left part) contains only 1, -1 and 0 (other values are possible if there are dependent current and voltage sources)
- The x matrix
  - is an (N+M)×1 vector that holds the unknown quantities (node voltages and the currents through the independent voltage sources)
  - the top N elements are the n node voltages

  - the bottom M elements represent the currents through the M independent voltage sources in the circuit

- The z matrix
  - is an (N+M)×1 vector that holds only known quantities
  - the top N elements are either zero or the sum and difference of independent current sources in the circuit
  - the bottom M elements represent the M independent voltage sources in the circuit

The circuit is solved by a simple matrix manipulation:

$$[x] = [A]^{-1} \cdot [z] \tag{1.2}$$

Though this may be difficult by hand, it is straightforward and so is easily done by computer.

1.1.1 Generating the MNA matrices

The following section is an algorithmic approach to the concept of the Modified Nodal Analysis. There are three matrices we need to generate, the A matrix, the x matrix and the z matrix. Each of these will be created by combining several individual sub-matrices.

1.1.2 The A matrix

The A matrix will be developed as the combination of 4 smaller matrices, G, B, C, and D.

$$A = \begin{bmatrix} G & B \\ C & D \end{bmatrix} \tag{1.3}$$

The A matrix is (M+N)×(M+N) (N is the number of nodes, and M is the number of independent voltage sources) and:

- the G matrix is N×N and is determined by the interconnections between the circuit elements
- the B matrix is N×M and is determined by the connection of the voltage sources
- the C matrix is M×N and is determined by the connection of the voltage sources (B and C are closely related, particularly when only independent sources are considered)
- the D matrix is M×M and is zero if only independent sources are considered

Rules for making the G matrix

The G matrix is an N×N matrix formed in two steps.

1. Each element in the diagonal matrix is equal to the sum of the conductance (one over the resistance) of each element connected to the corresponding node. So the first diagonal element is the sum of conductances connected to node 1, the second diagonal element is the sum of conductances connected to node 2, and so on.

2. The off diagonal elements are the negative conductance of the element connected to the pair of corresponding node. Therefore a resistor between nodes 1 and 2 goes into the G matrix at location (1,2) and locations (2,1).

If an element is grounded, it will only have contribute to one entry in the G matrix – at the appropriate location on the diagonal. If it is ungrounded it will contribute to four entries in the matrix – two diagonal entries (corresponding to the two nodes) and two off-diagonal entries.

Rules for making the B matrix

The B matrix is an N×M matrix with only 0, 1 and -1 elements. Each location in the matrix corresponds to a particular voltage source (first dimension) or a node (second dimension). If the positive terminal of the ith voltage source is connected to node k, then the element (k,i) in the B matrix is a 1. If the negative terminal of the ith voltage source is connected to node k, then the element (k,i) in the B matrix is a -1. Otherwise, elements of the B matrix are zero.

If a voltage source is ungrounded, it will have two elements in the B matrix (a 1 and a -1 in the same column). If it is grounded it will only have one element in the matrix.

Rules for making the C matrix

The C matrix is an M×N matrix with only 0, 1 and -1 elements. Each location in the matrix corresponds to a particular node (first dimension) or voltage source (second dimension). If the positive terminal of the ith voltage source is connected to node k, then the element (i,k) in the C matrix is a 1. If the negative terminal of the ith voltage source is connected to node k, then the element (i,k) in the C matrix is a -1. Otherwise, elements of the C matrix are zero.

In other words, the C matrix is the transpose of the B matrix. This is not the case when dependent sources are present.

Rules for making the D matrix

The D matrix is an M×M matrix that is composed entirely of zeros. It can be non-zero if dependent sources are considered.

1.1.3 The x matrix

The x matrix holds our unknown quantities and will be developed as the combination of 2 smaller matrices v and j. It is considerably easier to define than the A matrix.

$$x = \begin{bmatrix} v \\ j \end{bmatrix} \tag{1.4}$$

The x matrix is 1×(M+N) (N is the number of nodes, and M is the number of independent voltage sources) and:

- the v matrix is 1×N and hold the unknown voltages
- the j matrix is 1×M and holds the unknown currents through the voltage sources

Rules for making the v matrix

The v matrix is an 1×N matrix formed of the node voltages. Each element in v corresponds to the voltage at the equivalent node in the circuit (there is no entry for ground – node 0).

For a circuit with N nodes we get:

$$v = \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_N \end{bmatrix} \tag{1.5}$$

Rules for making the j matrix

The j matrix is an 1×M matrix, with one entry for the current through each voltage source. So if there are M voltage sources V_1, V_2 through V_M, the j matrix will be:

$$j = \begin{bmatrix} i_{V_1} \\ i_{V_2} \\ \vdots \\ i_{V_M} \end{bmatrix} \tag{1.6}$$

1.1.4 The z matrix

The z matrix holds our independent voltage and current sources and will be developed as the combination of 2 smaller matrices i and e. It is quite easy to formulate.

$$z = \begin{bmatrix} i \\ e \end{bmatrix} \tag{1.7}$$

The z matrix is 1×(M+N) (N is the number of nodes, and M is the number of independent voltage sources) and:

- the i matrix is 1×N and contains the sum of the currents through the passive elements into the corresponding node (either zero, or the sum of independent current sources)
- the e matrix is 1×M and holds the values of the independent voltage sources

Rules for making the i matrix

The i matrix is an 1×N matrix with each element of the matrix corresponding to a particular node. The value of each element of i is determined by the sum of current sources into the corresponding node. If there are no current sources connected to the node, the value is zero.

Rules for making the e matrix

The e matrix is an 1×M matrix with each element of the matrix equal in value to the corresponding independent voltage source.

1.1.5 A simple example

The example given in fig. 1.1 illustrates applying the rules for building the MNA matrices and how this relates to basic equations used in circuit analysis.

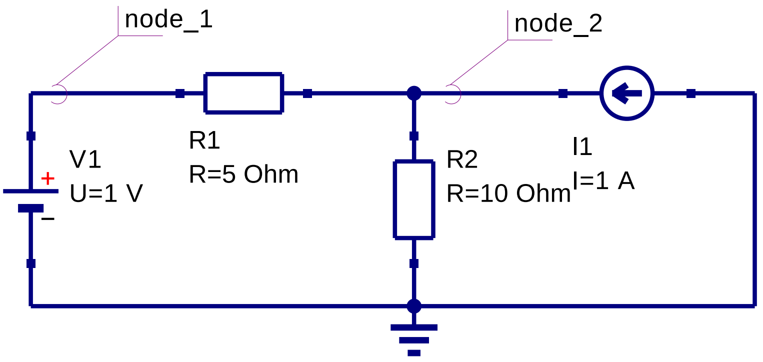


Figure 1.1: example circuit applied to modified nodal analysis

Going through the MNA algorithm

The G matrix is a 2×2 matrix because there are 2 different nodes apart from ground which is the reference node. On the diagonal you find the sum of the elements conductances connected to the nodes 1 and 2. The off-diagonal matrix entries contain the negative conductances of the elements connected between two nodes.

$$G = \begin{bmatrix} \frac{1}{R_1} & -\frac{1}{R_1} \\ -\frac{1}{R_1} & \frac{1}{R_1} + \frac{1}{R_2} \end{bmatrix} = \begin{bmatrix} 0.2 & -0.2 \\ -0.2 & 0.3 \end{bmatrix} \tag{1.8}$$

The B matrix (which is transposed to C) is a 1×2 matrix because there is one voltage source and 2 nodes. The positive terminal of the voltage source V_1 is connected to node 1. That is why

$$B = C^T = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \tag{1.9}$$

and the D matrix is filled with zeros only because there are no dependent (active and controlled) devices in the example circuit.

$$D = \begin{bmatrix} 0 \end{bmatrix} \tag{1.10}$$

The x matrix is a 1×3 matrix. The MNA equations deliver a solution for the unknown voltages at each node in a circuit except the reference node and the currents through each voltage source.

$$x = \begin{bmatrix} v_1 \\ v_2 \\ i_{V_1} \end{bmatrix} \tag{1.11}$$

The z matrix is according to the rules for building it a 1×3 matrix. The upper two entries are the sums of the currents flowing into node 1 and node 2. The lower entry is the voltage value of the voltage source V_1.

$$z = \begin{bmatrix} 0 \\ I_1 \\ U_1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \tag{1.12}$$

According to the MNA algorithm the equation system is represented by

$$[A] \cdot [x] = [z] \tag{1.13}$$

which is equivalent to

$$\begin{bmatrix} G & B \\ C & D \end{bmatrix} \cdot \begin{bmatrix} x \end{bmatrix} = \begin{bmatrix} z \end{bmatrix} \tag{1.14}$$

In the example eq. (1.14) expands to:

$$\begin{bmatrix} \frac{1}{R_1} & -\frac{1}{R_1} & 1 \\ -\frac{1}{R_1} & \frac{1}{R_1} + \frac{1}{R_2} & 0 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ i_{V_1} \end{bmatrix} = \begin{bmatrix} 0 \\ I_1 \\ U_1 \end{bmatrix} \tag{1.15}$$

The equation systems to be solved is now defined by the following matrix representation.

$$\begin{bmatrix} 0.2 & -0.2 & 1 \\ -0.2 & 0.3 & 0 \\ 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} v_1 \\ v_2 \\ i_{V_1} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 1 \end{bmatrix} \tag{1.16}$$

Using matrix inversion the solution vector x writes as follows:

$$[x] = [A]^{-1} \cdot [z] = \begin{bmatrix} v_1 \\ v_2 \\ i_{V_1} \end{bmatrix} = \begin{bmatrix} 1 \\ 4 \\ 0.6 \end{bmatrix} \tag{1.17}$$

The result in eq. (1.17) denotes the current through the voltage source V_1 is 0.6A, the voltage at node 1 is 1V and the voltage at node 2 is 4V.

How the algorithm relates to basic equations in circuit analysis

Expanding the matrix representation in eq. (1.15) to a set of equations denotes the following equation system consisting of 3 of them.

$$\text{I}: \quad 0 = \frac{1}{R_1} \cdot v_1 - \frac{1}{R_1} \cdot v_2 + i_{V_1} \qquad \text{KCL at node 1} \tag{1.18}$$

$$\text{II}: \quad I_1 = -\frac{1}{R_1} \cdot v_1 + \left(\frac{1}{R_1} + \frac{1}{R_2}\right) \cdot v_2 \qquad \text{KCL at node 2} \tag{1.19}$$

$$\text{III}: \quad U_1 = v_1 \qquad \text{constitutive equation} \tag{1.20}$$

Apparently eq. I and eq. II conform to Kirchhoff's current law at the nodes 1 and 2. The last equation is just the constitutive equation for the voltage source V_1. There are three unknowns (v_1, v_2 and i_{V_1}) and three equations, thus the system should be solvable.

Equation III indicates the voltage at node 1 is 1V. Applying this result to eq. II and transposing it to v_2 (the voltage at node 2) gives

$$v_2 = \frac{I_1 + \frac{1}{R_1} \cdot U_1}{\frac{1}{R_1} + \frac{1}{R_2}} = 4\text{V} \tag{1.21}$$

The missing current through the voltage source V_1 can be computed using both the results $v_2 = 4$V and $v_1 = 1$V by transforming equation I.

$$i_{V_1} = \frac{1}{R_1} \cdot v_2 - \frac{1}{R_1} \cdot v_1 = 0.6\text{A} \tag{1.22}$$

The small example, shown in fig. 1.1, and the excursus into artless math verifies that the MNA algorithm and classic electrical handiwork tend to produce the same results.

1.2 Extensions to the MNA

As noted in the previous sections the D matrix is zero and the B and C matrices are transposed each other and filled with either 1, -1 or 0 provided that there are no dependent sources within the circuit. This changes when introducing active (and controlled) elements. Examples are voltage controlled voltage sources, transformers and ideal operational amplifiers. The models are depicted in section 11 and 10

1.3 Non-linear DC Analysis

Previous sections described using the modified nodal analysis solving linear networks including controlled sources. It can also be used to solve networks with non-linear components like diodes and transistors. Most methods are based on iterative solutions of a linearised equation system. The best known is the so called Newton-Raphson method.

1.3.1 Newton-Raphson method

The Newton-Raphson method is going to be introduced using the example circuit shown in fig. 1.2 having a single unknown: the voltage at node 1.

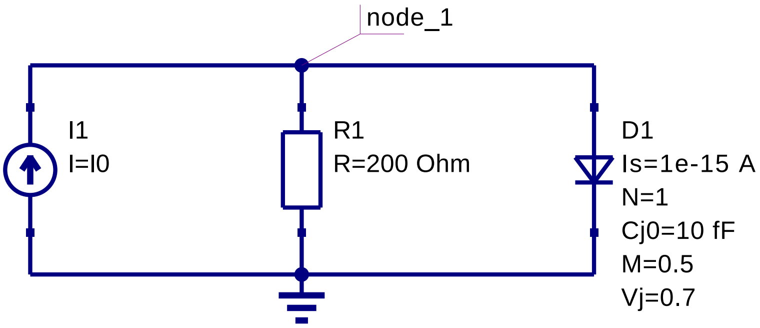


Figure 1.2: example circuit for non-linear DC analysis

The 1x1 MNA equation system to be solved can be written as

$$\left[G\right] \cdot \left[V_1\right] = \left[I_0\right] \tag{1.23}$$

whereas the value for G is now going to be explained. The current through a diode is simply determined by Schockley's approximation

$$I_d = I_S \cdot \left(e^{\frac{V_d}{V_T}} - 1\right) \tag{1.24}$$

Thus Kirchhoff's current law at node 1 can be expressed as

$$I_0 = \frac{V}{R} + I_S \cdot \left(e^{\frac{V}{V_T}} - 1\right) \tag{1.25}$$

By establishing eq. (1.26) it is possible to trace the problem back to finding the zero point of the function f.

$$f(V) = \frac{V}{R} + I_S \cdot \left(e^{\frac{V}{V_T}} - 1\right) - I_0 \tag{1.26}$$

Newton developed a method stating that the zero point of a functions derivative (i.e. the tangent) at a given point is nearer to the zero point of the function itself than the original point. In mathematical terms this means to linearise the function f at a starting value $V^{(0)}$.

$$f\left(V^{(0)} + \Delta V\right) \approx f\left(V^{(0)}\right) + \left.\frac{\partial f\left(V\right)}{\partial V}\right|_{V^{(0)}} \cdot \Delta V \quad \text{with} \quad \Delta V = V^{(1)} - V^{(0)} \tag{1.27}$$

Setting $f(V^{(1)}) = 0$ gives

$$V^{(1)} = V^{(0)} - \frac{f\left(V^{(0)}\right)}{\left.\frac{\partial f\left(V\right)}{\partial V}\right|_{V^{(0)}}} \tag{1.28}$$

or in the general case with m being the number of iteration

$$V^{(m+1)} = V^{(m)} - \frac{f\left(V^{(m)}\right)}{\left.\frac{\partial f\left(V\right)}{\partial V}\right|_{V^{(m)}}} \tag{1.29}$$

This must be computed until $V^{(m+1)}$ and $V^{(m)}$ differ less than a certain barrier.

$$\left|V^{(m+1)} - V^{(m)}\right| < \varepsilon_{abs} + \varepsilon_{rel} \cdot \left|V^{(m)}\right| \tag{1.30}$$

With very small ε_{abs} the iteration would break too early and for little ε_{rel} values the iteration aims to a useless precision for large absolute values of V.

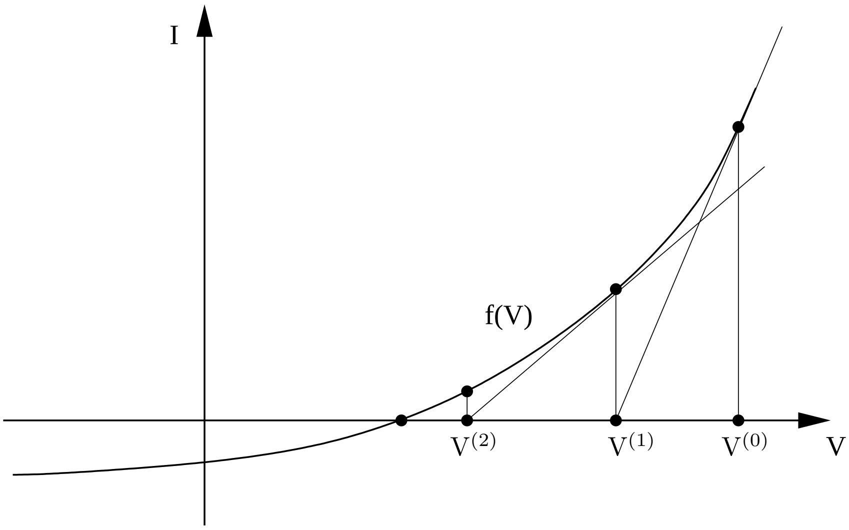


Figure 1.3: Newton-Raphson method for example circuit

With this theoretical background it is now possible to step back to eq. (1.26) being the determining equation for the example circuit. With

$$g_d^{(m)} = \left.\frac{\partial I_d}{\partial V}\right|_{V^{(m)}} = \frac{I_S}{V_T} \cdot e^{\frac{V^{(m)}}{V_T}} \tag{1.31}$$

and

$$\left.\frac{\partial f\left(V\right)}{\partial V}\right|_{V^{(m)}} = \frac{1}{R} + g_d^{(m)} \tag{1.32}$$

the eq. (1.29) can be written as

$$\left(g_d^{(m)} + \frac{1}{R}\right) \cdot V^{(m+1)} = I_0 - \left(I_d^{(m)} - g_d^{(m)} \cdot V^{(m)}\right) \tag{1.33}$$

when the expression

$$f\left(V^{(m)}\right) = \frac{1}{R} \cdot V^{(m)} + I_d^{(m)} - I_0 \tag{1.34}$$

based upon eq. (1.26) is taken into account. Comparing the introductory MNA equation system in eq. (1.23) with eq. (1.33) proposes the following equivalent circuit for the diode model.

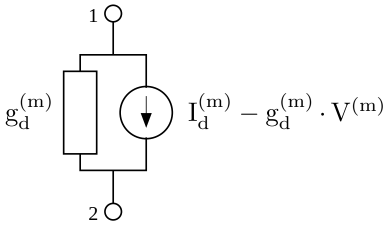


Figure 1.4: accompanied equivalent circuit for intrinsic diode

With

$$I_{eq} = I_d^{(m)} - g_d^{(m)} \cdot V^{(m)} \tag{1.35}$$

the MNA matrix entries can finally be written as

$$\begin{bmatrix} g_d & -g_d \\ -g_d & g_d \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} -I_{eq} \\ I_{eq} \end{bmatrix} \tag{1.36}$$

In analog ways all controlled current sources with non-linear current-voltage dependency built into diodes and transistors can be modeled. The left hand side of the MNA matrix (the A matrix) is called Jacobian matrix which is going to be build in each iteration step. For the solution vector x possibly containing currents as well when voltage sources are in place a likely convergence criteria as defined in eq. (1.30) must be defined for the currents.

1.3.2 Multi-dimensional Newton-Raphson method

Having understood the one-dimensional example, it is now only a small step to the general multi-dimensional algorithm: The node voltage becomes a vector $\mathbf{V}^{(m)}$, factors become the corresponding matrices and differentiations become Jacobian matrices.

The function whose zero must be found is the transformed MNA equation 1.23:

$$\mathbf{f}(\mathbf{V}^{(m)}) = \mathbf{G} \cdot \mathbf{V}^{(m)} - \mathbf{I}_0^{(m)} \tag{1.37}$$

The only difference to the linear case is that the vector $\mathbf{I}_0$ also contains the currents flowing out of the non-linear components. The iteration formula of the Newton-Raphson method writes:

$$\mathbf{V}^{(m+1)} = \mathbf{V}^{(m)} - \left(\left.\frac{\partial \mathbf{f}(\mathbf{V})}{\partial \mathbf{V}}\right|_{V^{(m)}} \right)^{-1} \cdot \mathbf{f}(\mathbf{V}^{(m)}) \tag{1.38}$$

Note that the Jacobian matrix is nothing else but the real part of the MNA matrix for the AC analysis:

$$\mathbf{J}^{(m)} = \left.\frac{\partial \mathbf{f}(\mathbf{V})}{\partial \mathbf{V}}\right|_{\mathbf{V}^{(m)}} = \mathbf{G} - \left.\frac{\partial \mathbf{I}_0}{\partial \mathbf{V}}\right|_{\mathbf{V}^{(m)}} = \mathbf{G} - \mathbf{J}_{nl}^{(m)} = \mathrm{Re}\left(\mathbf{G}_{AC}\right) \tag{1.39}$$

where the index nl denotes the non-linear terms only. Putting equation 1.39 into equation 1.38 and multiplying it with the Jacobian matrix leads to

$$\begin{aligned} \mathbf{J}^{(m)} \cdot \mathbf{V}^{(m+1)} &= \mathbf{J}^{(m)} \cdot \mathbf{V}^{(m)} - \mathbf{f}(\mathbf{V}^{(m)}) && (1.40) \\ &= \left(\mathbf{G} - \mathbf{J}_{nl}^{(m)}\right) \cdot \mathbf{V}^{(m)} - \mathbf{G} \cdot \mathbf{V}^{(m)} + \mathbf{I}_0^{(m)} && (1.41) \\ &= -\mathbf{J}_{nl}^{(m)} \cdot \mathbf{V}^{(m)} + \mathbf{I}_0^{(m)} && (1.42) \end{aligned}$$

So, bringing the Jacobian back to the right side results in the new iteration formula:

$$\mathbf{V}^{(m+1)} = \left(\mathbf{J}^{(m)}\right)^{-1} \cdot \left(-\mathbf{J}_{nl}^{(m)} \cdot \mathbf{V}^{(m)} + \mathbf{I}_0^{(m)}\right) \tag{1.43}$$

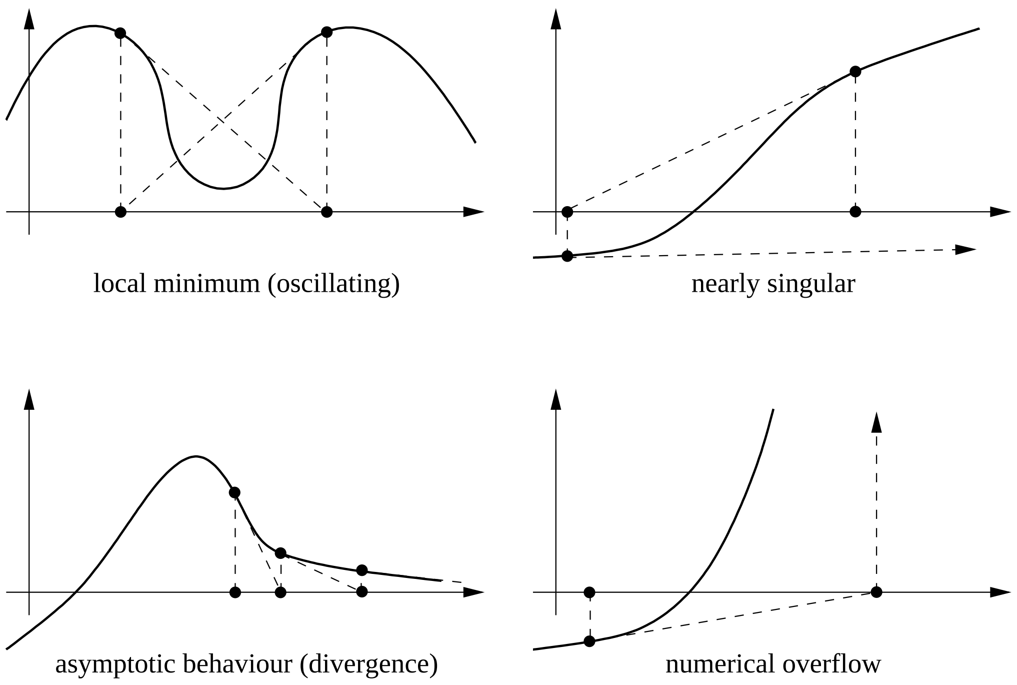
The negative sign in front of $\mathbf{J}_{nl}$ is due to the definition of $\mathbf{I}_0$ flowing out of the component. Note that $\mathbf{I}_0^{(m)}$ still contains contributions of linear and non-linear current sources.

1.4 Convergence

The implications during Newton-Raphson iterations solving the linear equation system

$$\left[A\left(x_k\right)\right] \cdot \left[x_{k+1}\right] = \left[z\left(x_k\right)\right] \tag{1.44}$$

are continuous device model equations (with continuous derivatives as well), floating nodes (make the Jacobian matrix A singular) and the initial guess x_0. The convergence problems which in fact occur are local minimums causing the matrix A to be singular, nearly singular matrices and overflow problems.



local minimum (oscillating)

nearly singular

asymptotic behaviour (divergence)

numerical overflow

1.4.1 Limiting schemes

The modified (damped) Newton-Raphson schemes are based on the limitation of the solution vector x^k in each iteration.

$$x^{k+1} = x^k + \alpha \cdot \Delta x^{k+1} \qquad \text{with} \qquad \Delta x^{k+1} = x^{k+1} - x^k \tag{1.45}$$

One possibility to choose a value for $\alpha \in [0, 1]$ is

$$\alpha = \frac{\gamma}{\left\|\Delta x^{k+1}\right\|_\infty} \tag{1.46}$$

This is a heuristic and does not ensure global convergence, but it can help solving some of the discussed problems. Another possibility is to pick a value α^k which minimizes the L_2 norm of the right hand side vector. This method performs a one-dimensional (line) search into Newton direction and guarantees global convergence.

$$x^{k+1} = x^k + \alpha^k \cdot \Delta x^{k+1} \qquad \text{with an } \alpha^k \text{ which minimizes} \qquad \left\|z\left(x^k + \alpha^k \cdot \Delta x^{k+1}\right)\right\|_2 \tag{1.47}$$

The one remaining problem about that line search method for convergence improvement is its iteration into local minimums where the Jacobian matrix is singular. The damped Newton-Raphson method "pushes" the matrix into singularity as depicted in fig. 1.5.

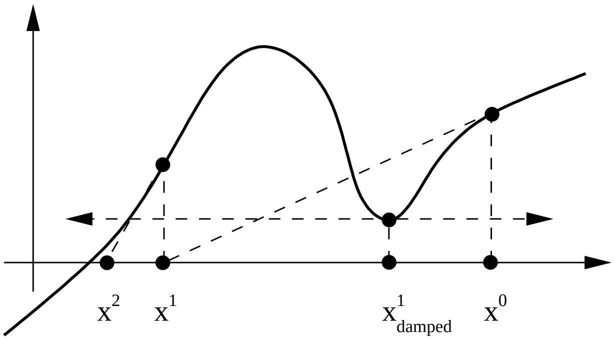


Figure 1.5: singular Jacobian problem

1.4.2 Continuation schemes

The basic idea behind this Newton-Raphson modification is to generate a sequence of problems such that a problem is a good initial guess for the following one, because Newton basically converges given a close initial guess.

The template algorithm for this modification is to solve the equation system

$$[A] \cdot [x] - [z] = 0 \tag{1.48}$$

$$F(x(\lambda), \lambda) = 0 \tag{1.49}$$

with the parameter $\lambda \in [0, 1]$ given that $x(\lambda)$ is sufficiently smooth. $F(x(0), 0)$ starts the continuation and $F(x(1), 1)$ ends the continuation. The algorithm outline is as follows: First solve the problem $F(x(0), 0)$, e.g. set $\lambda = \Delta\lambda = 0.01$ and try to solve $F(x(\lambda), \lambda)$. If Newton-Raphson converged then increase λ by $\Delta\lambda$ and double $\Delta\lambda = 2 \cdot \Delta\lambda$, otherwise half $\Delta\lambda = 0.5 \cdot \Delta\lambda$ and set $\lambda = \lambda_{prev} + \Delta\lambda$. Repeat this until $\lambda = 1$.

Source stepping

Applied to the solution of (non-linear) electrical networks one may think of $\alpha \in [0, 1]$ as a multiplier for the source vector S yielding $S(\alpha) = \alpha S$. Varying α form 0 to 1 and solve at each α. The actual circuit solution is done when $\alpha = 1$. This method is called source stepping. The solution vector $x(\alpha)$ is continuous in α (hence the name continuation scheme).

Minimal derivative stepping

Another possibility to improve convergence of almostly singular electrical networks is the so called g_{min} stepping, i.e. adding a tiny conductance to ground at each node of the Jacobian A matrix. The continuation starts e.g. with $g_{min} = 0.01$ and ends with $g_{min} = 0$ reached by the algorithm described in section 1.4.2. The equation system is slightly modified by adding the current g_{min} to each diagonal element of the matrix A.

1.4.3 Improved component models

Linearising the exponential diode eq. (1.54) in the forward region a numerical overflow can occur. The diagram in fig. 1.6 visualises this situation. Starting with $V^{(0)}$ the next iteration value gets $V^{(1)}$ which results in an indefinite large diode current. It can be limited by iterating in current instead of voltage when the computed voltage exceeds a certain value.

How this works is going to be explained using the diode model shown in fig. 1.4. When iterating in voltage (as normally done) the new diode current is

$$\hat{I}_d^{(m+1)} = g_d^{(m)} \left(\hat{V}^{(m+1)} - V^{(m)}\right) + I_d^{(m)} \tag{1.50}$$

The computed value $\hat{V}^{(m+1)}$ in iteration step $m+1$ is not going to be used for the following step when $V^{(m)}$ exceeds the critical voltage V_{CRIT} which gets explained in the below paragraphs. Instead, the value resulting from

$$I_d^{(m+1)} = I_S \cdot \left(e^{\frac{V^{(m+1)}}{nV_T}} - 1\right) \tag{1.51}$$

is used (i.e. iterating in current). With

$$\hat{I}_d^{(m+1)} \overset{!}{=} I_d^{(m+1)} \qquad \text{and} \qquad g_d^{(m)} = \frac{I_S}{n \cdot V_T} \cdot e^{\frac{V^{(m)}}{n \cdot V_T}} \tag{1.52}$$

the new voltage can be written as

$$V^{(m+1)} = V^{(m)} + nV_T \cdot \ln\left(\frac{\hat{V}^{(m+1)} - V^{(m)}}{nV_T} + 1\right) \tag{1.53}$$

Proceeding from Shockley's simplified diode equation the critical voltage is going to be defined. The explained algorithm can be used for all exponential DC equations used in diodes and transistors.

$$I\left(V\right) = I_S \cdot \left(e^{\frac{V}{nV_T}} - 1\right) \tag{1.54}$$

$$y\left(x\right) = f\left(x\right) \tag{1.55}$$

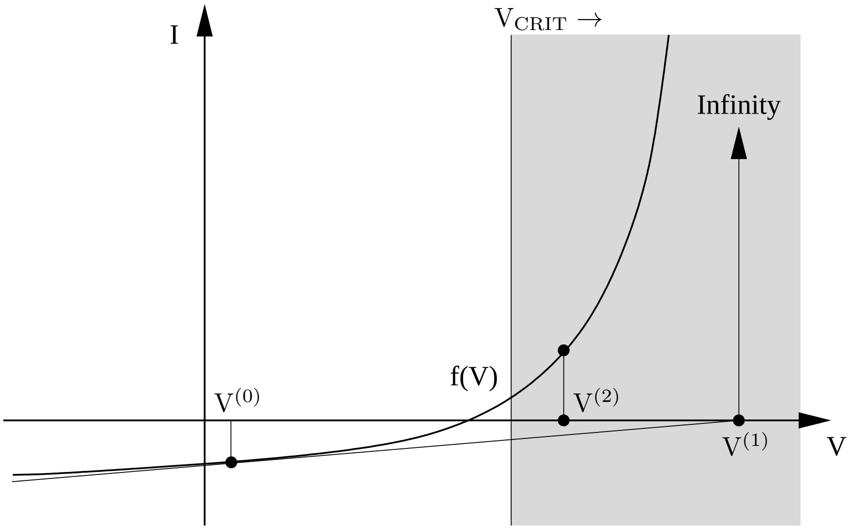


Figure 1.6: numerical problem with Newton-Raphson algorithm

The critical voltage V_{CRIT} is the voltage where the curve radius of eq. (1.54) has its minimum with I and V having equally units. The curve radius R for the explicit definition in eq. (1.55) can be written as

$$R = \left| \frac{\left(1 + \left(\frac{dy}{dx}\right)^2\right)^{3/2}}{\frac{d^2y}{dx^2}} \right| \tag{1.56}$$

Finding this equations minimum requires the derivative.

$$\frac{dR}{dx} = \frac{\frac{d^2y}{dx^2} \cdot \frac{3}{2}\left(1 + \left(\frac{dy}{dx}\right)^2\right)^{1/2} \cdot 2 \cdot \frac{dy}{dx} \cdot \frac{d^2y}{dx^2} - \left(1 + \left(\frac{dy}{dx}\right)^2\right)^{3/2} \cdot \frac{d^3y}{dx^3}}{\left(\frac{d^2y}{dx^2}\right)^2} \tag{1.57}$$

The diagram in fig. 1.7 shows the graphs of eq. (1.56) and eq. (1.57) with $n = 1$, $I_S = 100\text{nA}$ and $V_T = 25\text{mV}$.

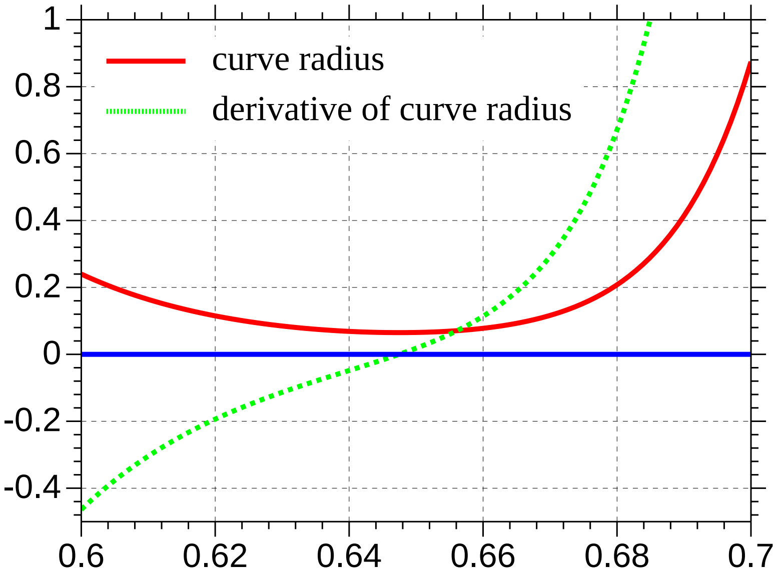


Figure 1.7: curve radius of exponential diode curve and its derivative

With the following higher derivatives of eq. (1.54)

$$\frac{dI\left(V\right)}{dV} = \frac{I_S}{nV_T} \cdot e^{\frac{V}{nV_T}} \tag{1.58}$$

$$\frac{d^2I\left(V\right)}{dV^2} = \frac{I_S}{n^2V_T^2} \cdot e^{\frac{V}{nV_T}} \tag{1.59}$$

$$\frac{d^3I\left(V\right)}{dV^3} = \frac{I_S}{n^3V_T^3} \cdot e^{\frac{V}{nV_T}} \tag{1.60}$$

the critical voltage results in

$$\frac{dR}{dx} \overset{!}{=} 0 = 3 - \frac{n^2V_T^2}{I_S^2} \cdot e^{-2\frac{V}{nV_T}} - 1 \quad \rightarrow \quad V_{CRIT} = nV_T \cdot \ln\left(\frac{nV_T}{I_S\sqrt{2}}\right) \tag{1.61}$$

In order to avoid numerical errors a minimum value of the pn-junction's derivative (i.e. the currents tangent in the operating point) g_{min} is defined. On the one hand this avoids very large deviations of the appropriate voltage in the next iteration step in the backward region of the pn-junction and on the other hand it avoids indefinite large voltages if g_d itself suffers from numerical errors and approaches zero.

The quadratic input I-V curve of field-effect transistors as well as the output characteristics of these devices can be handled in similar ways. The limiting (and thereby improving the convergence behaviour) algorithm must somehow ensure that the current and/or voltage deviation from one iteration step to the next step is not too large a value. Because of the wide range of existing variations how these curves are exactly modeled there is no standard strategy to achieve this. Anyway, the threshold voltage V_{Th} should play an important role as well as the direction which the current iteration step follows.

1.5 Overall solution algorithm for DC Analysis

In this section an overall solution algorithm for a DC analysis for linear as well as non-linear networks is given. With non-linear network elements at hand the Newton-Raphson (NR) algorithm is applied.

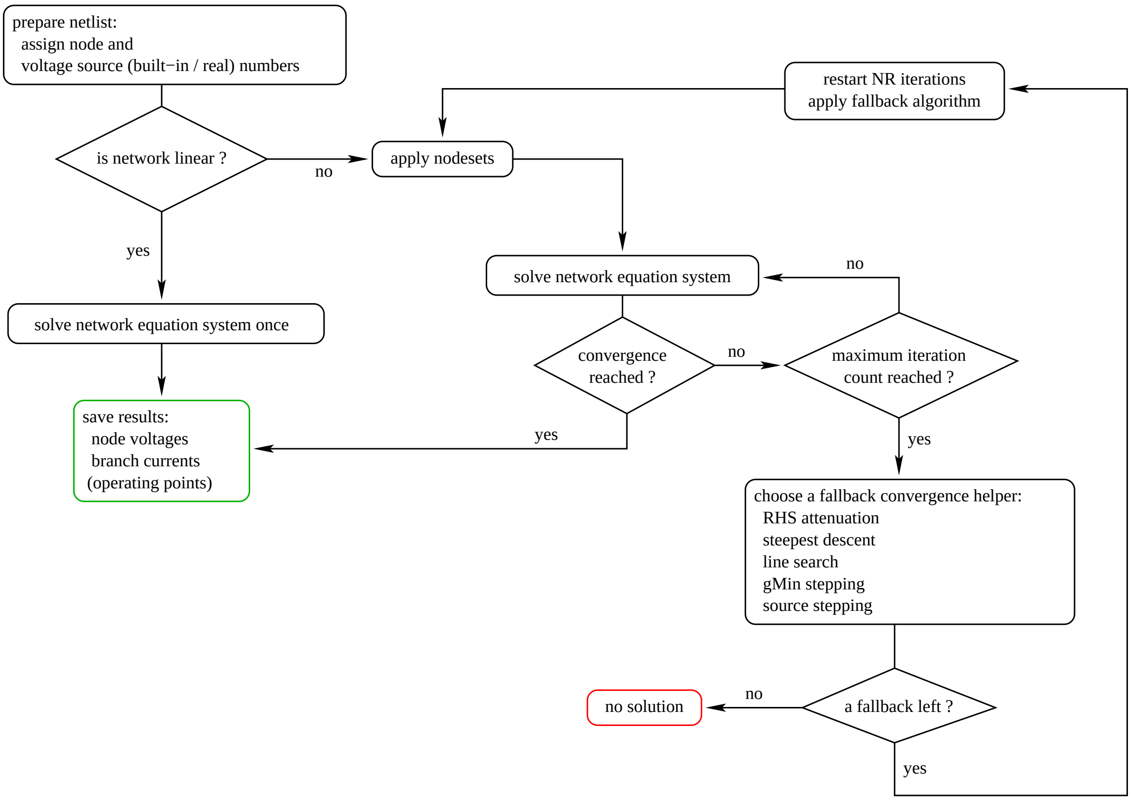


Figure 1.8: DC solution algorithm flow chart

The algorithm shown in fig. 1.8 has been proved to be able to find DC solutions for a large variety of networks. It must be said that the application of any of the fallback convergence helpers indicates a nearly or definitely singular equation system (e.g. floating nodes or overdetermining sources). The convergence problems are either due to an apparently "wrong" network topology or to the model implementation of non-linear components. For some of the problems also refer to the facts mentioned in section 19.2 on page 251. In some cases it may even occur that tiny numerical inaccuracies lead to non-convergences whereas the choice of a more accurate (but probably slower) equation system solver can help. With network topologies having more than a single stable solution (e.g. bistable flip-flops) it is recommended to apply nodesets, i.e. forcing the Newton-Raphson iteration into a certain direction by initial values.

When having problems to get a circuit have its DC solution the following actions can be taken to solve these problems.

- check circuit topology (e.g. floating nodes or overdetermining sources)
- check model parameters of non-linear components
- apply nodesets
- choose a more accurate equation system solver
- relax the convergence tolerances if possible
- increase the maximum iteration count
- choose the prefered fallback algorithm

The presented concepts are common to most circuit simulators each having to face the mentioned aspects. And probably facing it in a different manner with more or less big differences in their implementation details especially regarding the (fallback) convergence helpers. None of the algorithms based on Newton-Raphson ensures global convergence, thus very few documents have been published either for the complexity of the topic or for uncertainties in the detailed implementation each carrying the attribute “can help” or “may help”.

So for now the application of a circuit simulator to find the DC solution of a given network sometimes keeps being a task for people knowing what they want to achieve and what they can roughly expect.

Chapter 2

AC Analysis

The AC analysis is a small signal analysis in the frequency domain. Basically this type of simulation uses the same algorithms as the DC analysis (section 1.1 on page 7). The AC analysis is a linear modified nodal analysis. Thus no iterative process is necessary. With the Y-matrix of the components, i.e. now a complex matrix, and the appropriate extensions it is necessary to solve the equation system (2.1) similar to the (linear) DC analysis.

$$[A] \cdot [x] = [z] \qquad \text{with} \qquad A = \begin{bmatrix} Y & B \\ C & D \end{bmatrix} \tag{2.1}$$

Non-linear components have to be linearized at the DC bias point. That is, before an AC simulation with non-linear components can be performed, a DC simulation must be completed successfully. Then, the MNA stamp of the non-linear components equals their entries of the Jacobian matrix, which was already computed during the DC simulation. In addition to this real-valued elements, a further stamp has to be applied: The Jacobian matrix of the non-linear charges multiplied by $j\omega$ (see also section 11.8).

Chapter 3

AC Noise Analysis

3.1 Definitions

First some definition must be done:

Reciprocal Networks:
Two networks A and B are reciprocal to each other if their transimpedances have the following relation:

$$Z_{mn,A} = Z_{nm,B} \tag{3.1}$$

That means: Drive the current I into node n of circuit A and at node m the voltage $I \cdot Z_{mn,A}$ appears. In circuit B it is just the way around.

Adjoint Networks:
Network A and network B are adjoint to each other if the following equation holds for their MNA matrices:

$$[A]^T = [B] \tag{3.2}$$

3.2 The Algorithm

To calculate the small signal noise of a circuit, the AC noise analysis has to be applied [1]. This technique uses the principle of the AC analysis described in chapter 2 on page 22. In addition to the MNA matrix A one needs the noise current correlation matrix $\underline{C}_Y$ of the circuit, that contains the equivalent noise current sources for every node on its main diagonal and their correlation on the other positions.

The basic concept of the AC noise analysis is as follows: The noise voltage at node i should be calculated, so the voltage arising due to the noise source at node j is calculated first. This has to be done for every n nodes and after that adding all the noise voltages (by paying attention to their correlation) leads to the overall voltage. But that would mean to solve the MNA equation n times. Fortunately there is a more easy way. One can perform the above-mentioned n steps in one single step, if the reciprocal MNA matrix is used. This matrix equals the MNA matrix itself, if the network is reciprocal. A network that only contains resistors, capacitors, inductors, gyrators and transformers is reciprocal.

The question that needs to be answered now is: How to get the reciprocal MNA matrix for an arbitrary network? This is equivalent to the question: How to get the MNA matrix of the adjoint network. The answer is quite simple: Just transpose the MNA matrix!

For any network, calculating the noise voltage at node i is done by the following three steps:

1. Solving MNA equation: $$[A]^T \cdot [x] = [A]^T \cdot [v] = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ -1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \\ \\ \\ \leftarrow i\text{-th row} \\ \\ \\ \\ \end{matrix} \tag{3.3}$$

2. Creating noise correlation matrix: $$(\underline{C}_Y) \tag{3.4}$$

3. Calculating noise voltage: $$v_{noise,i} = \sqrt{[v]^T \cdot (\underline{C}_Y) \cdot [v]^*} \tag{3.5}$$

If the correlation between several noise voltages is also wanted, the procedure is straight forward: Perform step 1 for every desired node, put the results into a matrix and replace the vector $[v]$ in step 3 by this matrix. This results in the complete correlation matrix. Indeed, the above-mentioned algorithm is only a specialisation of transforming the noise current correlation matrix $(\underline{C}_Y)$ into the noise voltage correlation matrix $(\underline{C}_Z)$ (see section 5.5.1).

If the normal AC analysis has already be done with LU decomposition, then the most time consuming work of step 1 has already be done.

$$\text{instead of} \qquad Y = L \cdot U \qquad \text{we have} \qquad Y^T = U^T \cdot L^T \tag{3.6}$$

I.e. U^T becomes the new L matrix and L^T becomes the new U matrix, and the matrix equation do not need to be solved again, because only the right-hand side was changed. So altogether this is a quickly done task. This is also true for every further node whose noise voltage should be calculated, because only the right-hand side of step 1 changes, i.e. LU substitution is needed only.

When reusing the LU decomposed MNA matrix of the usual AC analysis some issues must be considered. The decomposition representation changes during the AC noise analysis as the matrix A gets transposed. This means:

$$A = L \cdot U \;\; \text{with} \;\; L = \begin{bmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ l_{n1} & \dots & \dots & l_{nn} \end{bmatrix} \;\; \text{and} \;\; U = \begin{bmatrix} 1 & u_{12} & \dots & u_{1n} \\ 0 & 1 & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & 1 \end{bmatrix} \tag{3.7}$$

becomes

$$A^T = U^T \cdot L^T \;\; \text{with} \;\; L = \begin{bmatrix} 1 & 0 & \dots & 0 \\ l_{21} & 1 & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ l_{n1} & \dots & \dots & 1 \end{bmatrix} \;\; \text{and} \;\; U = \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & u_{nn} \end{bmatrix} \tag{3.8}$$

Thus the forward substitution (as described in section 19.2.4) and the backward substitution (as described in section 19.2.4) must be slightly modified, because now the diagonal elements l_{ii} can be neglected in the forward substitution but the u_{ii} elements must be considered in the backward substitution.

$$y_i = z_i - \sum_{k=1}^{i-1} y_k \cdot l_{ik} \qquad\qquad i = 1, \dots, n \tag{3.9}$$

$$x_i = \frac{y_i}{u_{ii}} - \sum_{k=i+1}^{n} x_k \cdot \frac{u_{ik}}{u_{ii}} \qquad i = n, \ldots, 1 \tag{3.10}$$

Another issue is the row exchanging of the ac matrix A. After transposing of A, the exchanged rows become exchanged columns. This means, that the right-hand side of equation 3.3 must not be fitted accordingly, but the result vector $[v]$ must.

3.2.1 A Simple Example

The network that is depicted in figure 3.1 is given. The MNA equation is (see chapter 1.1):

$$[A] \cdot [x] = \begin{bmatrix} 1/R_1 & 0 \\ G & 1/R_2 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3.11}$$

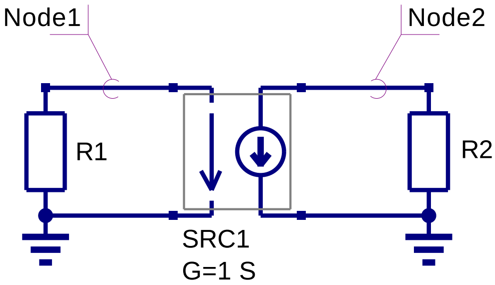


Figure 3.1: simple non-reciprocal network

Because of the controlled current source, the circuit is not reciprocal. The noise voltage at node 2 is the one to search for. Yes, this is very easy to calculate, because it is a simple example, but the algorithm described above should be used. This can be achived by solving the equations

$$\begin{bmatrix} 1/R_1 & 0 \\ G & 1/R_2 \end{bmatrix} \cdot \begin{bmatrix} Z_{11} \\ Z_{21} \end{bmatrix} = \begin{bmatrix} -1 \\ 0 \end{bmatrix} \tag{3.12}$$

and

$$\begin{bmatrix} 1/R_1 & 0 \\ G & 1/R_2 \end{bmatrix} \cdot \begin{bmatrix} Z_{12} \\ Z_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \tag{3.13}$$

So, the MNA matrix must be solved two times: First to get the transimpedance from node 1 to node 2 (i.e. Z_{21}) and second to get the transimpedance from node 2 to node 2 (i.e. Z_{22}). But why solving it two times, if only one voltage should be calculated? With every step transimpedances are calculated that are not need. Is there no more effective way?

Fortunately, there is Tellegen's Theorem: A network and its adjoint network are reciprocal to each other. That is, transposing the MNA matrix leads to the one of the reciprocal network. To check it out:

$$[A]^T \cdot [x] = \begin{bmatrix} 1/R_1 & G \\ 0 & 1/R_2 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3.14}$$

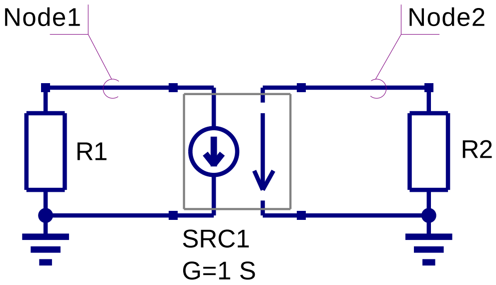


Figure 3.2: simple network to compare with adjoint network

Compare the transposed matrix with the reciprocal network in figure 3.2. It is true! But now it is:

$$\begin{bmatrix} 1/R_1 & G \\ 0 & 1/R_2 \end{bmatrix} \cdot \begin{bmatrix} Z_{12,reciprocal} \\ Z_{22,reciprocal} \end{bmatrix} = \begin{bmatrix} 1/R_1 & G \\ 0 & 1/R_2 \end{bmatrix} \cdot \begin{bmatrix} Z_{21} \\ Z_{22} \end{bmatrix} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \tag{3.15}$$

Because Z_{21} of the original network equals Z_{12} of the reciprocal network, the one step delivers exactly what is needed. So the next step is:

$$([A]^T)^{-1} \cdot \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} R_1 & -G \cdot R_1 \cdot R_2 \\ 0 & R2 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ -1 \end{bmatrix} = \begin{bmatrix} G \cdot R_1 \cdot R_2 \\ -R_2 \end{bmatrix} = \begin{bmatrix} Z_{21} \\ Z_{22} \end{bmatrix} \tag{3.16}$$

Now, as the transimpedances are known, the noise voltage at node 2 can be computed. As there is no correlation, it writes as follows:

$$\begin{aligned} < v_{node2}^2 > &= < v_{R1,node2}^2 > + < v_{R2,node2}^2 > && (3.17) \\ &= < i_{R1}^2 > \cdot Z_{21} \cdot Z_{21}^* + < i_{R2}^2 > \cdot Z_{22} \cdot Z_{22}^* && (3.18) \\ &= \frac{4 \cdot k \cdot T \cdot \Delta f}{R_1} \cdot (G \cdot R_1 \cdot R_2)^2 + \frac{4 \cdot k \cdot T \cdot \Delta f}{R_2} \cdot (-R_2)^2 && (3.19) \\ &= 4 \cdot k \cdot T \cdot \Delta f \cdot \big(R_1 \cdot (G \cdot R_2)^2 + R_2\big) && (3.20) \end{aligned}$$

That's it. Yes, this could have be computed more easily, but now the universal algorithm is also clear.

3.3 Noise Current Correlation Matrix

The sections 10 and 11 show the noise current correlation matrices of noisy components. The equations are built for RMS noise currents with 1Hz bandwidth.

Chapter 4

Scattering parameters

4.1 Introduction and definition

Voltage and current are hard to measure at very high frequencies. Short and open circuits (used to define most n-port parameters) are hard to realize at high frequencies. Therefore, microwave engineers work with so-called scattering parameters (s-parameters), that uses waves and matched terminations (normally 50Ω). This procedure also minimizes reflection problems.

A (normalized) power wave is defined as ingoing wave $\underline{a}$ or outgoing wave $\underline{b}$ [2]:

$$\underline{a} = \underbrace{\frac{\underline{u} + \underline{Z}_0 \cdot \underline{i}}{2}}_{\underline{U}_{forward}} \cdot \frac{1}{\sqrt{|\text{Re}\underline{Z}_0)|}} \qquad \underline{b} = \underbrace{\frac{\underline{u} - \underline{Z}_0^* \cdot \underline{i}}{2}}_{\underline{U}_{backward}} \cdot \frac{1}{\sqrt{|\text{Re}\underline{Z}_0)|}} \tag{4.1}$$

where $\underline{u}$ is (effective) voltage, $\underline{i}$ (effective) current flowing into the device and $\underline{Z}_0$ reference impedance. The actual power delivered by the source to the load is as follows.

$$P = \left(|\underline{a}|^2 - |\underline{b}|^2\right) \tag{4.2}$$

From the above equation, it becomes clear that $|\underline{a}|^2$ is the available power, i.e. the maximum power that the source can deliver. Sometimes waves are defined with peak voltages and peak currents. The only difference that appears then is the relation to power:

$$P = \frac{1}{2} \cdot \left(|\underline{a}|^2 - |\underline{b}|^2\right) \tag{4.3}$$

Now, characterizing an n-port is straight-forward:

$$\begin{pmatrix} \underline{b}_1 \\ \vdots \\ \underline{b}_n \end{pmatrix} = \begin{pmatrix} \underline{S}_{11} & \cdots & \underline{S}_{1n} \\ \vdots & \ddots & \vdots \\ \underline{S}_{n1} & \cdots & \underline{S}_{nn} \end{pmatrix} \cdot \begin{pmatrix} \underline{a}_1 \\ \vdots \\ \underline{a}_n \end{pmatrix} \tag{4.4}$$

One final note: The reference impedance $\underline{Z}_0$ can be arbitrary chosen. Normally, it's real-valued (e.g. 50Ω), and there is no urgent reason to use a complex one. The definitions in equation 4.1, however, are made form complex impedances. These waves relate to power, but they differ from the voltage and current waves introduced in the following chapter. For real reference impedances both definitions equal each other.

4.2 Waves on Transmission Lines

This section should derive the existence of the voltage and current waves on a transmission line. This way, it also proofs that the definitions from the last section make sense.

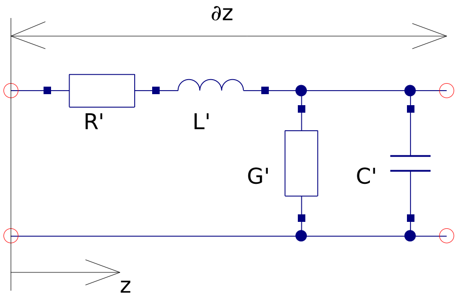


Figure 4.1: Infinite short piece of transmission line

Figure 4.1 shows the equivalent circuit of an infinite short piece of an arbitrary transmission line. The names of the components all carry a single quotation mark which indicates a per-length quantity. Thus, the units are ohms/m for R', henry/m for L', siemens/m for G' and farad/m for C'. Writing down the change of voltage and current across a piece with length ∂z results in the transmission line equations.

$$\frac{\partial u}{\partial z} = -R' \cdot i(z) - L' \cdot \frac{\partial i}{\partial t} \tag{4.5}$$

$$\frac{\partial i}{\partial z} = -G' \cdot u(z) - C' \cdot \frac{\partial u}{\partial t} \tag{4.6}$$

Transforming these equations into frequency domain leads to:

$$\frac{\partial \underline{U}}{\partial z} = -\underline{I}(z) \cdot (R' + j\omega L') \tag{4.7}$$

$$\frac{\partial \underline{I}}{\partial z} = -\underline{U}(z) \cdot (G' + j\omega C') \tag{4.8}$$

Taking equation 4.8 and setting it into the first derivative of equation 4.7 creates the wave equation:

$$\frac{\partial^2 \underline{U}}{\partial z^2} = \underline{\gamma}^2 \cdot \underline{U} \tag{4.9}$$

with $\underline{\gamma}^2 = (\alpha + j\beta)^2 = (R' + j\omega L') \cdot (G' + j\omega C')$. The complete solution of the wave equation is:

$$\underline{U}(z) = \underbrace{\underline{U}_1 \cdot \exp(-\underline{\gamma} \cdot z)}_{\underline{U}_f(z)} + \underbrace{\underline{U}_2 \cdot \exp(\underline{\gamma} \cdot z)}_{\underline{U}_b(z)} \tag{4.10}$$

As can be seen, there is a voltage wave $\underline{U}_f(z)$ travelling forward (in positive z direction) and there is a voltage wave $\underline{U}_b(z)$ travelling backwards (in negative z direction). By setting equation 4.10 into equation 4.7, it becomes clear that the current behaves in the same way:

$$\underline{I}(z) = \underbrace{\frac{\underline{\gamma}}{R' + j\omega L'}}_{\underline{Y}_L} \cdot \left(\underline{U}_f(z) - \underline{U}_b(z)\right) =: \underline{I}_f(z) + \underline{I}_b(z) \tag{4.11}$$

Note that both current waves are counted positive in positive z direction. In literature, the backward flowing current wave $\underline{I}_b(z)$ is sometime counted the other way around which would avoid the negative sign within some of the following equations.
Equation 4.11 introduces the characteristic admittance $\underline{Y}_L$. The propagation constant $\underline{\gamma}$ and the characteristic impedance $\underline{Z}_L$ are the two fundamental properties describing a transmission line.

$$\underline{Z}_L = \frac{1}{\underline{Y}_L} = \frac{\underline{U}_f}{\underline{I}_f} = -\frac{\underline{U}_b}{\underline{I}_b} = \sqrt{\frac{R' + j\omega L'}{G' + j\omega C'}} \approx \sqrt{\frac{L'}{C'}} \tag{4.12}$$

Note that $\underline{Z}_L$ is real-valued if the line loss (due to R' and G') is zero. Usually, this is a good approximation in reality. The combination of equation equation 4.11 and 4.12 shows what happens at the end of a transmission line, if it's terminated with the impedance $\underline{Z}_e$:

$$\underline{Z}_e = \frac{\underline{U}_e}{\underline{I}_e} = \frac{\underline{U}_f + \underline{U}_b}{\underline{I}_f + \underline{I}_b} = \frac{\underline{U}_f + \underline{U}_b}{\underline{I}_f - \underline{U}_b/\underline{Z}_L} = \underline{Z}_L \cdot \frac{\underline{U}_f + \underline{U}_b}{\underline{Z}_L \cdot \underline{I}_f - \underline{U}_b} = \underline{Z}_L \cdot \frac{1 + \underline{U}_b/\underline{U}_f}{1 - \underline{U}_b/\underline{U}_f} \tag{4.13}$$

$$\Rightarrow \quad \frac{\underline{Z}_e}{\underline{Z}_L} \cdot \left(1 - \frac{\underline{U}_b}{\underline{U}_f}\right) = 1 + \frac{\underline{U}_b}{\underline{U}_f} \quad \Rightarrow \quad \frac{\underline{Z}_e}{\underline{Z}_L} - 1 = \left(\frac{\underline{Z}_e}{\underline{Z}_L} + 1\right) \cdot \frac{\underline{U}_b}{\underline{U}_f} \tag{4.14}$$

As can be seen, the ration of $\underline{Z}_e$ and $\underline{Z}_e$ determines the ratio of forward and backward travelling voltage wave. Therefore, this is an important quantity which is named reflection coefficient $\underline{r}$:

$$\underline{r} = \frac{\underline{U}_b}{\underline{U}_f} = -\frac{\underline{I}_b}{\underline{I}_f} = \frac{\underline{Z}_e - \underline{Z}_L}{\underline{Z}_e + \underline{Z}_L} = \frac{\underline{Z}_e/\underline{Z}_L - 1}{\underline{Z}_e/\underline{Z}_L + 1} = \frac{\underline{U}_e - \underline{Z}_L \cdot \underline{I}_e}{\underline{U}_e + \underline{Z}_L \cdot \underline{I}_e} \tag{4.15}$$

This means that a part of the voltage and current wave is reflected back if the end of a transmission line is not terminated by an impedance that equals $\underline{Z}_L$. The same effect occurs in the middle of a transmission line, if its characteristic impedance changes.

$\underline{U} = \underline{U}_f + \underline{U}_b$	$\underline{I} = \underline{I}_f + \underline{I}_b$
$\underline{U}_f = \frac{1}{2} \cdot (\underline{U} + \underline{I} \cdot \underline{Z}_L)$	$\underline{I}_f = \frac{1}{2} \cdot (\underline{U}/\underline{Z}_L + \underline{I})$
$\underline{U}_b = \frac{1}{2} \cdot (\underline{U} - \underline{I} \cdot \underline{Z}_L)$	$\underline{I}_b = \frac{1}{2} \cdot (\underline{I} - \underline{U}/\underline{Z}_L)$

Note that for power waves the reflection coefficient has a slightly different definition, because the maximum output power is reached with a conjugate-complex termination:

$$\underline{r} = \frac{\underline{Z}_e - \underline{Z}_L^*}{\underline{Z}_e + \underline{Z}_L} \tag{4.16}$$

4.3 Computing with S-parameters

4.3.1 S-parameters in CAE programs

The most common task of a simulation program is to compute the S parameters of an arbitrary network that consists of many elementary components connected to each other. To perform this, one can build a large matrix containing the S parameters of all components and then use matrix operations to solve it. However this method needs heavy algorithms. A more elegant possibility was published in [3]. Each step computes only one connection and so unites two connected components to a single S parameter block. This procedure has to be done with every connection until there is only one block left whose S parameters therefore are the simulation result.

Connecting port k of circuit $(\underline{S})$ with port l of circuit $(\underline{T})$, the new S-parameters are

$$\underline{S}'_{ij} = \underline{S}_{ij} + \frac{\underline{S}_{kj} \cdot \underline{T}_{ll} \cdot \underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \tag{4.17}$$

with i and j both being ports of $(\underline{S})$. Furthermore, it is

$$\underline{S}'_{mj} = \frac{\underline{S}_{kj} \cdot \underline{T}_{ml}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \tag{4.18}$$

with m being a port of the circuit $(\underline{T})$. If two ports of the same circuit $(\underline{S})$ are connected, the new S-parameters are

$$\underline{S}'_{ij} = \underline{S}_{ij} + \frac{\underline{S}_{kj} \cdot \underline{S}_{il} \cdot (1 - \underline{S}_{lk}) + \underline{S}_{lj} \cdot \underline{S}_{ik} \cdot (1 - \underline{S}_{kl}) + \underline{S}_{kj} \cdot \underline{S}_{ll} \cdot \underline{S}_{ik} + \underline{S}_{lj} \cdot \underline{S}_{kk} \cdot \underline{S}_{il}}{(1 - \underline{S}_{kl}) \cdot (1 - \underline{S}_{lk}) - \underline{S}_{kk} \cdot \underline{S}_{ll}}. \tag{4.19}$$

If more than two ports are connected at a node, one have to insert one or more ideal tee components. Its S-parameters write as follows.

$$\left(\underline{S}\right) = \frac{1}{3}\cdot\begin{pmatrix} -1 & 2 & 2\\ 2 & -1 & 2\\ 2 & 2 & -1 \end{pmatrix} \tag{4.20}$$

For optimisation reasons it may be desirable to insert a cross if at least four components are connected at one node. Its S-parameters write as follows.

$$\left(\underline{S}\right) = \frac{1}{2}\cdot\begin{pmatrix} -1 & 1 & 1 & 1\\ 1 & -1 & 1 & 1\\ 1 & 1 & -1 & 1\\ 1 & 1 & 1 & -1 \end{pmatrix} \tag{4.21}$$

4.3.2 Theoretical Background

The formulas (4.17), (4.18) and (4.19) were obtained using the “nontouching-loop” rule being an analytical method for solving a flow graph. In order to derive these equations, first a few basic definitions have to be understood.

A “path” is a series of branches into the same direction with no node touched more than once. A paths value is the product of the coefficients of the branches. A “loop” is formed when a path starts and finishes at the same node. A “first-order” loop is a path coming to closure with no node passed more than once. Its value is the product of the values of all branches encountered on the route. A “second-order” loop consists of two first-order loops not touching each other at any node. Its value is calculated as the product of the values of the two first-order loops. Third- and higher-order loops are three or more first-order loops not touching each other at any node.

The nontouching-loop rule can be applied to solve any flow graph. In the following equation in symbolic form T represents the ratio of the dependent variable in question and the independent variable.

$$T = \frac{\begin{array}{r} P_1\cdot\left(1-\Sigma L_1^{(1)}+\Sigma L_2^{(1)}-\Sigma L_3^{(1)}+\ldots\right)+P_2\cdot\left(1-\Sigma L_1^{(2)}+\Sigma L_2^{(2)}-\Sigma L_3^{(2)}+\ldots\right)\\ +P_3\cdot\left(1-\Sigma L_1^{(3)}+\Sigma L_2^{(3)}-\Sigma L_3^{(3)}+\ldots\right)+P_4\cdot(1-\ldots)+\ldots \end{array}}{1-\Sigma L_1+\Sigma L_2-\Sigma L_3+\ldots} \tag{4.22}$$

In eq. (4.22) ΣL_1 stands for the sum of all first-order loops, ΣL_2 is the sum of all second-order loops, and so on. P_1, P_2, P_3 etc., stand for the values of all paths that can be found from the independent variable to the dependent variable. $\Sigma L_1^{(1)}$ denotes the sum of those first-order loops which do not touch (hence the name) the path of P_1 at any node, $\Sigma L_2^{(1)}$ denotes then the sum of those second-order loops which do not touch the path P_1 at any point, $\Sigma L_1^{(2)}$ consequently denotes the sum of those first-order loops which do not touch the path of P_2 at any point. Each path is multiplied by the factor in parentheses which involves all the loops of all orders that the path does not touch.

When connecting two different networks the signal flow graph in fig. 4.2 is used to compute the new S-parameters. With equally reference impedances on port k and port l the relations $\underline{a}_k = \underline{b}_l$ and $\underline{a}_l = \underline{b}_k$ are satisfied.

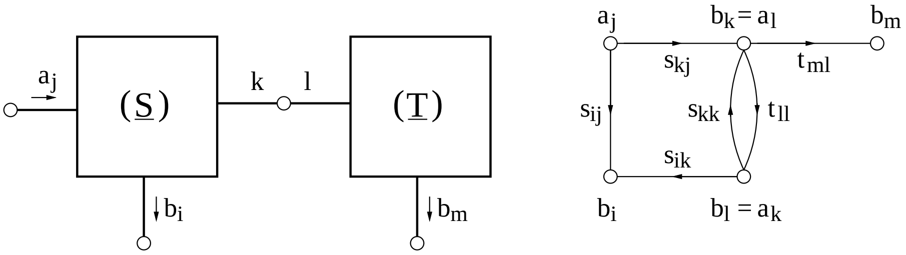


Figure 4.2: signal flow graph of a joint between ports k and l on different networks

There is only one first-order loop (see fig. 4.3) within this signal flow graph. This loops value yields to

$$L_{11} = \underline{S}_{kk} \cdot \underline{T}_{ll} \tag{4.23}$$

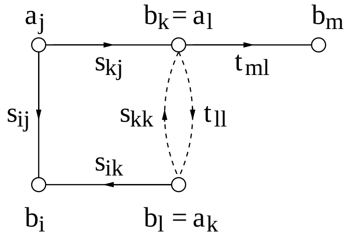


Figure 4.3: loops in the signal flow graph when connecting ports k and l on different networks

The paths that can be found from the independent variable $\underline{a}_j$ to the dependent variable $\underline{b}_i$ (as depicted in fig. 4.4) can be written as

$$P_1 = \underline{S}_{kj} \cdot \underline{T}_{ll} \cdot \underline{S}_{ik} \tag{4.24}$$

$$P_2 = \underline{S}_{ij} \tag{4.25}$$

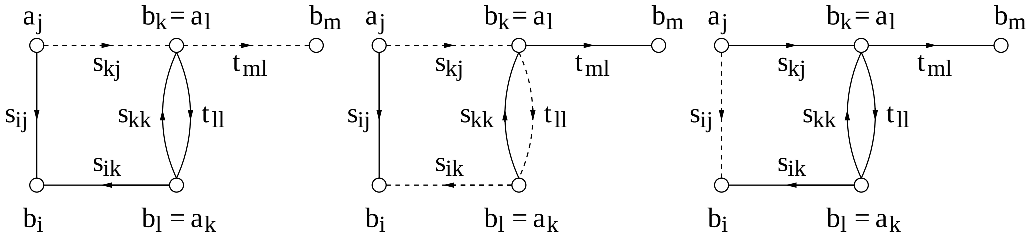


Figure 4.4: paths in the signal flow graph when connecting ports k and l on different networks

Applying the nontouching-loop rule, i.e. eq. (4.22), gives the new S-parameter $\underline{S}'_{ij}$

$$\begin{aligned} \underline{S}'_{ij} = \frac{\underline{b}_i}{\underline{a}_j} &= \frac{P_1 \cdot (1 - L_{11}) + P_2 \cdot 1}{1 - L_{11}} \\ &= \frac{\underline{S}_{ij} \cdot (1 - \underline{S}_{kk} \cdot \underline{T}_{ll}) + \underline{S}_{kj} \cdot \underline{T}_{ll} \cdot \underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} = \underline{S}_{ij} + \frac{\underline{S}_{kj} \cdot \underline{T}_{ll} \cdot \underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \end{aligned} \tag{4.26}$$

The only path that can be found from the independent variable $\underline{a}_j$ to the dependent variable $\underline{b}_m$ (as depicted in fig. 4.4) can be written as

$$P_1 = \underline{S}_{kj} \cdot \underline{T}_{ml} \tag{4.27}$$

Thus the new S-parameter $\underline{S}'_{mj}$ yields to

$$\underline{S}'_{mj} = \frac{\underline{b}_m}{\underline{a}_j} = \frac{P_1 \cdot 1}{1 - L_{11}} = \frac{\underline{S}_{kj} \cdot \underline{T}_{ml}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \tag{4.28}$$

When connecting the same network the signal flow graph in fig. 4.5 is used to compute the new S-parameters. With equally reference impedances on port k and port l the relations $\underline{a}_k = \underline{b}_l$ and $\underline{a}_l = \underline{b}_k$ are satisfied.

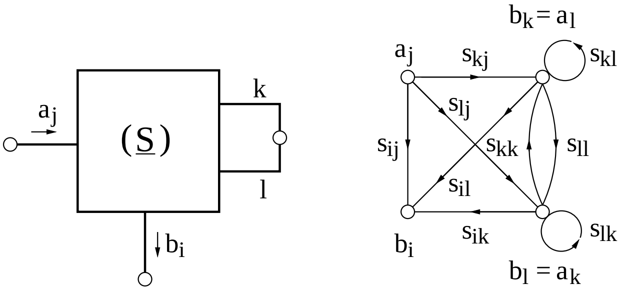


Figure 4.5: signal flow graph of a joint between ports k and l on the same network

There are three first-order loops and a second-order loop (see fig. 4.6) within this signal flow graph. These loops' values yield to

$$L_{11} = \underline{S}_{kk} \cdot \underline{S}_{ll} \tag{4.29}$$
$$L_{12} = \underline{S}_{kl} \tag{4.30}$$
$$L_{13} = \underline{S}_{lk} \tag{4.31}$$
$$L_{21} = L_{12} \cdot L_{13} = \underline{S}_{kl} \cdot \underline{S}_{lk} \tag{4.32}$$

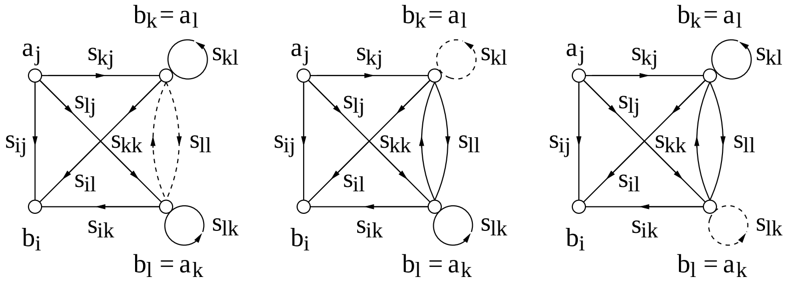


Figure 4.6: loops in the signal flow graph when connecting ports k and l on the same network

There are five different paths that can be found from the independent variable $\underline{a}_j$ to the dependent variable $\underline{b}_i$ (as depicted in fig. 4.7) which can be written as

$$P_1 = \underline{S}_{kj} \cdot \underline{S}_{ll} \cdot \underline{S}_{ik} \tag{4.33}$$

$$P_2 = \underline{S}_{kj} \cdot \underline{S}_{il} \tag{4.34}$$

$$P_3 = \underline{S}_{lj} \cdot \underline{S}_{ik} \tag{4.35}$$

$$P_4 = \underline{S}_{ij} \tag{4.36}$$

$$P_5 = \underline{S}_{lj} \cdot \underline{S}_{kk} \cdot \underline{S}_{il} \tag{4.37}$$

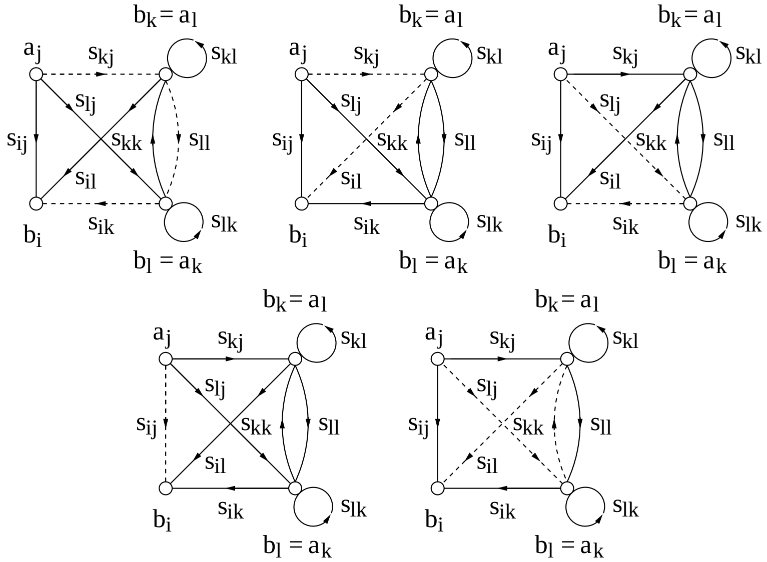


Figure 4.7: paths in the signal flow graph when connecting ports k and l on the same network

Thus the new S-parameter $\underline{S}'_{ij}$ yields to

$$\begin{aligned}
\underline{S}'_{ij} &= \frac{P_1 + P_2 \cdot (1 - L_{13}) + P_3 \cdot (1 - L_{12}) + P_4 \cdot (1 - (L_{11} + L_{12} + L_{13}) + L_{21}) + P_5}{1 - (L_{11} + L_{12} + L_{13}) + L_{21}} \\
&= P_4 + \frac{P_1 + P_2 \cdot (1 - L_{13}) + P_3 \cdot (1 - L_{12}) + P_5}{1 - (L_{11} + L_{12} + L_{13}) + L_{21}} \\
&= \underline{S}_{ij} + \frac{\underline{S}_{kj} \cdot \underline{S}_{ll} \cdot \underline{S}_{ik} + \underline{S}_{kj} \cdot \underline{S}_{il} \cdot (1 - \underline{S}_{lk}) + \underline{S}_{lj} \cdot \underline{S}_{ik} \cdot (1 - \underline{S}_{kl}) + \underline{S}_{lj} \cdot \underline{S}_{kk} \cdot \underline{S}_{il}}{1 - (\underline{S}_{kk} \cdot \underline{S}_{ll} + \underline{S}_{kl} + \underline{S}_{lk}) + \underline{S}_{kl} \cdot \underline{S}_{lk}} \\
&= \underline{S}_{ij} + \frac{\underline{S}_{kj} \cdot \underline{S}_{ll} \cdot \underline{S}_{ik} + \underline{S}_{kj} \cdot \underline{S}_{il} \cdot (1 - \underline{S}_{lk}) + \underline{S}_{lj} \cdot \underline{S}_{ik} \cdot (1 - \underline{S}_{kl}) + \underline{S}_{lj} \cdot \underline{S}_{kk} \cdot \underline{S}_{il}}{(1 - \underline{S}_{kl}) \cdot (1 - \underline{S}_{lk}) - \underline{S}_{kk} \cdot \underline{S}_{ll}}
\end{aligned} \tag{4.38}$$

This short introduction to signal flow graphs and their solution using the nontouching-loop rule verifies the initial formulas used to compute the new S-parameters for the reduced subnetworks.

4.3.3 Differential S-parameter ports

The implemented algorithm for the S-parameter analysis calculates S-parameters in terms of the ground node. In order to allow differential S-parameters as well it is necessary to insert an ideal impedance transformer with a turns ratio of 1:1 between the differential port and the device under test.

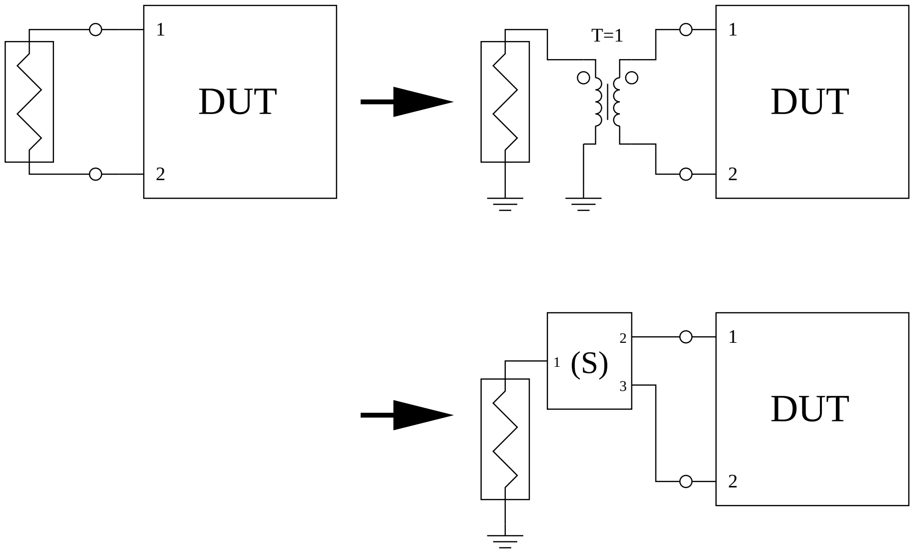


Figure 4.8: transformation of differential port into single ended port

The S-parameter matrix of the inserted ideal transformer being a three port device can be written as follows.

$$(\underline{S}) = \frac{1}{3} \cdot \begin{pmatrix} 1 & 2 & -2 \\ 2 & 1 & 2 \\ -2 & 2 & 1 \end{pmatrix} \tag{4.39}$$

This transformation can be applied to each S-parameter port in a circuit regardless whether it is actually differential or not.

It is also possible to do the impedance transformation within this step (for S-parameter ports with impedances different than 50Ω). This can be done by using a transformer with an impedance ration of

$$r = T^2 = \frac{50\Omega}{Z} \tag{4.40}$$

With Z being the S-parameter port impedance. The S-parameter matrix of the inserted ideal transformer now writes as follows.

$$(\underline{S}) = \frac{1}{2 \cdot Z_0 + Z} \cdot \begin{pmatrix} 2 \cdot Z_0 - Z & 2 \cdot \sqrt{Z_0 \cdot Z} & -2 \cdot \sqrt{Z_0 \cdot Z} \\ 2 \cdot \sqrt{Z_0 \cdot Z} & Z & 2 \cdot Z_0 \\ -2 \cdot \sqrt{Z_0 \cdot Z} & 2 \cdot Z_0 & Z \end{pmatrix} \tag{4.41}$$

With Z being the new S-parameter port impedance and Z_0 being 50Ω.

4.4 S-Parmeters via AC Analysis

In the chapter above it was shown how to make an s-parameter analysis by connecting all components step-by-step together. This is an elegant concept. Nonetheless, a more practical approach is to use an AC analysis, because this way, the s-parameters of the components need not to be computed. Instead, the full s-parameters are calculated by solving the MNA matrix and obtaining the result from the following equations:

$$\underline{S}_{mn} = \frac{\underline{U}_m - \underline{I}_m \cdot Z_m}{\underline{U}_n + \underline{I}_n \cdot Z_n} \cdot \sqrt{\frac{Z_n}{Z_m}} \tag{4.42}$$

$$= \frac{2 \cdot \underline{U}_m - \underline{U}_{0,m}}{\underline{U}_{0,n}} \cdot \sqrt{\frac{Z_n}{Z_m}} \tag{4.43}$$

for $m = n$

$$\underline{S}_{nn} = 2 \cdot \frac{\underline{U}_n}{\underline{U}_{0,n}} - 1 \tag{4.44}$$

and for $m \neq n$

$$\underline{S}_{mn} = 2 \cdot \frac{\underline{U}_m}{\underline{U}_{0,n}} \cdot \sqrt{\frac{Z_n}{Z_m}} \tag{4.45}$$

As can be seen, to get the n-th column of the s-parameter matrix port n must be stimulated whereas all other ports must be passive. Hence, for an n-port s-parameter simulation the MNA matrix must be solved n times. LU decomposition should be used, because for each of the n solutions the right-hand side changes only. By using the short-circuit current stimulation $I_{0,n} = 2\text{V}/Z_n$ for port n, its open-circuit voltage is $U_{0,n} = 2\text{V}$ and the above-mentioned formulas become:

for $m = n$

$$\underline{S}_{nn} = \underline{U}_n - 1 \tag{4.46}$$

and for $m \neq n$

$$\underline{S}_{mn} = \underline{U}_m \cdot \sqrt{\frac{Z_n}{Z_m}} \tag{4.47}$$

where U_n is the voltage at port n. These equations become more complicated, if the reference impedances $\underline{Z}_n$ are not real-valued. With the definition of power waves (equation 4.1), the s-parameters yield:

$$\underline{S}_{mn} = \frac{2 \cdot \underline{U}_m \cdot \text{Re}(\underline{Z}_m) - \underline{I}_{0,m} \cdot |\underline{Z}_m|^2}{\underline{I}_{0,n} \cdot \underline{Z}_n \cdot \underline{Z}_m} \cdot \sqrt{\frac{|\text{Re}(\underline{Z}_n)|}{|\text{Re}(\underline{Z}_m)|}} \tag{4.48}$$

4.5 Mixed-Mode S-Parmeters

In most cases the s-parameters are referenced to ground. This is shown in the chapters before and fits perfectly the most popular microwave topologies, like coaxial cables. But as topologies without ground-reference exist (like dipole antennas), there is also sense in defining s-parameters this way.

The definition of mixed-mode s-parameters [4] can be seen in figure 4.9. Each port has two terminals. This configuration exhibits two independent modes:

$$\text{common mode:} \quad U_c = \frac{1}{2} \cdot (U_1 + U_2) \quad \text{and} \quad I_c = I_1 + I_2 \tag{4.49}$$

$$\text{differential mode:} \quad U_d = U_1 - U_2 \quad \text{and} \quad I_d = \frac{1}{2} \cdot (I_1 - I_2) \tag{4.50}$$

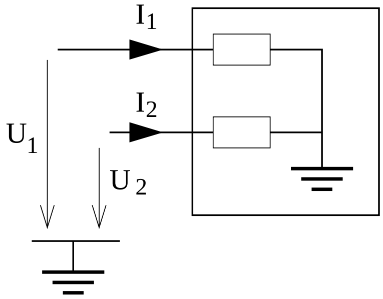


Figure 4.9: two-terminal port

Measuring the ground-referenced s-parameters of each terminal, the mixed-mode s-parameters of a two-port can be calculated as follows. (Here, terminal 1 and 2 are port 1, as well as terminal 3 and 4 are port 2.)

$$S_{dd11} = \frac{1}{2} \cdot (S_{11} - S_{21} - S_{12} + S_{22}) \tag{4.51}$$

$$S_{dd12} = \frac{1}{2} \cdot (S_{13} - S_{23} - S_{14} + S_{24}) \tag{4.52}$$

$$S_{dd21} = \frac{1}{2} \cdot (S_{31} - S_{41} - S_{32} + S_{42}) \tag{4.53}$$

$$S_{dd22} = \frac{1}{2} \cdot (S_{33} - S_{43} - S_{34} + S_{44}) \tag{4.54}$$

$$S_{dc11} = \frac{1}{2} \cdot (S_{11} - S_{21} + S_{12} - S_{22}) \tag{4.55}$$

$$S_{dc12} = \frac{1}{2} \cdot (S_{13} - S_{23} + S_{14} - S_{24}) \tag{4.56}$$

$$S_{dc21} = \frac{1}{2} \cdot (S_{31} - S_{41} + S_{32} - S_{42}) \tag{4.57}$$

$$S_{dc22} = \frac{1}{2} \cdot (S_{33} - S_{43} + S_{34} - S_{44}) \tag{4.58}$$

$$S_{cd11} = \frac{1}{2} \cdot (S_{11} + S_{21} - S_{12} - S_{22}) \tag{4.59}$$

$$S_{cd12} = \frac{1}{2} \cdot (S_{13} + S_{23} - S_{14} - S_{24}) \tag{4.60}$$

$$S_{cd21} = \frac{1}{2} \cdot (S_{31} + S_{41} - S_{32} - S_{42}) \tag{4.61}$$

$$S_{cd22} = \frac{1}{2} \cdot (S_{33} + S_{43} - S_{34} - S_{44}) \tag{4.62}$$

$$S_{cc11} = \frac{1}{2} \cdot (S_{11} + S_{21} + S_{12} + S_{22}) \tag{4.63}$$

$$S_{cc12} = \frac{1}{2} \cdot (S_{13} + S_{23} + S_{14} + S_{24}) \tag{4.64}$$

$$S_{cc21} = \frac{1}{2} \cdot (S_{31} + S_{32} + S_{41} + S_{42}) \tag{4.65}$$

$$S_{cc22} = \frac{1}{2} \cdot (S_{33} + S_{43} + S_{34} + S_{44}) \tag{4.66}$$

The concept is easily extendable to n-ports by replacing indices accordingly.

4.6 Applications

4.6.1 Stability

A very important task in microwave design (especially for amplifiers) is the question, whether the circuit tends to unwanted oscillations. A two-port oscillates if, despite of no signal being fed into it, AC power issues from at least one of its ports. This condition can be easily expressed in terms of RF quantities, so a circuit is stable if:

$$|\underline{r}_1| < 1 \qquad \text{and} \qquad |\underline{r}_2| < 1 \tag{4.67}$$

with $\underline{r}_1$ being reflexion coefficient of port 1 and $\underline{r}_2$ the one of port 2.

A further question can be asked: What conditions must be fulfilled to have a two-port be stable for all combinations of passive impedance terminations at port 1 and port 2? Such a circuit is called unconditionally stable. [5] is one of the best discussions dealing with this subject.

A circuit is unconditionally stable if the following two relations hold:

$$K = \frac{1 - |S_{11}|^2 - |S_{22}|^2 + |\Delta|^2}{2 \cdot |S_{12} \cdot S_{21}|} > 1 \tag{4.68}$$

$$|\Delta| = |S_{11} \cdot S_{22} - S_{12} \cdot S_{21}| < 1 \tag{4.69}$$

with Δ being the determinant of the S parameter matrix of the two port. K is called Rollet stability factor. Two relations must be fulfilled to have a necessary and sufficient criterion.

A more practical criterion (necessary and sufficient) for unconditional stability is obtained with the μ-factor:

$$\mu = \frac{1 - |S_{11}|^2}{|S_{22} - S_{11}^* \cdot \Delta| + |S_{12} \cdot S_{21}|} > 1 \tag{4.70}$$

Because of symmetry reasons, a second stability factor must exist that also gives a necessary and sufficient criterion for unconditional stability:

$$\mu' = \frac{1 - |S_{22}|^2}{|S_{11} - S_{22}^* \cdot \Delta| + |S_{12} \cdot S_{21}|} > 1 \tag{4.71}$$

For conditional stable two-ports it is interesting which load and which source impedance may cause instability. This can be seen using stability circles [6]. A disadvantage of this method is that the radius of the below-mentioned circles can become infinity. (A circle with infinite radius is a line.)

Within the reflexion coefficient plane of the load (r_L-plane), the stability circle is:

$$\underline{r}_{center} = \frac{S_{22}^* - S_{11} \cdot \Delta^*}{|S_{22}|^2 - |\Delta|^2} \tag{4.72}$$

$$\text{Radius} = \frac{|S_{12}| \cdot |S_{21}|}{|S_{22}|^2 - |\Delta|^2} \tag{4.73}$$

If the center of the r_L-plane lies within this circle and $|S_{11}| \leq 1$ then the circuit is stable for all reflexion coefficients inside the circle. If the center of the r_L-plane lies outside the circle and $|S_{11}| \leq 1$ then the circuit is stable for all reflexion coefficients outside the circle.

Very similar is the situation for reflexion coefficients in the source plane (r_S-plane). The stability circle is:

$$\underline{r}_{center} = \frac{S_{11}^* - S_{22} \cdot \Delta^*}{|S_{11}|^2 - |\Delta|^2} \tag{4.74}$$

$$\text{Radius} = \frac{|S_{12}| \cdot |S_{21}|}{|S_{11}|^2 - |\Delta|^2} \tag{4.75}$$

If the center of the r_S-plane lies within this circle and $|S_{22}| \leq 1$ then the circuit is stable for all reflexion coefficients inside the circle. If the center of the r_S-plane lies outside the circle and $|S_{22}| \leq 1$ then the circuit is stable for all reflexion coefficients outside the circle.

4.6.2 Gain

Maximum available and stable power gain (only for unconditional stable 2-ports) [6]:

$$G_{max} = \left| \frac{S_{21}}{S_{12}} \right| \cdot \left(K - \sqrt{K^2 - 1} \right) \tag{4.76}$$

where K is Rollet stability factor.

The (bilateral) transmission power gain (or transducer power gain) of a two-port is the ratio between the power delivered to the load and the available power of the source.

$$G_T = \frac{|S_{21}|^2 \cdot (1 - |r_S|^2) \cdot (1 - |r_L|^2)}{|1 - r_S \cdot S_{11} - r_L \cdot S_{22} + \Delta \cdot r_S \cdot r_L|^2} \tag{4.77}$$

The transducer power gain can be split into three parts [6]:

$$G_T = G_S \cdot G_0 \cdot G_L \tag{4.78}$$

with

$$G_S = \frac{(1 - |r_S|^2) \cdot (1 - |r_1|^2)}{|1 - r_S \cdot r_1|^2} \tag{4.79}$$

$$G_0 = |S_{21}|^2 \tag{4.80}$$

$$G_L = \frac{1 - |r_L|^2}{|1 - r_L \cdot S_{22}|^2 \cdot (1 - |r_1|^2)} \tag{4.81}$$

where r_1 is reflexion coefficient of the two-port input:

$$r_1 = S_{11} + \frac{S_{12} \cdot S_{21} \cdot r_L}{1 - r_L \cdot S_{22}} \tag{4.82}$$

The curves of constant gain are circles in the reflexion coefficient plane. The circle for the load-mismatched two-port with gain G_L is

$$\underline{r}_{center} = \frac{(S_{22}^* - S_{11} \cdot \Delta^*) \cdot G_L}{G_L \cdot (|S_{22}|^2 - |\Delta|^2) + 1} \tag{4.83}$$

$$\text{Radius} = \frac{\sqrt{1 - G_L \cdot (1 - |S_{11}|^2 - |S_{22}|^2 + |\Delta|^2) + G_L^2 \cdot |S_{12} \cdot S_{21}|^2}}{G_L \cdot (|S_{22}|^2 - |\Delta|^2) + 1} \tag{4.84}$$

The circle for the source-mismatched two-port with gain G_S is

$$\underline{r}_{center} = \frac{G_S \cdot r_1^*}{1 - |r_1|^2 \cdot (1 - G_S)} \tag{4.85}$$

$$\text{Radius} = \frac{\sqrt{1 - G_S} \cdot (1 - |r_1|^2)}{1 - |r_1|^2 \cdot (1 - G_S)} \tag{4.86}$$

The available power gain G_A of a two-port is reached when the source is conjugately matched to the input port and the load is conjugately matched to the output port. It is:

$$G_A = \frac{|S_{21}|^2 \cdot (1 - |r_S|^2)}{|1 - S_{11} \cdot r_S|^2 - |S_{22} - \Delta \cdot r_S|^2} \tag{4.87}$$

with $\Delta = S_{11}S_{22} - S_{12}S_{21}$. The curves with constant gain G_A are circles in the source reflexion coefficient plane (r_S-plane). The center $r_{S,c}$ and the radius R_S are:

$$r_{S,c} = \frac{g_A \cdot C_1^*}{1 + g_A \cdot (|S_{11}|^2 - |\Delta|^2)} \tag{4.88}$$

$$R_S = \frac{\sqrt{1 - 2 \cdot K \cdot g_A \cdot |S_{12}S_{21}| + g_A^2 \cdot |S_{12}S_{21}|^2}}{|1 + g_A \cdot (|S_{11}|^2 - |\Delta|^2)|} \tag{4.89}$$

with $C_1 = S_{11} - S_{22}^* \cdot \Delta$, $g_A = G_A/|S_{21}|^2$ and K Rollet stability factor.

The operating (or effective) power gain G_P of a two-port is the output power delivered to the load divided by the input power delivered to the amplifier. It is:

$$G_P = \frac{|S_{21}|^2 \cdot (1 - |r_L|^2)}{|1 - S_{22} \cdot r_L|^2 - |S_{11} - \Delta \cdot r_L|^2} \tag{4.90}$$

with $\Delta = S_{11}S_{22} - S_{12}S_{21}$. The curves with constant gain G_P are circles in the load reflexion coefficient plane (r_L-plane). The center $r_{L,c}$ and the radius R_L are:

$$r_{L,c} = \frac{g_P \cdot C_2^*}{1 + g_P \cdot (|S_{22}|^2 - |\Delta|^2)} \tag{4.91}$$

$$R_L = \frac{\sqrt{1 - 2 \cdot K \cdot g_P \cdot |S_{12}S_{21}| + g_P^2 \cdot |S_{12}S_{21}|^2}}{|1 + g_P \cdot (|S_{22}|^2 - |\Delta|^2)|} \tag{4.92}$$

with $C_2 = S_{22} - S_{11}^* \cdot \Delta$, $g_P = G_P/|S_{21}|^2$ and K Rollet stability factor.

4.6.3 Power Matching

A frequent task in microwave engineering is to match a load impedance Z_L to an internal source impedance Z_S. It aims at getting the maximum possible power out of the RF generator. This occurs if the load impedance equals the conjugate-complex of the source impedance. There are many ways to achieve this power matching. The most easy and obvious one uses a series and a parallel reactance, as depicted in figure 4.10. This combination always succeeds, if real value of source and load impedance is greater zero. But it has a small bandwidth only.

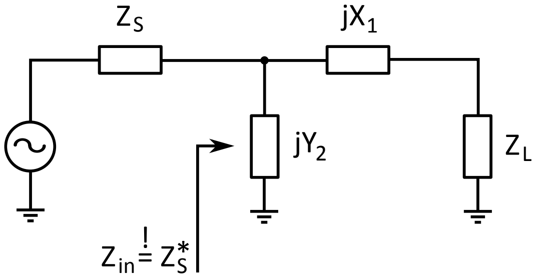


Figure 4.10: Power match with an L circuit

According to figure 4.11 the reactance X_1 and the susceptance Y_2 are calculated as follows:

$$r^2 = \left(\frac{0.5}{G_S}\right)^2 = X^2 + \left(R_L - \frac{0.5}{G_S}\right)^2 \tag{4.93}$$

$$\Rightarrow X^2 = \frac{0.25}{G_S^2} - R_L^2 + \frac{R_L}{G_S} - \frac{0.25}{G_S^2} \tag{4.94}$$

$$\Rightarrow X = \pm\sqrt{R_L \cdot \left(\frac{1}{G_S} - R_L\right)} \tag{4.95}$$

with

$$\frac{1}{G_S} = \frac{1}{\mathrm{Re}\left(\frac{1}{R_S+jX_S}\right)} = \frac{1}{\mathrm{Re}\left(\frac{R_S-jX_S}{R_S^2+X_S^2}\right)} = \frac{R_S^2+X_S^2}{R_S} \tag{4.96}$$

this yields

$$X_1 = X - X_L = \pm\sqrt{R_L \cdot \left(\frac{R_S^2+X_S^2}{R_S} - R_L\right)} - X_L \tag{4.97}$$

Finally, the remaining susceptance must be compensated, i.e.:

$$Y_2 = -\mathrm{Im}\left(\frac{1}{R_L+jX}\right) - \mathrm{Im}\left(\frac{1}{R_S+jX_S}\right) = \frac{X}{R_L^2+X^2} + \frac{X_S}{R_S^2+X_S^2} \tag{4.98}$$

This method is possible if $R_L \leq 1/G_S$. If this isn't the case, source and load impedance has to be exchanged, i.e. the series and parallel elements X_1 and Y_2 have to be mirrored.

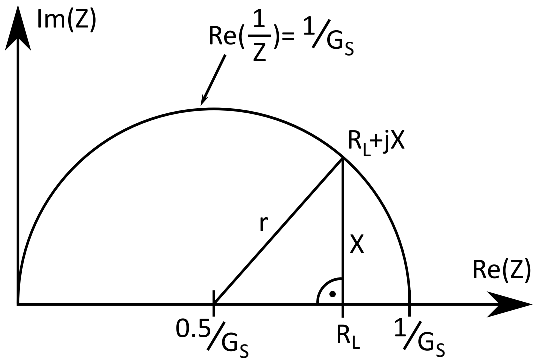


Figure 4.11: Impedance plane showing the matching path

4.6.4 Two-Port Matching

Obtaining concurrent power matching of input and output in a bilateral circuit is not such simple, due to the backward transmission S_{12}. However, in linear circuits, this task can be easily solved by the following equations:

$$\Delta = S_{11} \cdot S_{22} - S_{12} \cdot S_{21} \tag{4.99}$$

$$B = 1 + |S_{11}|^2 - |S_{22}|^2 - |\Delta|^2 \tag{4.100}$$

$$C = S_{11} - S_{22}^* \cdot \Delta \tag{4.101}$$

$$r_S = \frac{1}{2 \cdot C} \cdot \left(B - \sqrt{B^2 - |2 \cdot C|^2}\right) \tag{4.102}$$

Here r_S is the reflexion coefficient that the circuit needs to see at the input port in order to reach concurrently matched in- and output. For the reflexion coefficient at the output r_L the same equations hold by simply changing the indices (exchange 1 by 2 and vice versa).

4.6.5 De-embeding

During measurements it often happens that some parts of the device under test (DUT) needs to be separated from the rest of the result. For example: An amplifier should be measured, but because of its high output signal, an attenuator must be inserted (embeded) to prevent damaging the s-parameter system. So the measurement results have to be re-calculated afterwards in order to remove the influence of the attenuator. This procedure is called de-embeding and is done by measuring the s-paramters S_{embed} of the attenuator alone, take them to calculate a new s-parameter set $S_{deembed}$ and finally use a CAE program to connect the blocks together:
$(S_{deembed} \quad \rightarrow \quad [S_{embed} \quad \rightarrow \quad S_{DUT}]) \quad \equiv \quad S_{DUT}$
The s-parameters $S_{deembed}$ are computed from the original s-parameters S_{embed} as follows:

$$S_{deembed} = \frac{1}{\Delta S} \cdot \begin{pmatrix} S_{11} & -S_{21} \\ -S_{12} & S_{22} \end{pmatrix} \tag{4.103}$$

where ΔS is the determinante of the s-parameter matrix.

Chapter 5

Noise Waves

5.1 Definition

In microwave circuits described by scattering parameters, it is advantageous to regard noise as noise waves [7]. The noise characteristics of an n-port is then defined completely by one outgoing noise wave $\underline{b}_{noise,n}$ at each port (see 2-port example in fig. 5.1) and the correlation between these noise sources. Therefore, mathematically, you can characterize a noisy n-port by its $n \times n$ scattering matrix $(\underline{S})$ and its $n \times n$ noise wave correlation matrix $(\underline{C})$.

$$
\begin{split}
(\underline{C}) &= \begin{pmatrix}
\overline{\underline{b}_{noise,1} \cdot \underline{b}_{noise,1}^*} & \overline{\underline{b}_{noise,1} \cdot \underline{b}_{noise,2}^*} & \cdots & \overline{\underline{b}_{noise,1} \cdot \underline{b}_{noise,n}^*} \\
\overline{\underline{b}_{noise,2} \cdot \underline{b}_{noise,1}^*} & \overline{\underline{b}_{noise,2} \cdot \underline{b}_{noise,2}^*} & \cdots & \overline{\underline{b}_{noise,2} \cdot \underline{b}_{noise,n}^*} \\
\vdots & \vdots & \ddots & \vdots \\
\overline{\underline{b}_{noise,n} \cdot \underline{b}_{noise,1}^*} & \overline{\underline{b}_{noise,n} \cdot \underline{b}_{noise,2}^*} & \cdots & \overline{\underline{b}_{noise,n} \cdot \underline{b}_{noise,n}^*}
\end{pmatrix} \\
&= \begin{pmatrix}
\underline{c}_{11} & \underline{c}_{12} & \cdots & \underline{c}_{1n} \\
\underline{c}_{21} & \underline{c}_{22} & \cdots & \underline{c}_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\underline{c}_{n1} & \underline{c}_{n2} & \cdots & \underline{c}_{nn}
\end{pmatrix}
\end{split}
\tag{5.1}
$$

Where $\overline{x}$ is the time average of x and $\underline{x}^*$ is the conjugate complex of $\underline{x}$. Noise correlation matrices are hermitian matrices because the following equations hold.

$$\text{Im}\left(\underline{c}_{nn}\right) = \text{Im}\left(\overline{|b_{noise,n}|^2}\right) = 0 \tag{5.2}$$

$$\underline{c}_{nm} = \underline{c}_{mn}^* \tag{5.3}$$

Where Im($\underline{x}$) is the imaginary part of $\underline{x}$ and $|\underline{x}|$ is the magnitude of $\underline{x}$.

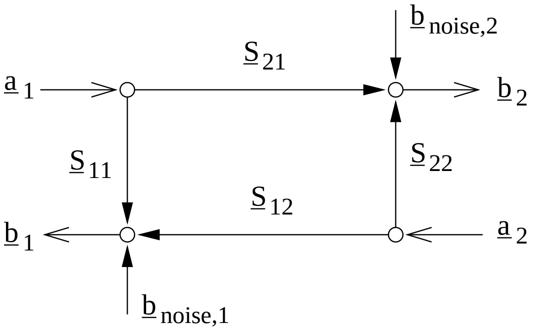


Figure 5.1: signal flow graph of a noisy 2-port

5.2 Noise Parameters

Having the noise wave correlation matrix, one can easily compute the noise parameters [7]. The following equations calculate them with regard to port 1 (input) and port 2 (output). (If one uses an n-port and want to calculate the noise parameters regarding to other ports, one has to replace the index numbers of S- and c-parameters accordingly. I.e. replace "1" with the number of the input port and "2" with the number of the output port.)

Noise figure:

$$F = \frac{SNR_{in}}{SNR_{out}} = 1 + \frac{T_e}{T_0} = 1 + \frac{\underline{c}_{22}}{k \cdot T_0 \cdot |\underline{S}_{21}|^2} \tag{5.4}$$

$$NF\,[\text{dB}] = 10 \cdot \lg F \tag{5.5}$$

with T_e being the equivalent noise temperature of the input port.
Optimal source reflection coefficient (normalized according to the input port impedance):

$$\Gamma_{opt} = \eta_2 \cdot \left(1 - \sqrt{1 - \frac{1}{|\eta_2|^2}}\right) \tag{5.6}$$

With

$$\eta_1 = \underline{c}_{11} \cdot |\underline{S}_{21}|^2 - 2 \cdot \text{Re}\,(\underline{c}_{12} \cdot \underline{S}_{21} \cdot \underline{S}_{11}^*) + \underline{c}_{22} \cdot |\underline{S}_{11}|^2 \tag{5.7}$$

$$\eta_2 = \frac{1}{2} \cdot \frac{\underline{c}_{22} + \eta_1}{\underline{c}_{22} \cdot \underline{S}_{11} - \underline{c}_{12} \cdot \underline{S}_{21}} \tag{5.8}$$

Minimum noise figure:

$$F_{min} = 1 + \frac{\underline{c}_{22} - \eta_1 \cdot |\Gamma_{opt}|^2}{k \cdot T_0 \cdot |\underline{S}_{21}|^2 \cdot (1 + |\Gamma_{opt}|^2)} \tag{5.9}$$

$$NF_{min} = 10 \cdot \lg F_{min} \tag{5.10}$$

Equivalent noise resistance:

$$R_n = \frac{Z_{port,in}}{4 \cdot k \cdot T_0} \cdot \left(\underline{c}_{11} - 2 \cdot \text{Re}\left(\underline{c}_{12} \cdot \left(\frac{1 + \underline{S}_{11}}{\underline{S}_{21}}\right)^*\right) + \underline{c}_{22} \cdot \left|\frac{1 + \underline{S}_{11}}{\underline{S}_{21}}\right|^2\right) \tag{5.11}$$

With $Z_{port,in}$ internal impedance of input port
Boltzmann constant $k = 1.380658 \cdot 10^{-23}$ J/K
standard temperature $T_0 = 290$K

Calculating the noise wave correlation coefficients from the noise parameters is straightforward as well.

$$\underline{c}_{11} = k \cdot T_{min} \cdot \left(|S_{11}|^2 - 1\right) + K_x \cdot |1 - S_{11} \cdot \Gamma_{opt}|^2 \tag{5.12}$$

$$\underline{c}_{22} = |S_{21}|^2 \cdot \left(k \cdot T_{min} + K_x \cdot |\Gamma_{opt}|^2\right) \tag{5.13}$$

$$\underline{c}_{12} = \underline{c}_{21}^* = -S_{21}^* \cdot \Gamma_{opt}^* \cdot K_x + \frac{S_{11}}{S_{21}} \cdot \underline{c}_{22} \tag{5.14}$$

with

$$K_x = \frac{4 \cdot k \cdot T_0 \cdot R_n}{Z_0 \cdot |1 + \Gamma_{opt}|^2} \tag{5.15}$$

$$T_{min} = T_0 \cdot (F_{min} - 1) \tag{5.16}$$

Once having the noise parameters, one can calculate the noise figure for every source admittance $Y_S = G_S + j \cdot B_s$, source impedance $Z_S = R_S + j \cdot X_s$, or source reflection coefficient r_S.

$$\begin{aligned}
F = \frac{SNR_{in}}{SNR_{out}} &= \frac{T_{equi}}{T_0} + 1 && (5.17)\\
&= F_{min} + \frac{G_n}{R_S} \cdot \left((R_S - R_{opt})^2 + (X_S - X_{opt})^2\right) && (5.18)\\
&= F_{min} + \frac{G_n}{R_S} \cdot \left|\underline{Z}_S - \underline{Z}_{opt}\right|^2 && (5.19)\\
&= F_{min} + \frac{R_n}{G_S} \cdot \left((G_S - G_{opt})^2 + (B_S - B_{opt})^2\right) && (5.20)\\
&= F_{min} + \frac{R_n}{G_S} \cdot \left|\underline{Y}_S - \underline{Y}_{opt}\right|^2 && (5.21)\\
&= F_{min} + 4 \cdot \frac{R_n}{Z_0} \cdot \frac{\left|\underline{\Gamma}_{opt} - \underline{r}_S\right|^2}{\left(1 - |\underline{r}_S|^2\right) \cdot \left|1 + \underline{\Gamma}_{opt}\right|^2} && (5.22)
\end{aligned}$$

Where SNR_{in} and SNR_{out} are the signal to noise ratios at the input and output, respectively, T_{equi} is the equivalent (input) noise temperature. Note that G_n does not equal $1/R_n$.

All curves with constant noise figures are circles (in all planes, i.e. impedance, admittance and reflection coefficient). A circle in the reflection coefficient plane has the following parameters.

center point:

$$\underline{r}_{center} = \frac{\Gamma_{opt}}{1 + N} \tag{5.23}$$

radius:

$$R = \frac{\sqrt{N^2 + N \cdot \left(1 - |\Gamma_{opt}|^2\right)}}{1 + N} \tag{5.24}$$

with

$$N = \frac{Z_0}{4 \cdot R_n} \cdot (F - F_{min}) \cdot |1 + \Gamma_{opt}|^2 \tag{5.25}$$

5.3 Noise Wave Correlation Matrix in CAE

Due to the similar concept of S parameters and noise correlation coefficients, the CAE noise analysis can be performed quite alike the S parameter analysis (section 4.3.1). As each step uses the S parameters to calculate the noise correlation matrix, the noise analysis is best done step by step in parallel with the S parameter analysis. Performing each step is as follows: We have the noise wave correlation matrices ($(\underline{C})$, $(\underline{D})$) and the S parameter matrices ($(\underline{S})$, $(\underline{T})$) of two arbitrary circuits and want to know the correlation matrix of the special circuit resulting from connecting two circuits at one port.

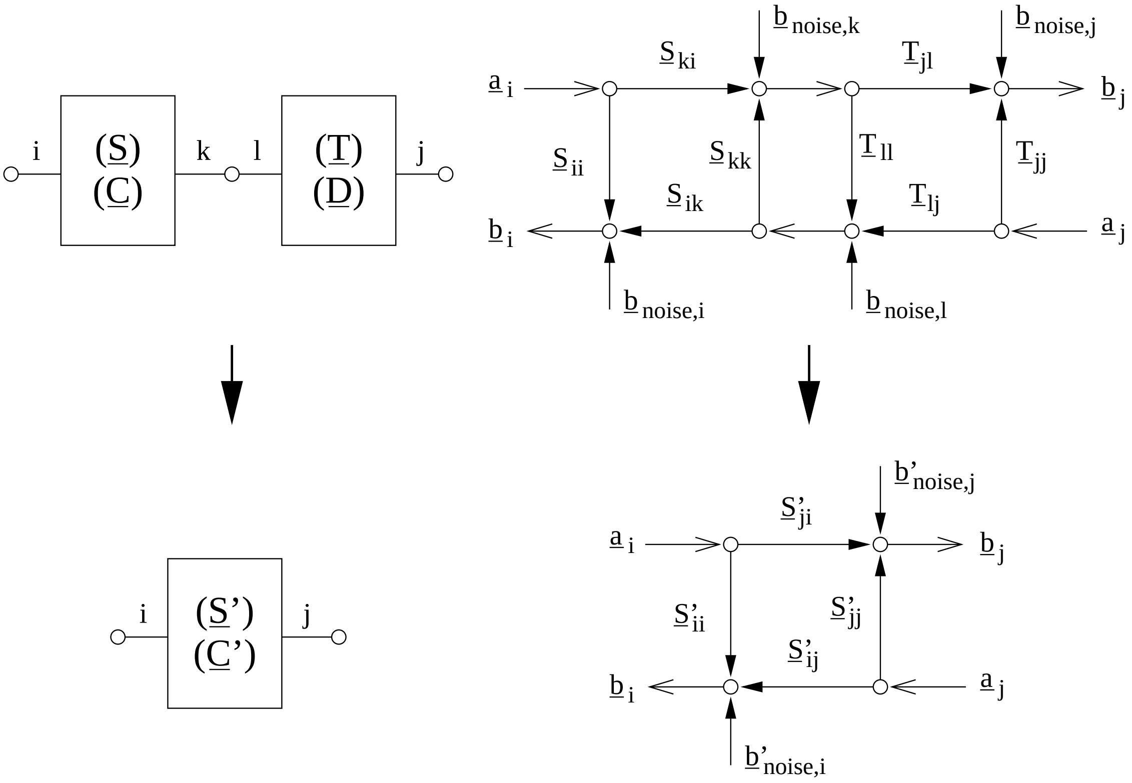


Figure 5.2: connecting two noisy circuits, scheme (left) and signal flow graph (right)

An example is shown in fig. 5.2. What we have to do is to transform the inner noise waves $\underline{b}_{noise,k}$ and $\underline{b}_{noise,l}$ to the open ports. Let us look upon the example. According to the signal flow graph the resulting noise wave $\underline{b}'_{noise,i}$ writes as follows:

$$\underline{b}'_{noise,i} = \underline{b}_{noise,i} + \underline{b}_{noise,k} \cdot \frac{\underline{T}_{ll} \cdot \underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} + \underline{b}_{noise,l} \cdot \frac{\underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \tag{5.26}$$

The noise wave $\underline{b}_{noise,j}$ does not contribute to $\underline{b}'_{noise,i}$, because no path leads to port i. Calculating $\underline{b}'_{noise,j}$ is quite alike:

$$\underline{b}'_{noise,j} = \underline{b}_{noise,j} + \underline{b}_{noise,l} \cdot \frac{\underline{T}_{jl} \cdot \underline{S}_{kk}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} + \underline{b}_{noise,k} \cdot \frac{\underline{T}_{jl}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \tag{5.27}$$

Now we can derive the first element of the new noise correlation matrix by multiplying eq. (5.26) with eq. (5.27).

$$\begin{aligned}
\underline{c}'_{ij} = & \overline{\underline{b}'_{noise,i} \cdot \underline{b}'^*_{noise,j}} \\
= & \overline{\underline{b}_{noise,i} \cdot \underline{b}^*_{noise,j}} \\
& + \overline{\underline{b}_{noise,i} \cdot \underline{b}^*_{noise,l}} \cdot \left(\frac{\underline{T}_{jl} \cdot \underline{S}_{kk}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \right)^* + \overline{\underline{b}_{noise,i} \cdot \underline{b}^*_{noise,k}} \cdot \left(\frac{\underline{T}_{jl}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \right)^* \\
& + \overline{\underline{b}_{noise,k} \cdot \underline{b}^*_{noise,j}} \cdot \frac{\underline{T}_{ll} \cdot \underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \\
& + \overline{\underline{b}_{noise,k} \cdot \underline{b}^*_{noise,l}} \cdot \frac{\underline{T}_{ll} \cdot \underline{S}_{ik} \cdot \underline{T}^*_{jl} \cdot \underline{S}^*_{kk}}{|1 - \underline{S}_{kk} \cdot \underline{T}_{ll}|^2} + \overline{\underline{b}_{noise,k} \cdot \underline{b}^*_{noise,k}} \cdot \frac{\underline{T}_{ll} \cdot \underline{S}_{ik} \cdot \underline{T}^*_{jl}}{|1 - \underline{S}_{kk} \cdot \underline{T}_{ll}|^2} \\
& + \overline{\underline{b}_{noise,l} \cdot \underline{b}^*_{noise,j}} \cdot \frac{\underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \\
& + \overline{\underline{b}_{noise,l} \cdot \underline{b}^*_{noise,l}} \cdot \frac{\underline{S}_{ik} \cdot \underline{T}^*_{jl} \cdot \underline{S}^*_{kk}}{|1 - \underline{S}_{kk} \cdot \underline{T}_{ll}|^2} + \overline{\underline{b}_{noise,l} \cdot \underline{b}^*_{noise,k}} \cdot \frac{\underline{S}_{ik} \cdot \underline{T}^*_{jl}}{|1 - \underline{S}_{kk} \cdot \underline{T}_{ll}|^2}
\end{aligned} \tag{5.28}$$

The noise waves of different circuits are uncorrelated and therefore their time average product equals zero (e.g. $\overline{\underline{b}_{noise,i} \cdot \underline{b}^*_{noise,j}} = 0$). Thus, the final result is:

$$\begin{aligned}
\underline{c}'_{ij} = (\underline{c}'_{ji})^* = & (\underline{c}_{kk} \cdot \underline{T}_{ll} + \underline{d}_{ll} \cdot \underline{S}^*_{kk}) \cdot \frac{\underline{S}_{ik} \cdot \underline{T}^*_{jl}}{|1 - \underline{S}_{kk} \cdot \underline{T}_{ll}|^2} \\
& + \underline{c}_{ik} \cdot \left(\frac{\underline{T}_{jl}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \right)^* + \underline{d}_{lj} \cdot \frac{\underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}}
\end{aligned} \tag{5.29}$$

All other cases of connecting circuits can be calculated the same way using the signal flow graph. The results are listed below.

If index i and j are within the same circuit, it results in fig. 5.3. The following formula holds:

$$\begin{aligned}
\underline{c}'_{ij} = (\underline{c}'_{ji})^* = \underline{c}_{ij} & + (\underline{c}_{kk} \cdot |\underline{T}_{ll}|^2 + \underline{d}_{ll}) \cdot \frac{\underline{S}_{ik} \cdot \underline{S}^*_{jk}}{|1 - \underline{S}_{kk} \cdot \underline{T}_{ll}|^2} \\
& + \underline{c}_{ik} \cdot \left(\frac{\underline{T}_{ll} \cdot \underline{S}_{jk}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}} \right)^* + \underline{c}_{kj} \cdot \frac{\underline{T}_{ll} \cdot \underline{S}_{ik}}{1 - \underline{S}_{kk} \cdot \underline{T}_{ll}}
\end{aligned} \tag{5.30}$$

This equation is also valid, if i equals j.

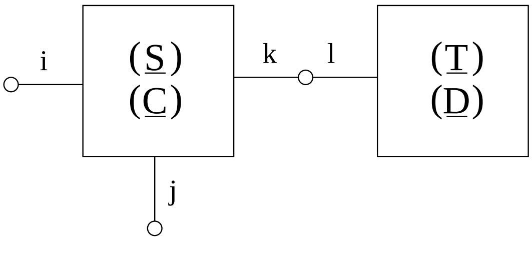


Figure 5.3: connecting two noisy circuits

If the connected ports k and l are from the same circuit, the following equations must be applied (see also fig. 5.4) to obtain the new correlation matrix coefficients.

$$M = (1 - \underline{S}_{kl}) \cdot (1 - \underline{S}_{lk}) - \underline{S}_{kk} \cdot \underline{S}_{ll} \tag{5.31}$$

$$K_1 = \frac{\underline{S}_{il} \cdot (1 - \underline{S}_{lk}) + \underline{S}_{ll} \cdot \underline{S}_{ik}}{M} \tag{5.32}$$

$$K_2 = \frac{\underline{S}_{ik} \cdot (1 - \underline{S}_{kl}) + \underline{S}_{kk} \cdot \underline{S}_{il}}{M} \tag{5.33}$$

$$K_3 = \frac{\underline{S}_{jl} \cdot (1 - \underline{S}_{lk}) + \underline{S}_{ll} \cdot \underline{S}_{jk}}{M} \tag{5.34}$$

$$K_4 = \frac{\underline{S}_{jk} \cdot (1 - \underline{S}_{kl}) + \underline{S}_{kk} \cdot \underline{S}_{jl}}{M} \tag{5.35}$$

$$\begin{aligned} \underline{c}'_{ij} = \underline{c}_{ij} + \underline{c}_{kj} \cdot K_1 + \underline{c}_{lj} \cdot K_2 + K_3^* \cdot (\underline{c}_{ik} + \underline{c}_{kk} \cdot K_1 + \underline{c}_{lk} \cdot K_2) + \\ K_4^* \cdot (\underline{c}_{il} + \underline{c}_{kl} \cdot K_1 + \underline{c}_{ll} \cdot K_2) \end{aligned} \tag{5.36}$$

These equations are also valid if i equals j.

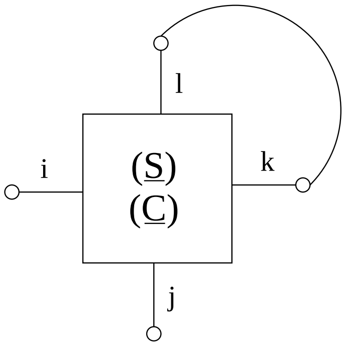


Figure 5.4: connection within a noisy circuits

The absolute values of the noise correlation coefficients are very small. To achieve a higher numerical precision, it is recommended to normalize the noise matrix with $k \cdot T_0$. After the simulation they do not have to be denormalized, because the noise parameters can be calculated by using equation (5.4) to (5.11) and omitting all occurrences of $k \cdot T_0$.

The transformer concept to deal with different port impedances and with differential ports (as described in section 4.3.3) can also be applied to this noise analysis.

5.4 Noise Parameters via AC Analysis

Above it was shown how to compute the noise wave correlation matrix of a circuit and how to calculate the noise parameters with this matrix. This is a nice method, but it is more practical to use the AC noise analysis. The noise of the s-parameter ports must be switched off during this simulation. Then, the noise parameters are obtained by the noise voltage $u_{n,2}$ at the output port.

$$F = \frac{SNR_{in}}{SNR_{out}} = 1 + \frac{u_{n,2}^2 \cdot \mathrm{Re}(R_S)}{4 \cdot k_B \cdot T_0 \cdot |Z_{21}|^2} \tag{5.37}$$

where R_S is the internal resistance of the input port and Z_{21} is the transimpedance between port 1 and port 2. The latter one was already calculated for getting S_{21}.

In order to get the complete set of noise parameters, the noise voltages at the input port $u_{n,1}$ and output port $u_{n,2}$ as well as its correlation $u_{n,12}$ must be calculated (see chapter 3.2). Because the port terminations are set noiseless during this simulation, the noise wave correlation coefficients can then easy computed:

$$\underline{c}_{11} = \overline{\underline{b}_{noise,1} \cdot \underline{b}^*_{noise,1}} = \left(\frac{\underline{u}_{n,1} - Z_1 \cdot \underline{i}_{n,1}}{2 \cdot \sqrt{Z_1}}\right)^2 = \frac{\underline{u}^2_{n,1}}{Z_1} \tag{5.38}$$

$$\underline{c}_{22} = \overline{\underline{b}_{noise,2} \cdot \underline{b}^*_{noise,2}} = \left(\frac{\underline{u}_{n,2} - Z_2 \cdot \underline{i}_{n,2}}{2 \cdot \sqrt{Z_2}}\right)^2 = \frac{\underline{u}^2_{n,2}}{Z_2} \tag{5.39}$$

$$\underline{c}_{12} = \overline{\underline{b}_{noise,1} \cdot \underline{b}^*_{noise,2}} = \frac{\underline{u}^2_{n,12}}{\sqrt{Z_1 \cdot Z_2}} \tag{5.40}$$

where Z_1 and Z_2 are the port impedances of input and output port, respectively. Having the above-mentioned noise waves, all noise parameters can be calculated with the equations 5.4 to 5.11 on page 43.

5.5 Noise Correlation Matrix Transformations

The noise wave correlation matrix of a passive linear circuit generating thermal noise can simply be calculated using Bosma's theorem. The noise wave correlation matrices of active devices can be determined by forming the noise current correlation matrix and then transforming it to the equivalent noise wave correlation matrix.

The noise current correlation matrix (also called the admittance representation) $\underline{C}_Y$ is an $n \times n$ matrix.

$$\underline{C}_Y = \begin{pmatrix} \overline{\underline{i}_1 \cdot \underline{i}_1^*} & \overline{\underline{i}_1 \cdot \underline{i}_2^*} & \cdots & \overline{\underline{i}_1 \cdot \underline{i}_n^*} \\ \overline{\underline{i}_2 \cdot \underline{i}_1^*} & \overline{\underline{i}_2 \cdot \underline{i}_2^*} & \cdots & \overline{\underline{i}_2 \cdot \underline{i}_n^*} \\ \vdots & \vdots & \ddots & \vdots \\ \overline{\underline{i}_n \cdot \underline{i}_1^*} & \overline{\underline{i}_n \cdot \underline{i}_2^*} & \cdots & \overline{\underline{i}_n \cdot \underline{i}_n^*} \end{pmatrix} = \begin{pmatrix} \underline{c}_{11} & \underline{c}_{12} & \cdots & \underline{c}_{1n} \\ \underline{c}_{21} & \underline{c}_{22} & \cdots & \underline{c}_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \underline{c}_{n1} & \underline{c}_{n2} & \cdots & \underline{c}_{nn} \end{pmatrix} \tag{5.41}$$

This definition is very likely the one made by eq. (5.1). The matrix has the same properties as well. Because in most transistor models the noise behaviour is expressed as the sum of effects of noise current sources it is easier to form this matrix representation.

5.5.1 Transformations

There are three usable noise correlation matrix representations for multiport circuits.

- admittance representation $\underline{C}_Y$ - based on noise currents
- impedance representation $\underline{C}_Z$ - based on noise voltages
- wave representation $\underline{C}_S$ - based on noise waves

According to Scott W. Wedge and David B. Rutledge [7] the transformations between these representations write as follows.

	$\underline{C}_Y$	$\underline{C}_Z$	$\underline{C}_S$
$\underline{C}_Y$	$\underline{C}_Y$	$Y \cdot \underline{C}_Z \cdot Y^+$	$(E+Y) \cdot \underline{C}_S \cdot (E+Y)^+$
$\underline{C}_Z$	$Z \cdot \underline{C}_Y \cdot Z^+$	$\underline{C}_Z$	$(E+Z) \cdot \underline{C}_S \cdot (E+Z)^+$
$\underline{C}_S$	$\frac{1}{4}(E+S) \cdot \underline{C}_Y \cdot (E+S)^+$	$\frac{1}{4}(E-S) \cdot \underline{C}_Z \cdot (E-S)^+$	$\underline{C}_S$

The signal as well as correlation matrices in impedance and admittance representations are assumed to be normalized in the above table. E denotes the identity matrix and the $^+$ operator indicates the transposed conjugate matrix (also called adjoint or adjugate).

Each noise correlation matrix transformation requires the appropriate signal matrix representation which can be obtained using the formulas given in section 19.1 on page 242.

For 2-ports there is another important noise correlation matrix [8], the so-called chain representation. It is defined by two noise sources at the input (port 1), a series voltage source and a parallel current source:

$$\underline{C}_A = \begin{pmatrix} \overline{\underline{u}_1 \cdot \underline{u}_1^*} & \overline{\underline{u}_1 \cdot \underline{i}_1^*} \\ \overline{\underline{i}_1 \cdot \underline{u}_1^*} & \overline{\underline{i}_1 \cdot \underline{i}_1^*} \end{pmatrix} = \begin{pmatrix} R_n & 0.5 \cdot (F_{min} - 1) - R_n \cdot \underline{Y}_{opt}^* \\ 0.5 \cdot (F_{min} - 1) - R_n \cdot \underline{Y}_{opt} & R_n \cdot |\underline{Y}_{opt}|^2 \end{pmatrix} \tag{5.42}$$

Transformations into impedance or admittance representation is done in the usual way:

	$\underline{C}_Y$	$\underline{C}_Z$	$\underline{C}_A$
$\underline{C}_Y$	$\underline{C}_Y$	$Y \cdot \underline{C}_Z \cdot Y^+$	$\begin{pmatrix} -y_{11} & 1 \\ -y_{21} & 0 \end{pmatrix} \cdot \underline{C}_A \cdot \begin{pmatrix} -y_{11} & 1 \\ -y_{21} & 0 \end{pmatrix}^+$
$\underline{C}_Z$	$Z \cdot \underline{C}_Y \cdot Z^+$	$\underline{C}_Z$	$\begin{pmatrix} 1 & -z_{11} \\ 0 & -z_{21} \end{pmatrix} \cdot \underline{C}_A \cdot \begin{pmatrix} 1 & -z_{11} \\ 0 & -z_{21} \end{pmatrix}^+$
$\underline{C}_A$	$\begin{pmatrix} 0 & a_{12} \\ 1 & a_{22} \end{pmatrix} \cdot \underline{C}_Y \cdot \begin{pmatrix} 0 & a_{12} \\ 1 & a_{22} \end{pmatrix}^+$	$\begin{pmatrix} 1 & -a_{11} \\ 0 & -a_{21} \end{pmatrix} \cdot \underline{C}_Z \cdot \begin{pmatrix} 1 & -a_{11} \\ 0 & -a_{21} \end{pmatrix}^+$	$\underline{C}_A$

5.6 Noise Wave Correlation Matrix of Components

Many components do not produce any noise. Every element of their noise correlation matrix therefore equals exactly zero. Examples are lossless, passive components, i.e. capacitors, inductors, transformers, circulators, phase shifters. Furthermore ideal voltage and current sources (without internal resistance) as well as gyrators also do not produce any noise.

If one wants to calculate the noise wave correlation matrix of a component, the most universal method is to take noise voltages and noise currents and then derive the noise waves by the use of equation (4.1). However, this can be very difficult.

A passive, linear circuit produces only thermal noise and thus its noise waves can be calculated with Bosma's theorem (assuming thermodynamic equilibrium).

$$(\underline{C}) = k \cdot T \cdot \left((\underline{E}) - (\underline{S}) \cdot (\underline{S})^{*T} \right) \tag{5.43}$$

with $(\underline{S})$ being the S parameter matrix and $(\underline{E})$ identity matrix. Of course, this theorem can also be written with impedance and admittance representation of the noise correlation matrix:

$$\underline{C}_Z = 4 \cdot k \cdot T \cdot \mathrm{Re}(\underline{Z}) \tag{5.44}$$

$$\underline{C}_Y = 4 \cdot k \cdot T \cdot \mathrm{Re}(\underline{Y}) \tag{5.45}$$

Chapter 6

Transient Analysis

The transient simulation is a non-linear time-domain analysis of a network with arbitrary excitations (e.g. rectangular sources). Thus, it's like a DC analysis with additionally current sources stemming from loading and unloading of energy storing components (inductors and capacitors).

In the following sections the transient analysis is explained by implementing a linear capacitor. The equations for inductors and non-linear components are described later. The relations between current and voltage of a capacitance is given by the following differential equation:

$$I(t) = \frac{dQ}{dt} = C \cdot \frac{dV}{dt} \tag{6.1}$$

This immediately shows what happens during a transient simulation: When the voltage V across a capacitor changes over time t, a current I is flowing into it or out of it. This also changes the charge Q whose value needs to be tracked by the simulation engine. That's why it's more practical to transform the above equation into an integral equation:

$$Q(t) = Q(t = 0) + \int_0^t I(\tau) \cdot d\tau \tag{6.2}$$

So now the task will be to solve this formula numerically. The result is the current I through the capacitor which is then added to the node currents in order to calculate the network response at a specific time.

6.1 Integration methods

During the transient analysis the node voltages of a circuit are computed at discrete time spots t_n. Thus the notation of quantities is simplified by using indices, e.g. $Q(t_n) = Q_n$. So finally the charge at t_{n+1} is calculated by its change from the previous time spot t_n:

$$Q_{n+1} = Q_n + \int_{t_n}^{t_{n+1}} I(\tau) \cdot d\tau \tag{6.3}$$

The time step h usually varies from step to step:

$$h_n = t_{n+1} - t_n \tag{6.4}$$

There are many different methods for performing numerical integration. The differential equations in transient simulations often contains very different time constants and as a consequence they are very stiff (i.e. its eigenvalues strongly vary). Therefore, finding a stable and universal algorithm turns out to be difficult.

This section describes linear multi-step (LMS) integration methods. This kind of formula is used in all traditional circuit simulators. The following list contains the most important properties:

- single- and multi-step methods
  Single step methods only use Q_n and/or I_n in order to calculate Q_{n+1}, multi step methods use Q_i and/or I_i with $0 \leq i < n$ as well. A p-step method is expressed by the following formula:
  $$Q_{n+1} = \sum_{i=0}^{p} a_i \cdot Q_{n-i} + h_n \sum_{i=-1}^{p} b_i \cdot I_{n-i} \tag{6.5}$$
  So it can completely be defined by its coefficients a_i and b_i.

- order
  The order k of an integration method is the maximum power of the polynom that can approximate the solution with the same accuracy. I.e. the local truncation error ε_{LTE} can be approximated by:
  $$\varepsilon_{LTE} = E \cdot h_n^{k+1} \cdot \frac{d^{k+1} V_n}{dt^{k+1}} + O\left(h_n^{k+2}\right) \tag{6.6}$$
  with E being a constant error factor and $O(\ldots)$ being a higher order term that can be neglected. High order methods can handle larger time steps which leads to faster simulation runs. But low order methods are more stable.

- implicit and explicit methods
  Explicit integration formulas don't contain I_{n+1}, so they predict the result without solving an equation system (at least for linear circuits). Hence, they are faster than implicit methods, but not suited for circuit simulations, because of not being stiff stable.

Linear multi-step methods are limited by the two Dahlquist barriers:

- First Dahlquist Stability Barrier
  The order of accuracy p of a zero-stable k-step method satisfies
  $$p \leq \begin{cases} k+2 & k \quad \text{is even} \\ k+1 & k \quad \text{is odd} \\ k & \text{is explicit} \end{cases} \tag{6.7}$$

- Second Dahlquist Stability Barrier
  An explicit method cannot be A-stable. The order of accuracy of an implicit A-stable method satisfies $p \leq 2$.

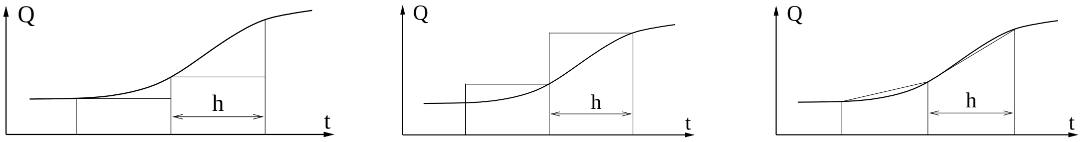
Indeed many other methods exist [9], [10] that don't exhibits these limitations:

- Multi-stage methods (so-called Runge-Kutta) like Gauss, RADAU or SDIRK exhibit very good stability properties. But they use intermediate steps and therefore create high computational costs.

- General linear methods (GLM) like DIMSIM tries to combine the advantages of LMS and Runge-Kutta methods, but still need much computation power.

- Multi-derivative methods like the Obreshkov or Rosenbrock formulae are further algorithms with good stability, but again need higher computation costs.

- Composite, cyclic composite or blended algorithms that uses combinations of different LMS methods are able to exhibit better stability properties, but the improvements in accuracy are usually limited.



explicit euler implicit euler trapezoidal

6.1.1 Explicit Euler Method

The explicit Euler (or forward Euler) algorithm is a first order single-step method. It's defined by the following formula:

$$Q_{n+1} = Q_n + h_n \cdot I_n \tag{6.8}$$

This method is conditional stable, and thus not usable for circuit simulations.

6.1.2 Implicit Euler Method

The implicit Euler (or backward Euler) algorithm is a first order single-step method. It's defined by the following formula:

$$Q_{n+1} = Q_n + h_n \cdot I_{n+1} \quad \Rightarrow \quad I_{n+1} = \frac{C}{h_n} \cdot (V_{n+1} - V_n) \tag{6.9}$$

This method is A-stable and L-stable. But because of being single-step, its accuracy is quite low.

6.1.3 Trapezoidal Method

The trapezoidal (or bilinear) algorithm is a second order single-step method. It approximates $Q(t)$ linearly between two time spots, which yields the following formula:

$$Q_{n+1} = Q_n + \frac{h_n}{2} \cdot (I_{n+1} + I_n) \quad \Rightarrow \quad I_{n+1} = \frac{2C}{h_n} V_{n+1} - \frac{2C}{h_n} V_n - I_n \tag{6.10}$$

The local truncation error ε_{LTE} writes:

$$\begin{aligned} \varepsilon_{LTE} &= \frac{1}{12} \cdot h_n^3 \cdot \left| \frac{\partial^3 V_n}{\partial t^3} \right| = \frac{h_n^3}{12} \cdot \left| \frac{1}{C} \cdot \frac{\partial^2 I_n}{\partial t^2} \right| \\ &\approx \frac{h_n^3}{12 \cdot C} \cdot \left| \frac{\frac{I_{n+1} - I_n}{h_n} - \frac{I_n - I_{n-1}}{h_{n-1}}}{h_n} \right| \\ &= \frac{h_n}{12 \cdot C} \cdot \left| (I_{n+1} - I_n) - \frac{h_n}{h_{n-1}} \cdot (I_n - I_{n-1}) \right| \end{aligned} \tag{6.11}$$

With an error constant of $E = 1/12$ this method is the most accurate 2nd order LMS method and thus also the most accurate A-stable LMS method. These properties make the trapezoidal rule to one of the most used integration methods in circuit simulation. Unfortunately, it conserves energy and therefore oscillations and numerical errors are not damped. Non-physical ringing may be the result especially after switching events, i.e. this method isn't L-stable.

6.1.4 Theta-Method

The theta-method is a modified trapezoidal algorithm defined by the following formula:

$$Q_{n+1} = Q_n + h_n \cdot (\theta \cdot I_{n+1} + (1 - \theta) \cdot I_n) \quad \Rightarrow \quad I_{n+1} = C \cdot \frac{V_{n+1} - V_n}{\theta \cdot h_n} - \frac{1 - \theta}{\theta} I_n \tag{6.12}$$

As can be seen this yields explicit Euler for $\theta = 0$, it yields trapezoidal for $\theta = 0.5$ and implicit Euler for $\theta = 1$. The local truncation errors gives:

$$\begin{aligned}\varepsilon_{LTE} &= \left(\theta - \frac{1}{2}\right) \cdot h_n^2 \cdot \frac{\partial^2 V_n}{\partial t^2} + \left(\frac{\theta}{2} - \frac{1}{3}\right) \cdot h_n^3 \cdot \frac{\partial^3 V_n}{\partial t^3} + O(h_n^4) \\ &\approx \left(\theta - \frac{1}{2}\right) \cdot \frac{h_n}{C} \cdot (I_{n+1} - I_n) + \left(\frac{\theta}{2} - \frac{1}{3}\right) \cdot \frac{h_n}{C} \cdot \left(I_{n+1} - I_n - \frac{h_n}{h_{n-1}} \cdot (I_n - I_{n-1})\right)\end{aligned} \tag{6.13}$$

Obviously, this method is explicit if $\theta = 0$ and implicit otherwise. Furthermore, it's second order if $\theta = 0.5$ and first order otherwise. It's A-stable for $\theta \geq 0.5$. Usually, $\theta \approx 0.54$ is chosen. This sacrifices a small piece of accuracy in order to improve stability, i.e. to avoid the non-physical ringing that may occur with the original trapezoidal method.

6.1.5 Implicit Gear

The Gear [11] formulae (also called BDF - backward differentiation formulae) are implicit multistep integration methods that are stable for an order $k \leq 6$.

$$Q_{n+1} = \sum_{i=0}^{k-1} a_i \cdot Q_{n-i} + h_n \cdot b_{-1} \cdot I_{n+1} \quad \Rightarrow \quad I_{n+1} = \frac{C}{b_{-1} \cdot h_n} \cdot \left(V_{n+1} - \sum_{i=0}^{k-1} a_i \cdot V_{n-i}\right) \tag{6.14}$$

The order $k = 1$ yields the implicit Euler method. The Gear formulas for constant step size h are as follows:

$$\text{BDF1:} \quad I_{n+1} = \frac{C}{h} \cdot (V_{n+1} - V_n) \tag{6.15}$$

$$\text{BDF2:} \quad I_{n+1} = \frac{C}{h} \cdot \left(\frac{3}{2} \cdot V_{n+1} - 2 \cdot V_n + \frac{1}{2} \cdot V_{n-1}\right) \tag{6.16}$$

$$\text{BDF3:} \quad I_{n+1} = \frac{C}{h} \cdot \left(\frac{11}{6} \cdot V_{n+1} - 3 \cdot V_n + \frac{3}{2} \cdot V_{n-1} - \frac{1}{3} \cdot V_{n-2}\right) \tag{6.17}$$

$$\text{BDF4:} \quad I_{n+1} = \frac{C}{h} \cdot \left(\frac{25}{12} \cdot V_{n+1} - 4 \cdot V_n + 3 \cdot V_{n-1} - \frac{4}{3} \cdot V_{n-2} + \frac{1}{4} \cdot V_{n-3}\right) \tag{6.18}$$

$$\text{BDF5:} \quad I_{n+1} = \frac{C}{h} \cdot \left(\frac{137}{60} \cdot V_{n+1} - 5 \cdot V_n + 5 \cdot V_{n-1} - \frac{10}{3} \cdot V_{n-2} + \frac{5}{4} \cdot V_{n-3} - \frac{1}{5} \cdot V_{n-4}\right) \tag{6.19}$$

$$\text{BDF6:} \quad I_{n+1} = \frac{C}{h} \cdot \left(\frac{147}{60} \cdot V_{n+1} - 6 \cdot V_n + \frac{15}{2} \cdot V_{n-1} - \frac{20}{3} \cdot V_{n-2} + \frac{15}{4} \cdot V_{n-3} - \frac{6}{5} \cdot V_{n-4} + \frac{1}{6} \cdot V_{n-5}\right) \tag{6.20}$$

The integration formula contains the coefficients $c_i = a_i / b_{-1} / h_n$ and $d_{-1} = -1 / b_{-1} / h_n$. Furthermore the step size h_n may change from step to step. To obtain the complete coefficients the following equation system has to be solved:

$$(A_{ij}) \cdot \begin{bmatrix} d_{-1} \\ c_0 \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} = \begin{bmatrix} 0 \\ -1/h_n \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad \text{with} \quad (A_{ij}) = \left(\frac{h_n + h_{n-1} + \ldots + h_{n-j+2}}{h_n}\right)^{i-1} \tag{6.21}$$

So the coefficients for 2^{nd} order Gear yield:

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & \frac{h_n + h_{n-1}}{h_n} \\ 0 & 1 & \frac{(h_n + h_{n-1})^2}{h_n^2} \end{bmatrix} \cdot \begin{bmatrix} d_{-1} \\ c_0 \\ c_1 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{-1}{h_n} \\ 0 \end{bmatrix} \tag{6.22}$$

$$\Rightarrow \quad c_1 = \frac{h_n}{h_{n-1} \cdot (h_n + h_{n-1})}, \quad c_0 = \frac{-1}{h_n} + \frac{-1}{h_{n-1}}, \quad d_{-1} = -c_0 - c_1 \tag{6.23}$$

The error constants (E in equation (6.6)) are listed in the following table:

order k	1	2	3	4	5	6
error constant E	$-\frac{1}{2}$	$-\frac{2}{9}$	$-\frac{3}{22}$	$-\frac{12}{125}$	$-\frac{10}{137}$	$-\frac{20}{343}$

This method is A-stable and L-stable for order $k \leq 2$. It's $A(\alpha)$-stable for order $k > 2$. These properties makes the Gear rules to one of the most used integration method in circuit simulations. However, these algorithms fail for circuits with highly oscillatory behaviour (i.e. with eigenvalues near the imaginary axis), because they damp oscillations ($k \leq 2$) or become unstable ($k > 2$). Therefore, the simulator must reduce the step-size or the order if instabilities are detected [12].

A modern circuit analysis often contains very complex component models (e.g. HICUM or BSIM) that consumes most of the simulation time. So higher order integration methods are very attractive for increasing the necessary time steps and decreasing the duration of the simulation. As the Gear formulae are the most stable methods for order $k > 2$, many circuit simulators use them successful for large circuits.

6.1.6 TR-BDF2

This is a one-step, 2nd order method, a composite algorithm that uses a trapezoidal integration step (TR) followed by a 2nd order Gear step (BDF2). The time step h is divided into two steps by the factor γ. I.e. first a complete iteration loop is performed with the trapezoidal formula from t_n to $t_n + \gamma h_n$:

$$Q_{n+\gamma} = Q_n + \frac{\gamma \cdot h_n}{2} \cdot (I_{n+1} + I_n) \quad \Rightarrow \quad I_{n+1} = \frac{2}{\gamma \cdot h_n} \cdot (Q_{n+\gamma} + Q_n) \tag{6.24}$$

After it converged, a complete iteration loop is performed with the second-order Gear formula from $t_n + \gamma h_n$ to t_{n+1}:

$$Q_{n+1} = \frac{1}{\gamma \cdot (2-\gamma)} \cdot Q_{n+\gamma} - \frac{(1-\gamma)^2}{\gamma \cdot (2-\gamma)} \cdot Q_n + h_n \cdot \frac{1-\gamma}{2-\gamma} \cdot I_{n+1} \tag{6.25}$$

$$\Rightarrow \quad I_{n+1} = \frac{2-\gamma}{h_n \cdot (1-\gamma)} \cdot Q_{n+1} - \frac{1}{h_n \cdot \gamma \cdot (1-\gamma)} \cdot Q_{n+\gamma} + \frac{1-\gamma}{h_n \cdot \gamma} \cdot Q_n \tag{6.26}$$

The fixed intermediate step makes this algorithm a 2-stage method. The local truncation error ε_{LTE} yields:

$$\varepsilon_{LTE} = 2 \cdot E \cdot \frac{h_n}{C} \cdot \left(\frac{I_n}{\gamma} - \frac{I_{n+\gamma}}{\gamma \cdot (1-\gamma)} + \frac{I_{n+1}}{1-\gamma} \right) \tag{6.27}$$

with the error constant

$$E = \frac{-3\gamma^2 + 4\gamma - 2}{12 \cdot (2-\gamma)} \tag{6.28}$$

It reaches its minimum with $\gamma = 2 - \sqrt{2} \approx 0.585786$. For this value the method is A-stable and L-stable, and with $E \approx -2/(24.7 \cdot (1-\gamma)) \approx -0.19$ a little more accurate than the BDF2 method ($E_{BDF2} \approx -0.22$). It was developed to combine the advantages of trapezoidal and BDF2 method, i.e. to get an L-stable algorithm (like BDF2) that doesn't damp circuit oscillations (like TR). A disadvantage is the fixed intermediate time step, which makes problems with an adaptive step-size control.

A generalization of this approach is the composition of theta- and BDF2-method [13] which also gives a 2nd order A- and L-stable formula, but with one degree of freedom.

6.1.7 Adams-Bashford

The Adams-Bashford algorithm is an explicit multi-step integration method with the following equation:

$$Q_{n+1} = a_0 \cdot Q_n + h \sum_{i=0}^{k-1} b_i \cdot I_{n-i} \tag{6.29}$$

The Adams-Bashford formula of order 1 yields the explicit Euler integration method. The coefficients of the 4th order Adams-Bashford formula can be calculated by the following equation system:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & -2 & -4 & -6 \\ 0 & 0 & 3 & 12 & 27 \\ 0 & 0 & -4 & -32 & -108 \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ b_0 \\ b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \tag{6.30}$$

The error constants (E in equation (6.6)) are listed in the following table:

order k	1	2	3	4	5	6
error constant E	$\frac{1}{2}$	$\frac{5}{12}$	$\frac{3}{8}$	$\frac{251}{720}$	$\frac{95}{288}$	$\frac{19087}{60480}$

This method is conditional stable.

6.1.8 Adams-Moulton

The Adams-Moulton algorithm is an implicit multi-step integration method with the following equation:

$$Q_{n+1} = a_0 \cdot Q_n + h \sum_{i=-1}^{k-1} b_i \cdot I_{n-i} \quad \Rightarrow \quad I_{n+1} = \frac{C}{b_{-1} \cdot h_n} V_{n+1} - \frac{a_0 \cdot C}{b_{-1} \cdot h_n} V_n - \sum_{i=0}^{k-1} \frac{b_i}{b_{-1}} \cdot I_{n-i} \tag{6.31}$$

The Adams-Moulton formula of order 1 yields the implicit Euler integration method and the formula of order 2 yields the trapezoidal rule. The coefficients of the 4th order Adams-Moulton formula can be calculated by the following equation system:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 2 & 0 & -2 & -4 \\ 0 & 3 & 0 & 3 & 12 \\ 0 & 4 & 0 & -4 & -32 \end{bmatrix} \cdot \begin{bmatrix} a_0 \\ b_{-1} \\ b_0 \\ b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \tag{6.32}$$

The error constants (E in equation (6.6)) are listed in the following table:

order k	1	2	3	4	5	6
error constant E	$-\frac{1}{2}$	$-\frac{1}{12}$	$-\frac{1}{24}$	$-\frac{19}{720}$	$-\frac{3}{160}$	$-\frac{863}{60480}$

This method is A-stable and L-stable for order $k = 1$. It's A-stable for order $k = 2$ and zero-stable for order $k > 2$.

6.1.9 Predictor-Corrector Methods

Explicit integration methods are fast because they don't need to iterate, but they become unstable for stiff problems. Implicit methods are stable but require iteration. The predictor-corrector method tries to combine the advantages of both algorithms. It uses an explicit method to create an initial guess (the predictor step). Then this result is used as starting point for the iteration with an implicit method (the corrector steps).

Let's show a typical example. The 3rd order Adams-Bashford formula creates the initial guess $Q_{n+1,0}$ by performing the following calculation once:

$$Q_{n+1,0} = Q_n + h \cdot \frac{23}{12} \cdot I_n - h \cdot \frac{16}{12} \cdot I_{n-1} + h \cdot \frac{5}{12} \cdot I_{n-2} \tag{6.33}$$

Obviously this equation don't need an iteration, because there's no quantity with index $n + 1$. If the index m is now introduced to label the number of the current iteration step, then the 3rd order Adams-Moulton formula yields:

$$Q_{n+1,m+1} = Q_n + h \cdot \frac{5}{12} \cdot Q_{n+1,m} + h \cdot \frac{8}{12} \cdot Q_n - h \cdot \frac{1}{12} \cdot Q_{n-1} \tag{6.34}$$

An iteration process is needed to find the solution of this equation, because it contains the result of the last iteration step (namely $Q_{n+1,m}$). But as it's an implicit method, there is a good chance for reaching convergence.

The local truncation error of the predictor and of the corrector method both follow equation (6.6). If the two methods have the same order k, it can be approximated by Milne's estimate:

$$\varepsilon_{LTE} \approx \frac{E_C}{E_P - E_C} \cdot (Q_{n+1,m} - Q_{n+1,0}) \tag{6.35}$$

with E_P being the error constant of the predictor and E_C being the error constant of the corrector.

6.2 Initialization

When starting a transient analysis a few things need to be noticed.

- Multi-step integration methods use several previous values for calculating the next voltage. These values don't exist at start-up (the first time step) and can't simply be set to zero. One solution is to use an integration method with lower order until the memory for previous steps has filled up. For example start with first order Gear and go up to fourth order Gear after the first seven steps are done.
- Sometimes it is wanted to give an energy-storage component an initial value before start-up. This is performed by setting the last memory to the corresponding value.

6.3 A simple example

The circuit in figure 6.1 should serve as an example, i.e. the charging of the capacitor should be simulated. At the beginning ($t = 0$) the voltage across the capacitor is $V_c(t = 0) = 0.5$V. The voltage after 1ms will be calculated, i.e. the time step is $\Delta t = 1$ms.

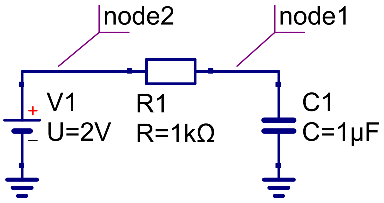


Figure 6.1: circuit for testing transient analysis

First, the MNA matrix has to be built. The three components enter their elements as follows.

- The resistor R_1 is connected between node 1 and node 2. Thus, it enters $1/R_1$ at position (1,1) and (2,2) as well as $-1/R_1$ at position (1,2) and (2,1).
- The voltage source V_1 is connected at node 1 to ground. It needs a third row and column. It enters a 1 at position (2,3) and (3,2) in order to transform a voltage source into a current source (gyrator). Furthermore, it enters V_1 at position 3 of the right-hand side vector.
- The capacitor C_1 is connected at node 1 to ground. The trapezoidal method will be used, thus the component enters $2 \cdot C_1/\Delta t$ at position (1,1). Furthermore, it enters $2 \cdot C_1/\Delta t \cdot V_c(t = 0) + I_c(t = 0)$ at position 1 of the right-hand side vector. Note that $I_c(t = 0)$ is zero because the voltage V_1 is applied at $t = 0$.

Finally, the complete matrix equation writes as follows.

$$\begin{bmatrix} \frac{1}{R_1} + \frac{2 \cdot C_1}{\Delta t} & \frac{-1}{R_1} & 0 \\ \frac{-1}{R_1} & \frac{1}{R_1} & 1 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{V1} \end{bmatrix} = \begin{bmatrix} \frac{2 \cdot C_1}{\Delta t} \cdot V_c(t=0) + I_c(t=0) \\ 0 \\ V_1 \end{bmatrix} \tag{6.36}$$

Filling in the numbers results in:

$$\begin{bmatrix} 0.003 & -0.001 & 0 \\ -0.001 & 0.001 & 1 \\ 0 & 1 & 0 \end{bmatrix}^{-1} \cdot \begin{bmatrix} 0.001 \\ 0 \\ 2 \end{bmatrix} = \begin{bmatrix} V_1 = V_c(t = 1\text{ms}) \\ V_2 \\ I_{V1} \end{bmatrix} = \begin{bmatrix} 1 \\ 2 \\ -0.001 \end{bmatrix} \tag{6.37}$$

Now, the result will be compared to the exact solution that gives:

$$V_c(t = 1\text{ms}) = (V_1 - V_c(t = 0)) \cdot \left(1 - \exp\left(\frac{-\Delta t}{R_1 \cdot C_1}\right)\right) + V_c(t = 0) = 1.448\text{V} \tag{6.38}$$

The difference is 0.448V (almost 50%). The reason is the fact that a time step of 1ms is much to large. A time step of 1μs is needed for an error of less than 1mV.

6.4 Local Truncation Error

Each numerical integration step has a limited accuracy because of limited order and finite step size h_n. The difference between the exact solution and the calculated solution is the local truncation error ε_{LTE}. Using equation (6.5) and setting $a_{-1} = -1$ the truncation error is defined as

$$\begin{aligned} \varepsilon_{LTE} &= Q_{n+1} - Q_{n+1,exact} \\ &= \sum_{i=0}^{p} a_i \cdot Q_{n-i,exact} + h \sum_{i=-1}^{p} b_i \cdot I_{n-i,exact} - Q_{n+1,exact} \\ &= \sum_{i=-1}^{p} a_i \cdot Q_{n-i,exact} + h \sum_{i=-1}^{p} b_i \cdot I_{n-i,exact} \end{aligned} \tag{6.39}$$

With the Taylor series expansions

$$Q_{n+i} = Q_n + \frac{(ih)}{1!}\dot{Q}_n + \frac{(ih)^2}{2!}\ddot{Q}_n + \ldots \tag{6.40}$$

$$I_{n+i} = \dot{Q}_{n+i} = \dot{Q}_n + \frac{(ih)}{1!}\ddot{Q}_n + \frac{(ih)^2}{2!}\dddot{Q}_n + \ldots \tag{6.41}$$

the local truncation error as defined by equation (6.39) can be written as

$$\varepsilon_{LTE} = E_0 \cdot Q_n + E_1 h \cdot \dot{Q}_n + E_2 h^2 \cdot \ddot{Q}_n + \ldots \tag{6.42}$$

The error terms E_0, E_1, E_2 etc. in their general form can then be expressed by the following equation.

$$E_q = -\frac{1}{q!} \cdot \sum_{i=-1}^{p-1} a_i \cdot (p-i)^q - \frac{1}{(q-1)!}\sum_{i=-1}^{p-1} b_i \cdot (p-i)^{q-1} \tag{6.43}$$

A linear multi-step integration method is of order k if

$$\varepsilon_{LTE} = E_{k+1} \cdot h^{k+1} \cdot \frac{d^{k+1}}{dt^{k+1}} Q_n + O\left(h^{k+2}\right) \tag{6.44}$$

The error constant E_{k+1} of an p-step integration method of order k is then defined as

$$E_{k+1} = -\frac{1}{(k+1)!} \cdot \sum_{i=-1}^{p-1} a_i \cdot (p-i)^{k+1} - \frac{1}{k!}\sum_{i=-1}^{p-1} b_i \cdot (p-i)^k \tag{6.45}$$

The practical computation of these error constants is now going to be explained using the Adams-Moulton formula of order 3 given by equation (6.34). For this method with $a_{-1} = -1$, $a_0 = 1$, $b_{-1} = 5/12$, $b_0 = 8/12$ and $b_1 = -1/12$ the following values are obtained using equation (6.43).

$$E_0 = -\frac{1}{0!} \cdot \left(-1 \cdot 2^0 + 1 \cdot 1^0\right) = 0 \tag{6.46}$$

$$E_1 = -\frac{1}{1!} \cdot \left(-1 \cdot 2^1 + 1 \cdot 1^1\right) - \frac{1}{0!} \cdot \left(\frac{5}{12}2^0 + \frac{8}{12}1^0 - \frac{1}{12}0^0\right) = 0 \tag{6.47}$$

$$E_2 = -\frac{1}{2!} \cdot \left(-1 \cdot 2^2 + 1 \cdot 1^2\right) - \frac{1}{1!} \cdot \left(\frac{5}{12}2^1 + \frac{8}{12}1^1 - \frac{1}{12}0^1\right) = 0 \tag{6.48}$$

$$E_3 = -\frac{1}{3!} \cdot \left(-1 \cdot 2^3 + 1 \cdot 1^3\right) - \frac{1}{2!} \cdot \left(\frac{5}{12}2^2 + \frac{8}{12}1^2 - \frac{1}{12}0^2\right) = 0 \tag{6.49}$$

$$E_4 = -\frac{1}{4!} \cdot \left(-1 \cdot 2^4 + 1 \cdot 1^4\right) - \frac{1}{3!} \cdot \left(\frac{5}{12}2^3 + \frac{8}{12}1^3 - \frac{1}{12}0^3\right) = -\frac{1}{24} \tag{6.50}$$

In similar ways it can be verified for each of the discussed linear multi-step integration methods that

$$E_p = 0 \quad \forall \quad 0 \le p \le k \tag{6.51}$$

6.5 Adaptive step-size control

For all numerical integration methods the choice of a proper step-size is essential. If the step size is too large, the results become inaccurate or even unstable. If the step size is too small the calculation requires more time than necessary without improving the accuracy. So the step-size h must be chosen such that the local truncation errors stays below a user-defined absolute and relative limit:

$$\varepsilon_{LTE} < \varepsilon_{abs} + \varepsilon_{rel} \cdot |Q_{n+1}| \tag{6.52}$$

Forming a step-error quotient

$$r = \frac{\varepsilon_{LTE}}{\varepsilon_{abs} + \varepsilon_{rel} \cdot |Q_{n+1}|} \tag{6.53}$$

yields the following algorithm for the step-size control. The initial step size h_0 is chosen sufficiently small. After each integration step the error quotient r of every energy-storage component gets computed in order to find the largest one r_{max}. If $r_{max} > 1$, then a reduction of the current step-size is necessary. I.e. the current result has to be rejected and needs to be calculated again with new step-size:

$$h_{n,new} = \left(\frac{0.8}{r_{max}}\right)^{\frac{1}{k+1}} \cdot h_{n,old} \tag{6.54}$$

with k denoting the order of the integration method. If necessary the process must be repeated. On the other hand if $r_{max} < 1$, then the calculated value in the current step gets accepted and the new step-size can be enlarged as follows:

$$h_{n+1} = \left(\frac{0.8}{r_{max}}\right)^{\frac{1}{k+1}} \cdot h_n \tag{6.55}$$

6.6 Stability

Rounding errors are unavoidable when evaluating the equations of integration methods. The evaluation must be repeated very often and thus the rounding error possibly accumulates. This is especially critical for stiff problems as encountered in circuit simulations, because small variations in the starting values easily lead to large variations in the solution values. If the numerical errors don't get damped by the integration formula, they will grow beyond all bounds. Therefrom the numerical methods used for the transient analysis are required to be stiffly stable and accurate as well. The regions requirements in the complex plane are visualized in the following figure.

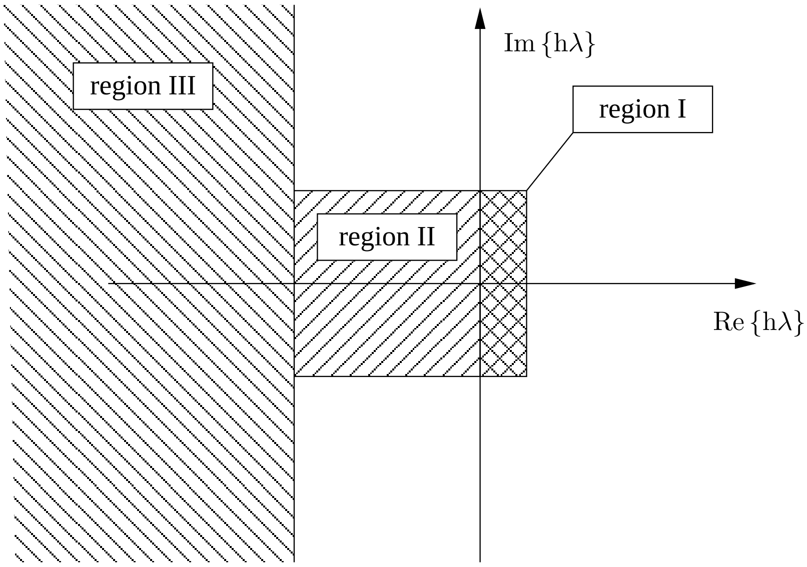


Figure 6.2: stability requirements for stiff differential equation systems

For values of $h\lambda$ in region II the numerical method must be stable and accurate, in region I accurate and in region III only stable. The area outside the specified regions are of no particular interest.

For the stability prediction of integration algorithms with regard to non-linear differential equations and equation systems the simple and linear test differential equation

$$\dot{x} = \lambda x \quad \text{with} \quad \lambda \in \mathbb{C}, \mathrm{Re}\left\{\lambda\right\} < 0, x \geq 0 \tag{6.56}$$

is used. The condition $\mathrm{Re}\left\{\lambda\right\} < 0$ ensures the solution to be decreasing. The general purpose method of resolution given in (6.5) can be solved by the polynomial method setting

$$x^k = z^k \quad \text{with} \quad z \in \mathbb{C} \tag{6.57}$$

Thus we get the characteristic polynom

$$\begin{aligned} \varphi\left(z\right) &= \varrho\left(z\right) + h\lambda \cdot \eta\left(z\right) = 0 && (6.58) \\ &= \sum_{i=-1}^{n-1} a_i \cdot z^{n-i} + h\lambda \sum_{i=-1}^{n-1} b_i \cdot z^{n-i} && (6.59) \end{aligned}$$

Because of the conditions for multi-step integration methods the above equation (6.58) can only be true for

$$|z| < 1 \tag{6.60}$$

which describes the inner unity circle on the complex plane. In order to compute the boundary of the area of absolute stability it is necessary to calculate

$$\mu\left(z\right) = h\lambda = -\frac{\varrho\left(z\right)}{\eta\left(z\right)} \quad \text{with} \quad z = e^{j\vartheta}, 0 \leq \vartheta \leq 2\pi \tag{6.61}$$

These equations describe closed loops. The inner of these loops describes the area of stability. Because $\lambda \leq 0$ and $h \geq 0$ only the left half of the complex plane is of particular interest. An integration algorithm is call zero-stable if the stability area encloses $\mu = 0$. Given this condition the algorithm is as a matter of principle usable, otherwise not. If an algorithms stability area encloses the whole left half plane it is called A-stable ("absolute" stable), i.e. the algorithm is stable for any h and all $\lambda < 0$. Any other kind of stability area introduces certain restrictions on the eigenvalues.

The figures 6.3, 6.4 and 6.5 show the stability for the discussed integration methods by visualizing the evaluation of equation (6.61). All implicit formulae are zero-stable and thus principally usable. However, circuit simulations cover a wide field of involved equations. As a consequence an algorithm needs a wide stability area in order be universally successful.

According to the second Dahlquist barrier the only LMS methods that are A-stable have an order $k \leq 2$. That's why the modified trapezoidal and the TR-BDF2 method are the most stable and reliable choice.

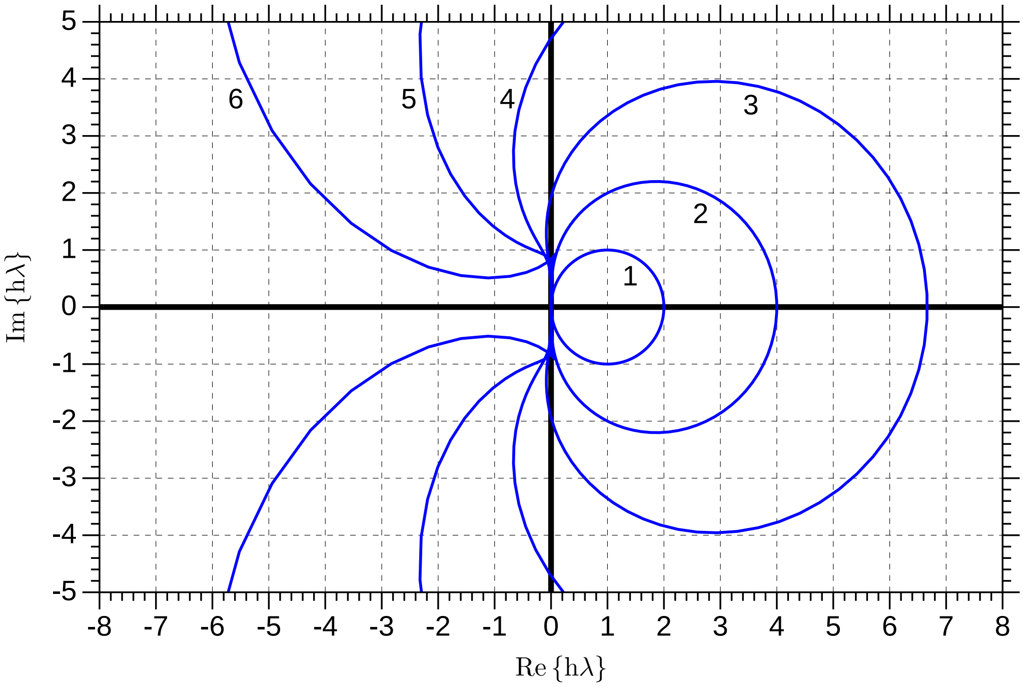


Figure 6.3: areas of absolute stability for order 1. . . 6 Gear formulae

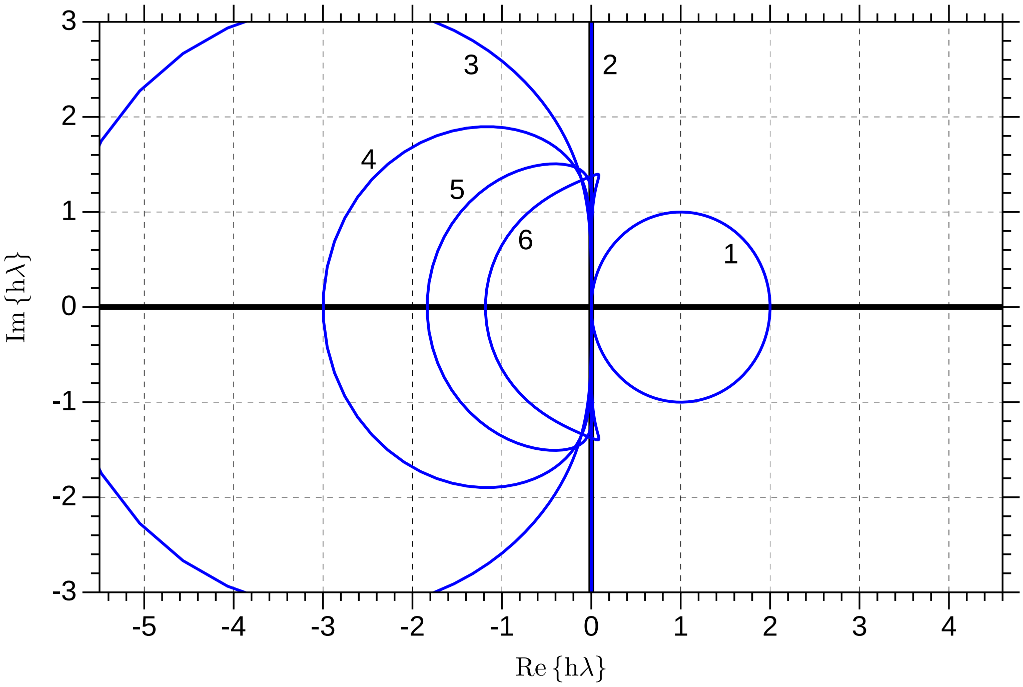


Figure 6.4: areas of absolute stability for order 1. . . 6 Adams-Moulton formulae

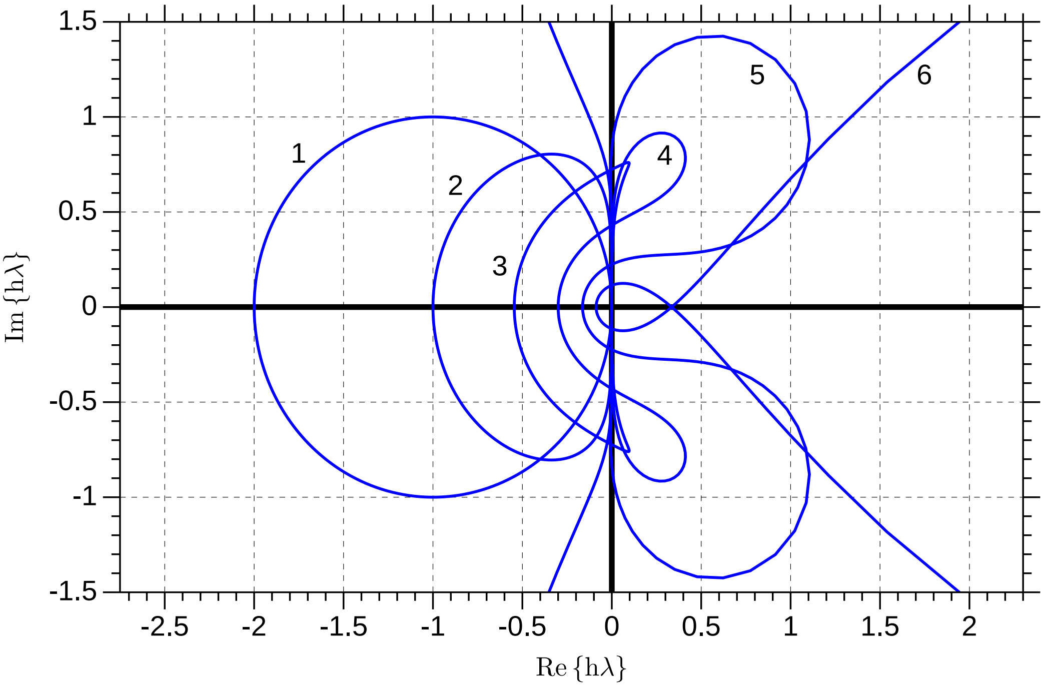


Figure 6.5: areas of absolute stability for order 1. . . 6 Adams-Bashford formulae

Figure 6.6 illustrates a typical problem for the stability of integration methods. A capacitor and a voltage source are connected together directly, i.e. without a damping resistor or something similar in between. After 1ms a linear voltage ramp generates a constant current that loads the capacitance. The only integration method that yields the correct result is backward Euler, because it approximates the charging current with a rectangle, and this is exactly what happens here. The trapezoidal method gives a step twice as high as the correct one, because it approximates the charging current with a triangle. Even more problematic is the stimulated oscillation that isn't damped. The theta-method shows a strongly damped oscillation. The second order Gear method creates a smaller current peak that already disappears after the first time step. Third order Gear shows an oscillation that disappears after the second time step. Oscillations by fourth order Gear disappears after the third time step etc.

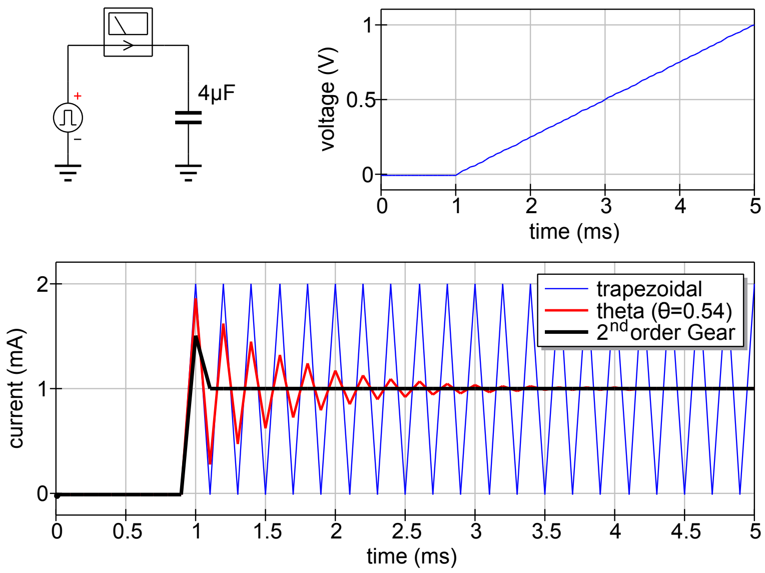


Figure 6.6: response of some integration methods to a step in current

6.7 Energy-storage components

As already mentioned it is essential for the transient analysis to consider the energy storing effects of components. The following section describes how the modified nodal analysis can be used to take this into account.

6.7.1 Capacitor

The integration formulae for linear capacitors have already been described in the chapters before. The equations for implicit Euler, trapezoidal, Gear and Adams-Moulton (AM) are as followed:

$$I_{n+1} = \frac{C}{h_n} V_{n+1} - \frac{C}{h_n} V_n \qquad \text{(implicit Euler)} \tag{6.62}$$

$$I_{n+1} = \frac{2C}{h_n} V_{n+1} - \frac{2C}{h_n} V_n - I_n \qquad \text{(trapezoidal)} \tag{6.63}$$

$$I_{n+1} = \frac{C}{b_{-1}\cdot h_n} V_{n+1} - \frac{a_0 \cdot C}{b_{-1}\cdot h_n} V_n - \frac{a_1 \cdot C}{b_{-1}\cdot h_n} V_{n-1} - \ldots - \frac{a_{k-1} \cdot C}{b_{-1}\cdot h_n} V_{n-k+1} \quad \text{(Gear)} \tag{6.64}$$

$$I_{n+1} = \underbrace{\frac{C}{b_{-1}\cdot h_n}}_{g_{eq}} V_{n+1} \underbrace{- \frac{a_0 \cdot C}{b_{-1}\cdot h_n} V_n - \frac{b_0}{b_{-1}} I_n - \frac{b_1}{b_{-1}} I_{n-1} - \ldots - \frac{b_{k-2}}{b_{-1}} I_{n-k+2}}_{I_{eq}} \quad \text{(AM)} \tag{6.65}$$

Each of these equations can be rewritten as

$$I_{n+1} = g_{eq} \cdot V_{n+1} + I_{eq} \tag{6.66}$$

which leads to the companion model representing a current source with its accompanied internal resistance. Thus the complete MNA matrix equation for an ideal capacitance writes as follows.

$$\begin{bmatrix} +g_{eq} & -g_{eq} \\ -g_{eq} & +g_{eq} \end{bmatrix} \cdot \begin{bmatrix} V_{1,n+1} \\ V_{2,n+1} \end{bmatrix} = \begin{bmatrix} -I_{eq} \\ +I_{eq} \end{bmatrix} \tag{6.67}$$

6.7.2 Inductor

The relation between current and voltage in a linear inductor is:

$$V(t) = \frac{d\Phi}{dt} = L \cdot \frac{dI}{dt} \tag{6.68}$$

Transforming this into an integral equation at discrete time spots yields:

$$\Phi_{n+1} = \Phi_n + \int_{t_n}^{t_{n+1}} I(\tau) \cdot d\tau \tag{6.69}$$

This leads to the following equations for implicit Euler, trapezoidal and Gear methods:

$$I_{n+1} = \frac{h_n}{L} V_{n+1} + I_n \qquad \text{(implicit Euler)} \tag{6.70}$$

$$I_{n+1} = \frac{h_n}{2 \cdot L} V_{n+1} + \frac{h_n}{2 \cdot L} V_n + I_n \qquad \text{(trapezoidal)} \tag{6.71}$$

$$I_{n+1} = \underbrace{\frac{b_{-1} \cdot h_n}{L}}_{g_{eq}} V_{n+1} \underbrace{+ a_0 \cdot I_n + a_1 \cdot I_{n-1} + \ldots + a_{k-1} \cdot I_{n-k+1}}_{I_{eq}} \qquad \text{(Gear)} \tag{6.72}$$

Again each of these equations can be rewritten as

$$I_{n+1} = g_{eq} \cdot V_{n+1} + I_{eq} \tag{6.73}$$

which leads to the same MNA matrix equation as for an ideal capacitance. A disadvantage of this approach is the fact that the inductance appears in the denominator. An alternative would be to use a gyrator in order to transform the inductance into a capacitance (see next section).

6.7.3 Coupled Inductors

In a non-ideal transformer, there are two (or more) coupled inductors. The model for the transient simulation is not very different from the one of a single inductor. In addition to each coil, the mutal inductance has to be counted for.

$$V_{L1} = L_1 \cdot \frac{dI_{L1}}{dt} + M_{12} \cdot \frac{dI_{L2}}{dt} + I_{L1} \cdot R_1 \tag{6.74}$$

$$\text{with} \quad M_{12} = k \cdot \sqrt{L_1 \cdot L_2} \tag{6.75}$$

$$\text{and} \quad R_1 \quad \text{ohmic resistance of coil 1} \tag{6.76}$$

In contrast to the previous section it's also possible to dissolve the equation for $V_{L1,n+1}$:

$$V_{L1,n+1} = r_{eq11} \cdot I_{L1,n+1} + r_{eq12} \cdot I_{L2,n+1} + V_{eq}(I_{L1,n}, I_{L2,n}, ...) \tag{6.77}$$

This keeps the inductance L in the numerator and makes it possible to include the ohmic resistance R_1 into r_{eq11}. For implicit Euler, it therefore follows:

$$V_{L1,n+1} = \underbrace{\left(\frac{L_1}{h_n} + R_1\right)}_{r_{eq11}} \cdot I_{L1,n+1} + \underbrace{\frac{k \cdot \sqrt{L_1 \cdot L_2}}{h_n}}_{r_{eq12}} \cdot I_{L2,n+1} \underbrace{- \left(\frac{L_1}{h_n} + R_1\right) \cdot I_{L1,n} - \frac{k \cdot \sqrt{L_1 \cdot L_2}}{h_n} \cdot I_{L2,n}}_{V_{eq1}} \tag{6.78}$$

The voltage across the secondary coil $V_{L2,n+1}$ goes likewise by just changing the indices. For every inductance this approach creates an additional (internal) node. So finally, the MNA matrix writes (port numbers are according to figure 10.2):

$$\begin{bmatrix} 0 & 0 & 0 & 0 & +1 & 0 \\ 0 & 0 & 0 & 0 & 0 & +1 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ +1 & 0 & 0 & -1 & -r_{eq11} & -r_{eq12} \\ 0 & +1 & -1 & 0 & -r_{eq21} & -r_{eq22} \end{bmatrix} \cdot \begin{bmatrix} V_{1,n+1} \\ V_{2,n+1} \\ V_{3,n+1} \\ V_{4,n+1} \\ I_{L1,n+1} \\ I_{L2,n+1} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ V_{eq1} \\ V_{eq2} \end{bmatrix} \tag{6.79}$$

These equations also give an idea on how to model more than two coupled inductors. For three coupled inductors, the voltage across coil 1 writes:

$$
\begin{aligned}
V_{L1} &= L_1 \cdot \frac{dI_{L1}}{dt} + M_{12} \cdot \frac{dI_{L2}}{dt} + M_{13} \cdot \frac{dI_{L3}}{dt} + I_{L1} \cdot R_1 && (6.80)\\
V_{L2} &= L_2 \cdot \frac{dI_{L2}}{dt} + M_{12} \cdot \frac{dI_{L1}}{dt} + M_{23} \cdot \frac{dI_{L3}}{dt} + I_{L2} \cdot R_2 && (6.81)\\
V_{L3} &= L_3 \cdot \frac{dI_{L3}}{dt} + M_{13} \cdot \frac{dI_{L1}}{dt} + M_{23} \cdot \frac{dI_{L2}}{dt} + I_{L3} \cdot R_3 && (6.82)\\
\text{with} \quad M_{12} &= k_{12} \cdot \sqrt{L_1 \cdot L_2} && (6.83)\\
\text{and} \quad M_{13} &= k_{13} \cdot \sqrt{L_1 \cdot L_3} && (6.84)\\
\text{and} \quad M_{23} &= k_{23} \cdot \sqrt{L_2 \cdot L_3} && (6.85)
\end{aligned}
$$

This can be easily extended to an arbitrary number of coupled inductors.

6.7.4 Non-linear Capacitance

For non-linear capacitances (like depletion capacitance) the following holds.

$$
I(t) = \frac{\partial Q}{\partial t} \qquad \text{and} \qquad C = \frac{\partial Q}{\partial V} \tag{6.86}
$$

which gives:

$$
\begin{aligned}
Q_{n+1} &= Q_n + h_n \cdot I_{n+1} && \text{(implicit Euler)} && (6.87)\\
Q_{n+1} &= Q_n + \frac{h_n}{2} \cdot (I_{n+1} + I_n) && \text{(trapezoidal)} && (6.88)\\
Q_{n+1} &= a_0 \cdot Q_n + a_1 \cdot Q_{n-1} + \ldots + a_{k-1} \cdot Q_{n-k+1} + b_{-1} \cdot h_n \cdot I_{n+1} && \text{(Gear)} && (6.89)
\end{aligned}
$$

By the use of the Newton-Raphson formula

$$
V_{n+1,m+1} = V_{n+1,m} - \frac{f\left(V_{n+1,m}\right)}{f'\left(V_{n+1,m}\right)} \tag{6.90}
$$

an iterative algorithm is obtained. The indices m indicated the m-th Newton-Raphson iteration. Now it is possible to derive the equations for the circuit simulation. For the implicit Euler integration it is as follows.

$$
\begin{aligned}
f\left(V_{n+1,m}\right) &= Q_{n+1,m} - Q_n - h_n \cdot I_{n+1,m} && (6.91)\\
f'\left(V_{n+1,m}\right) &= \frac{\partial f\left(V_{n+1,m}\right)}{\partial V_{n+1,m}} = C_{n+1,m} && (6.92)\\
\Rightarrow I_{n+1,m} &= \frac{1}{h_n} \cdot \left(Q_{n+1,m} + C_{n+1,m} \cdot \left(V_{n+1,m+1} - V_{n+1,m}\right) - Q_n\right) && (6.93)\\
&= \underbrace{\frac{C_{n+1,m}}{h_n}}_{g_{eq}} V_{n+1,m+1} + \underbrace{\frac{Q_{n+1,m} - Q_n}{h_n} - \frac{C_{n+1,m}}{h_n} \cdot V_{n+1,m}}_{I_{eq}} && (6.94)
\end{aligned}
$$

The double indices now indicate the n-th integration step and the m-th Newton-Raphson iteration.

The principle can be extended to the multi-dimension case as usual. To give an example, a non-linear capacitance is assumed that depends on two voltages V_1 and V_2. The backward Euler integration of the current flowing out of terminal 1 is thus:

$$
\begin{aligned}
I_{1,n+1,m+1} =& \frac{Q_{n+1,m} - Q_n}{h_n} + \\
& \frac{C_{11,n+1,m} \cdot \left(V_{1,n+1,m+1} - V_{1,n+1,m}\right) + C_{12,n+1,m} \cdot \left(V_{2,n+1,m+1} - V_{2,n+1,m}\right)}{h_n}
\end{aligned} \tag{6.95}
$$

For the trapezoidal integration the equations for the circuit simulation are as follows.

$$f\left(V_{n+1,m}\right) = Q_{n+1,m} - Q_n - \frac{h_n}{2} \cdot \left(I_{n+1,m} + I_n\right) \tag{6.96}$$

$$f'\left(V_{n+1,m}\right) = \frac{\partial f\left(V_{n+1,m}\right)}{\partial V_{n+1,m}} = C_{n+1,m} \tag{6.97}$$

$$\Rightarrow I_{n+1,m} = \frac{2}{h_n} \cdot \left(Q_{n+1,m} + C_{n+1,m} \cdot \left(V_{n+1,m+1} - V_{n+1,m}\right) - Q_n\right) - I_n \tag{6.98}$$

$$= \underbrace{\frac{2 \cdot C_{n+1,m}}{h_n}}_{g_{eq}} V_{n+1,m+1} + \underbrace{2 \cdot \frac{Q_{n+1,m} - Q_n - C_{n+1,m} \cdot V_{n+1,m}}{h_n} - I_n}_{I_{eq}} \tag{6.99}$$

So, finally, when convergence is reached the current for the next iteration step is:

$$I_{n+1} = \frac{2}{h_n} \cdot \left(Q_{n+1,m} - Q_n\right) - I_n \tag{6.100}$$

For the Gear integration the equations for the circuit simulation are as follows.

$$f\left(V_{n+1,m}\right) = Q_{n+1,m} - a_0 \cdot Q_n - a_1 \cdot Q_{n-1} - \ldots - a_{k-1} \cdot Q_{n-k+1} - b_{-1} \cdot h_n \cdot I_{n+1,m} \tag{6.101}$$

$$f'\left(V_{n+1,m}\right) = \frac{\partial f\left(V_{n+1,m}\right)}{\partial V_{n+1,m}} = C_{n+1,m} \tag{6.102}$$

$$\begin{aligned} I_{n+1,m} = & \underbrace{\frac{C_{n+1,m}}{b_{-1} \cdot h_n}}_{g_{eq}} V_{n+1,m+1} \\ & + \underbrace{\frac{Q_{n+1,m} - a_0 \cdot Q_n - \ldots - a_{k-1} \cdot Q_{n-k+1} - C_{n+1,m} \cdot V_{n+1,m}}{b_{-1} \cdot h_n}}_{I_{eq}} \end{aligned} \tag{6.103}$$

6.8 Components defined in the frequency domain

The time-domain simulation of components defined in the frequency-domain can be performed using an inverse Fourier transformation of the Y-parameters of the component (giving the impulse response) and an adjacent convolution with the prior node voltages (or branch currents) of the component.

This requires a memory of the node voltages and branch currents for each component defined in the frequency-domain. During a transient simulation the time steps are not equidistant and the maximum required memory length T_{end} of a component may not correspond with the time grid produced by the time step control (see section 6.5 on page 58) of the transient simulation. That is why an interpolation of exact values (voltage or current) at a given point in time is necessary.

Components defined in the frequency-domain can be divided into two major classes.

- Components with frequency-independent (non-dispersive) delay times and with or without constant losses.
- Components with frequency-dependent (dispersive) delay times and losses.

6.8.1 Components with frequency-independent delay times

Components with constant delay times are a special case. The impulse response corresponds to the node voltages and/or branch currents at some prior point in time optionally multiplied with a constant loss factor.

Voltage controlled current source

With no constant delay time the MNA matrix entries of a voltage controlled current source is determined by the following equations according to the node numbering in fig. 10.9 on page 115.

$$I_2 = -I_3 = G \cdot (V_1 - V_4) \tag{6.104}$$

The equations yield the following MNA entries during the transient analysis.

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ +G & 0 & 0 & -G \\ -G & 0 & 0 & +G \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} \tag{6.105}$$

With a constant delay time τ eq. (6.104) rewrites as

$$I_2(t) = -I_3(t) = G \cdot (V_1(t-\tau) - V_4(t-\tau)) \tag{6.106}$$

which yields the following MNA entries during the transient analysis.

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ V_3(t) \\ V_4(t) \end{bmatrix} = \begin{bmatrix} I_1(t) \\ -G \cdot (V_1(t-\tau) - V_4(t-\tau)) \\ +G \cdot (V_1(t-\tau) - V_4(t-\tau)) \\ I_4(t) \end{bmatrix} \tag{6.107}$$

Voltage controlled voltage source

The MNA matrix entries of a voltage controlled voltage source are determined by the following characteristic equation according to the node numbering in fig. 10.11 on page 116.

$$V_2 - V_3 = G \cdot (V_4 - V_1) \tag{6.108}$$

This equation yields the following augmented MNA matrix entries with a single extra branch equation.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ G & -1 & 1 & -G & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ J_1 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \end{bmatrix} \tag{6.109}$$

When considering an additional constant time delay τ eq. (6.108) must be rewritten as

$$V_2(t) - V_3(t) = G \cdot (V_4(t-\tau) - V_1(t-\tau)) \tag{6.110}$$

This representation requires a change of the MNA matrix entries which now yield the following matrix equation.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ V_3(t) \\ V_4(t) \\ J_1(t) \end{bmatrix} = \begin{bmatrix} I_1(t) \\ I_2(t) \\ I_3(t) \\ I_4(t) \\ G \cdot (V_4(t-\tau) - V_1(t-\tau)) \end{bmatrix} \tag{6.111}$$

Current controlled current source

With no time delay the MNA matrix entries of a current controlled current source are determined by the following equations according to the node numbering in fig. 10.10 on page 115.

$$I_2 = -I_3 = G \cdot I_1 = -G \cdot I_4 \tag{6.112}$$

$$V_1 = V_4 \tag{6.113}$$

These equations yield the following MNA matrix entries using a single extra branch equation.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & G \\ 0 & 0 & 0 & 0 & -G \\ 0 & 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ J_1 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \end{bmatrix} \tag{6.114}$$

When additional considering a constant delay time τ eq. (6.112) must be rewritten as

$$I_2\left(t\right) = -I_3\left(t\right) = G \cdot I_1\left(t-\tau\right) = -G \cdot I_4\left(t-\tau\right) \tag{6.115}$$

Thus the MNA matrix entries change as well yielding

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1\left(t\right) \\ V_2\left(t\right) \\ V_3\left(t\right) \\ V_4\left(t\right) \\ J_1\left(t\right) \end{bmatrix} = \begin{bmatrix} I_1\left(t\right) \\ -G \cdot J_1\left(t-\tau\right) \\ +G \cdot J_1\left(t-\tau\right) \\ I_4\left(t\right) \\ 0 \end{bmatrix} \tag{6.116}$$

Current controlled voltage source

The MNA matrix entries for a current controlled voltage source are determined by the following characteristic equations according to the node numbering in fig. 10.12 on page 117.

$$V_2 - V_3 = G \cdot I_2 = -G \cdot I_3 \tag{6.117}$$

$$V_1 = V_4 \tag{6.118}$$

These equations yield the following MNA matrix entries.

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 & G & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ J_1 \\ J_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \\ 0 \end{bmatrix} \tag{6.119}$$

With an additional time delay τ between the input current and the output voltage eq. (6.117) rewrites as

$$V_2\left(t\right) - V_3\left(t\right) = G \cdot I_2\left(t-\tau\right) = -G \cdot I_3\left(t-\tau\right) \tag{6.120}$$

Due to the additional time delay the MNA matrix entries must be rewritten as follows

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1\left(t\right) \\ V_2\left(t\right) \\ V_3\left(t\right) \\ V_4\left(t\right) \\ J_1\left(t\right) \\ J_2\left(t\right) \end{bmatrix} = \begin{bmatrix} I_1\left(t\right) \\ I_2\left(t\right) \\ I_3\left(t\right) \\ I_4\left(t\right) \\ 0 \\ G \cdot J_1\left(t-\tau\right) \end{bmatrix} \tag{6.121}$$

Ideal transmission line

The A-parameters of a transmission line (see eq (10.209) on page 119) are defined in the frequency domain. The equation system formed by these parameters write as

$$\text{I.}\quad V_1 = V_2 \cdot \cosh\left(\gamma \cdot l\right) + I_2 \cdot Z_L \cdot \sinh\left(\gamma \cdot l\right) \tag{6.122}$$

$$\text{II.}\quad I_1 = V_2 \cdot \frac{1}{Z_L} \sinh\left(\gamma \cdot l\right) + I_2 \cdot \cosh\left(\gamma \cdot l\right) \tag{6.123}$$

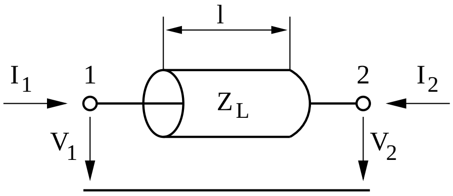


Figure 6.7: ideal transmission line

Applying $\text{I} + Z_L \cdot \text{II}$ and $\text{I} - Z_L \cdot \text{II}$ to the above equation system and using the following transformations

$$\cosh x + \sinh x = \frac{e^x + e^{-x}}{2} + \frac{e^x - e^{-x}}{2} = e^x \tag{6.124}$$

$$\cosh x - \sinh x = \frac{e^x + e^{-x}}{2} - \frac{e^x - e^{-x}}{2} = e^{-x} \tag{6.125}$$

yields

$$V_1 = V_2 \cdot e^{-\gamma \cdot l} + Z_L \cdot \left(I_1 + I_2 \cdot e^{-\gamma \cdot l}\right) \tag{6.126}$$

$$V_2 = V_1 \cdot e^{-\gamma \cdot l} + Z_L \cdot \left(I_2 + I_1 \cdot e^{-\gamma \cdot l}\right) \tag{6.127}$$

whereas γ denotes the propagation constant $\alpha + j\beta$, l the length of the transmission line and Z_L the line impedance.

These equations can be transformed from the frequency domain into the time domain using the inverse Fourier transformation. The frequency independent loss $\alpha \neq f\left(\omega\right)$ gives the constant factor

$$A = e^{-\alpha \cdot l} \tag{6.128}$$

The only remaining frequency dependent term is

$$e^{-j\beta \cdot l} = e^{-j\omega \cdot \tau} \quad \text{with} \quad \beta = \frac{\omega}{v_{ph}} = \frac{\omega}{c_0} = \frac{\omega \cdot \tau}{l} \tag{6.129}$$

which yields the following transformation

$$f\left(\omega\right) \cdot e^{-\gamma \cdot l} = A \cdot f\left(\omega\right) \cdot e^{-j\omega \cdot \tau} \Longleftrightarrow A \cdot f\left(t - \tau\right) \tag{6.130}$$

All the presented time-domain models with a frequency-independent delay time are based on this simple transformation. It can be applied since the phase velocity $v_{ph} \neq f\left(\omega\right)$ is not a function of the frequency. This is true for all non-dispersive transmission media, e.g. air or vacuum. The given transformation can now be applied to the eq. (6.126) and eq. (6.127) defined in the frequency-domain to obtain equations in the time-domain.

The length T_{end} of the memory needed by the ideal transmission line can be easily computed by

$$T_{end} = \tau = \frac{l}{v_{ph}} = \frac{l}{c_0} \tag{6.131}$$

whereas c_0 denotes the speed of light in free space (since there is no dielectric involved during transmission) and l the physical length of the transmission line.

The MNA matrix for a lossy transmission line during the transient analysis is augmented by two new rows and columns in order to consider the following branch equations.

$$V_1(t) = Z_L \cdot I_1(t) + A \cdot (Z_L \cdot I_2(t-\tau) + V_2(t-\tau)) \tag{6.132}$$

$$V_2(t) = Z_L \cdot I_2(t) + A \cdot (Z_L \cdot I_1(t-\tau) + V_1(t-\tau)) \tag{6.133}$$

Thus the MNA matrix entries can be written as

$$\begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & Z_L & 0 \\ 0 & -1 & 0 & Z_L \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ I_1(t) \\ I_2(t) \end{bmatrix} = \begin{bmatrix} I_1(t) \\ I_2(t) \\ A \cdot (V_2(t-\tau) + Z_L \cdot I_2(t-\tau)) \\ A \cdot (V_1(t-\tau) + Z_L \cdot I_1(t-\tau)) \end{bmatrix} \tag{6.134}$$

Ideal 4-terminal transmission line

The ideal 4-terminal transmission line is a two-port as well. It differs from the 2-terminal line as shown in figure 6.8 in two new node voltages and branch currents.

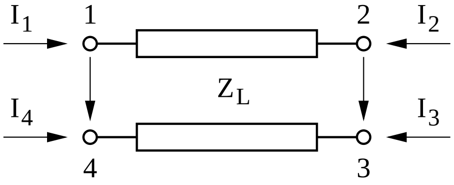


Figure 6.8: ideal 4-terminal transmission line

The differential mode of the ideal 4-terminal transmission line can be modeled by modifying the branch eqs. (6.132) and (6.133) of the 2-terminal line which yields

$$V_1(t) - V_4(t) = Z_L \cdot I_1(t) + A \cdot (Z_L \cdot I_2(t-\tau) + V_2(t-\tau) - V_3(t-\tau)) \tag{6.135}$$

$$V_2(t) - V_3(t) = Z_L \cdot I_2(t) + A \cdot (Z_L \cdot I_1(t-\tau) + V_1(t-\tau) - V_4(t-\tau)) \tag{6.136}$$

Two more conventions must be indroduced

$$I_1(t) = -I_4(t) \tag{6.137}$$

$$I_2(t) = -I_3(t) \tag{6.138}$$

which is valid for the differential mode (i.e. the odd mode) of the transmission line and represents a kind of current mirror on each transmission line port.

According to these consideration the MNA matrix entries during transient analysis are

$$\begin{bmatrix} . & . & . & . & 1 & 0 \\ . & . & . & . & 0 & 1 \\ . & . & . & . & 0 & -1 \\ . & . & . & . & -1 & 0 \\ -1 & 0 & 0 & 1 & Z_L & 0 \\ 0 & -1 & 1 & 0 & 0 & Z_L \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ V_3(t) \\ V_4(t) \\ J_1(t) \\ J_2(t) \end{bmatrix} = \begin{bmatrix} I_1(t) \\ I_2(t) \\ I_3(t) \\ I_4(t) \\ A \cdot (V_2(t-\tau) - V_3(t-\tau) + Z_L \cdot J_2(t-\tau)) \\ A \cdot (V_1(t-\tau) - V_4(t-\tau) + Z_L \cdot J_1(t-\tau)) \end{bmatrix} \tag{6.139}$$

Ideal coupled transmission line

The ideal coupled transmission line is a four-port device with the same nodes like the 4-terminal line in figure 6.8. There are two independent modes travelling along the line, the even mode (or common mode) and the odd mode (or differential mode).

$$V_1 = V_e + V_o \qquad V_4 = V_e - V_o \tag{6.140}$$

$$I_1 = I_e + I_o \qquad I_4 = I_e - I_o \tag{6.141}$$

or

$$V_e = \frac{1}{2} \cdot (V_1 + V_4) \qquad V_o = \frac{1}{2} \cdot (V_1 - V_4) \tag{6.142}$$

$$I_e = \frac{1}{2} \cdot (I_1 + I_4) \qquad I_o = \frac{1}{2} \cdot (I_1 - I_4) \tag{6.143}$$

Therefore, there are now four additional rows (ans columns) in the MNA matrix:

$$\begin{bmatrix} . & . & . & . & 1 & 1 & 0 & 0 \\ . & . & . & . & 0 & 0 & 1 & 1 \\ . & . & . & . & 0 & 0 & 1 & -1 \\ . & . & . & . & 1 & -1 & 0 & 0 \\ -0.5 & 0 & 0 & -0.5 & Z_{Le} & 0 & 0 & 0 \\ -0.5 & 0 & 0 & +0.5 & 0 & Z_{Lo} & 0 & 0 \\ 0 & -0.5 & -0.5 & 0 & 0 & 0 & Z_{Le} & 0 \\ 0 & -0.5 & +0.5 & 0 & 0 & 0 & 0 & Z_{Lo} \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ V_3(t) \\ V_4(t) \\ J_{1e}(t) \\ J_{1o}(t) \\ J_{2e}(t) \\ J_{2o}(t) \end{bmatrix}$$
$$= \begin{bmatrix} I_1(t) \\ I_2(t) \\ I_3(t) \\ I_4(t) \\ A_e \cdot \left(\frac{1}{2} \cdot (V_2(t-\tau) + V_3(t-\tau)) + Z_{Le} \cdot J_{2e}(t-\tau)\right) \\ A_o \cdot \left(\frac{1}{2} \cdot (V_2(t-\tau) - V_3(t-\tau)) + Z_{Lo} \cdot J_{2o}(t-\tau)\right) \\ A_e \cdot \left(\frac{1}{2} \cdot (V_1(t-\tau) + V_4(t-\tau)) + Z_{Le} \cdot J_{1e}(t-\tau)\right) \\ A_o \cdot \left(\frac{1}{2} \cdot (V_1(t-\tau) - V_4(t-\tau)) + Z_{Lo} \cdot J_{1o}(t-\tau)\right) \end{bmatrix} \tag{6.144}$$

Logical devices

The analogue models of logical (digital) components explained in section 11.7 on page 159 do not include delay times. With a constant delay time τ the determining equations for the logical components yield

$$u_{out}(t) = f\left(V_{in,1}(t-\tau), V_{in,2}(t-\tau), \ldots\right) \tag{6.145}$$

With the prior node voltages $V_{in,n}(t-\tau)$ known the MNA matrix entries in eq. (11.278) can be rewritten as

$$\begin{bmatrix} . & . & . & 1 \\ . & . & . & 0 \\ . & . & . & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_{out}(t) \\ V_{in,1}(t) \\ V_{in,2}(t) \\ I_{out}(t) \end{bmatrix} = \begin{bmatrix} I_0(t) \\ I_1(t) \\ I_2(t) \\ u_{out}(t) \end{bmatrix} \tag{6.146}$$

during the transient analysis. The components now appear to be simple linear components. The derivatives are not anymore necessary for the Newton-Raphson iterations. This happens to be because the output voltage does not depend directly on the input voltage(s) at exactly the same time point.

6.8.2 Components with frequency-dependent delay times and losses

In the general case a component with P ports which is defined in the frequency-domain can be represented by the following matrix equation.

$$\begin{bmatrix} Y_{11} & Y_{12} & \dots & Y_{1P} \\ Y_{21} & Y_{22} & & Y_{2P} \\ \vdots & & \ddots & \vdots \\ Y_{P1} & Y_{P2} & \dots & Y_{PP} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ \vdots \\ V_P \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ \vdots \\ I_P \end{bmatrix} \tag{6.147}$$

This matrix representation is the MNA representation during the AC analysis. With no specific time-domain model at hand the equation

$$\left[Y\left(j\omega\right)\right] \cdot \left[V\left(j\omega\right)\right] = \left[I\left(j\omega\right)\right] \tag{6.148}$$

must be transformed into the time-domain using a Fourier transformation.

The convolution integral

The multiplication in the frequency-domain is equivalent to a convolution in the time-domain after the transformation. It yields the following matrix equation

$$\left[H\left(t\right)\right] * \left[V\left(t\right)\right] = \left[I\left(t\right)\right] \tag{6.149}$$

whereas $H(t)$ is the impulse response based on the frequency-domain model and the $*$ operator denotes the convolution integral

$$H\left(t\right) * V\left(t\right) = \int_{-\infty}^{+\infty} H\left(\tau\right) \cdot V\left(t-\tau\right) d\tau \tag{6.150}$$

The lower bound of the given integral is set to zero since both the impulse response as well as the node voltages are meant to deliver no contribution to the integral. Otherwise the circuit appears to be unphysical. The upper limit should be bound to a maximum impulse response time T_{end}

$$H\left(t\right) * V\left(t\right) = \int_{0}^{T_{end}} H\left(\tau\right) \cdot V\left(t-\tau\right) d\tau \tag{6.151}$$

with

$$H\left(\tau\right) = 0 \quad \forall \quad \tau > T_{end} \tag{6.152}$$

Since there is no analytic represention for the impulse response as well as for the node voltages eq. (6.151) must be rewritten to

$$H\left(n\cdot\Delta t\right) * V\left(n\cdot\Delta t\right) = \sum_{k=0}^{N-1} H\left(k\cdot\Delta t\right) \cdot V\left(\left(n-k\right)\cdot\Delta t\right) \tag{6.153}$$

with

$$\Delta t = \frac{T_{end}}{N} \tag{6.154}$$

whereas N denotes the number of samples to be used during numerical convolution. Using the current time step $t = n \cdot \Delta t$ it is possible to express eq. (6.153) as

$$I\left(t\right) = H\left(0\right) \cdot V\left(t\right) + \underbrace{\sum_{k=1}^{N-1} H\left(k\cdot\Delta t\right) \cdot V\left(t-k\cdot\Delta t\right)}_{I_{eq}} \tag{6.155}$$

With $G = H(0)$ the resulting MNA matrix equation during the transient analysis gets

$$\left[G\right] \cdot \left[V(t)\right] = \left[I(t)\right] - \left[I_{eq}\right] \tag{6.156}$$

This means, the component defined in the frequency-domain can be expressed with an equivalent DC admittance G and additional independent current sources in the time-domain. Each independent current source at node r delivers the following current

$$I_{eq_r} = \sum_{c=1}^{P} \sum_{k=1}^{N-1} H_{rc}(k \cdot \Delta t) \cdot V_c(t - k \cdot \Delta t) \tag{6.157}$$

whereas V_c denotes the node voltage at node c at some prior time and H_{rc} the impulse response of the component based on the frequency-domain representation. The MNA matrix equation during transient analysis can thus be written as

$$\begin{bmatrix} G_{11} & G_{12} & \dots & G_{1P} \\ G_{21} & G_{22} & & G_{2P} \\ \vdots & & \ddots & \vdots \\ G_{P1} & G_{P2} & \dots & G_{PP} \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ \vdots \\ V_P(t) \end{bmatrix} = \begin{bmatrix} I_1(t) \\ I_2(t) \\ \vdots \\ I_P(t) \end{bmatrix} - \begin{bmatrix} I_{eq_1} \\ I_{eq_2} \\ \vdots \\ I_{eq_P} \end{bmatrix} \tag{6.158}$$

Frequency- to time-domain transformation

With the number of samples N being a power of two it is possible to use the Inverse Fast Fourier Transformation (IFFT). The transformation to be performed is

$$Y(j\omega) \Leftrightarrow H(t) \tag{6.159}$$

The maximum impulse response time of the component is specified by T_{end} requiring the following transformation pairs.

$$Y(j\omega_i) \Leftrightarrow H(t_i) \quad \text{with} \quad i = 0, 1, 2, \dots, N-1 \tag{6.160}$$

with

$$t_i = 0, \Delta t, 2 \cdot \Delta t, \dots, (N-1) \cdot \Delta t \tag{6.161}$$

$$\omega_i = 0, \frac{1}{T_{end}}, \frac{2}{T_{end}}, \dots, \frac{N/2}{T_{end}} \tag{6.162}$$

The frequency samples in eq. (6.162) indicate that only half the values are required to obtain the appropriate impulse response. This is because the impulse response $H(t)$ is real valued and that is why

$$Y(j\omega) = Y^*(-j\omega) \tag{6.163}$$

The maximum frequency considered is determined by the maximum impulse response time T_{end} and the number of time samples N.

$$f_{max} = \frac{N/2}{T_{end}} = \frac{1}{2 \cdot \Delta t} \tag{6.164}$$

It could prove useful to weight the Y-parameter samples in the frequency-domain by multiplying them with an appropriate windowing function (e.g. Kaiser-Bessel).
In order to reach an appropriate accuracy for the simulation result it is crucial that the maximum impulse response time T_{end} in the Fourier transformation is high enough. I.e. the impulse response must have decayed to almost zero at T_{end}. This is often hard to fulfill. Methods that avoid the problem exist. The so-called DeHoog algorithm is considered as the most powerful one [14].

Implementation considerations

For the method presented the Y-parameters of a component must be finite for all frequencies and must converge for $f \to f_{max}$. This cannot be ensured for the general case (e.g. for an ideal inductor or capacitor).

6.9 Noise in time domain

In order to generate a noise voltage or current in time domain pseudo-random number generators (PRNG) are used [15], [16]. A C++ function using this approach is depicted below.

Listing 6.1: noise generator

```
// *********************************************************************
// generates 2^31-2 random numbers using linear congruential generator
// f(z) = 16807 z mod (2^31 - 1)
// It provides uniform-distributed pseudo-random numbers from -1.0 to 1.0
// with mean = 0 and variance = 1/3.
// *********************************************************************
static unsigned int seed = 1;

double get_uniform_noise()
{
  unsigned int lo, hi;

  lo = 16807 * (seed & 0xFFFF);
  hi = 16807 * (seed >> 16);

  lo += (hi & 0x7FFF) << 16;
  lo += hi >> 15;

  if(lo > 0x7FFFFFFF)
    lo -= 0x7FFFFFFF;

  seed = lo;
  return double(seed) / double(0x40000000) - 1.0;
}
```

The uniform-distributed number sequence can be used to create noise with an arbitrary frequency dependency [17]:

$$e_n(f) = \sqrt{P_n(f)} \cdot \exp(j \cdot \pi \cdot n(f)) \tag{6.165}$$

where $P_n(f)$ is the power spectral density (PSD) and $n(f)$ is a pseudo-random number sequence uniform-distributed from -1 to +1. Performing an inverse Fourier transformation on the number sequence $e_n(f)$ generates time-domain noise with a normal-distributed amplitude.

Correlated noise can be created by a linear combination of uncorrelated noise:

$$u_1(f) = U_1 \cdot \left(\sqrt{1-k} \cdot e_{n1}(f) + \sqrt{k} \cdot e_{n3}(f)\right) \tag{6.166}$$

$$u_2(f) = U_2 \cdot \left(\sqrt{1-k} \cdot e_{n2}(f) + \sqrt{k} \cdot e_{n3}(f)\right) \tag{6.167}$$

where $e_{n1}(f)$ to $e_{n3}(f)$ are normalized number sequences according to equation 6.165 and k is the correlation coefficient, i.e. $k = 0$ means no correlation between $u_1(f)$ and $u_2(f)$ and $k = 1$ means complete correlation.

6.10 Periodic Steady-State Analysis

Periodic steady-state (PSS) analysis is a transient simulation that automatically finds the steady state condition for periodic signals. It works for autonomous circuits (oscillators) as well as for non-autonomous ones.

For non-autonomous circuits the principle is quite simple, because the period T is known. Hence, the algorithm performs a transient analysis and compares the state variables of two periods. The PSS is reached if the difference is below a user defined limit.

For autonomous circuits things are a little bit more complex, because the period T is unknown. Thus, the algorithm has to search for the correct periodicity.

Chapter 7

Harmonic Balance Analysis

Harmonic balance is a non-linear, frequency-domain, steady-state simulation. The voltage and current sources create discrete frequencies resulting in a spectrum of discrete frequencies at every node in the circuit. Linear circuit components are solely modeled in frequency domain. Non-linear components are modeled in time domain and Fourier-transformed before each solving step. The informations in this chapter are taken from [18] (chapter 3) which is a very nice and well-written publication on this topic.

The harmonic balance simulation is ideal for situations where transient simulation methods are problematic. These are:

- components modeled in frequency domain, for instance (dispersive) transmission lines
- circuit time constants large compared to period of simulation frequency
- circuits with lots of reactive components

Harmonic balance methods, therefore, are the best choice for most microwave circuits excited with sinusoidal signals (e.g. mixers, power amplifiers).

7.1 The Basic Concept

As the non-linear elements are still modeled in time domain, the circuit first must be separated into a linear and a non-linear part. The internal impedances Z_i of the voltage sources are put into the linear part as well. Figure 7.1 illustrates the concept. Let us define the following symbols:

M = number of (independent) voltage sources

N = number of connections between linear and non-linear subcircuit

K = number of calculated harmonics

L = number of nodes in linear subcircuit

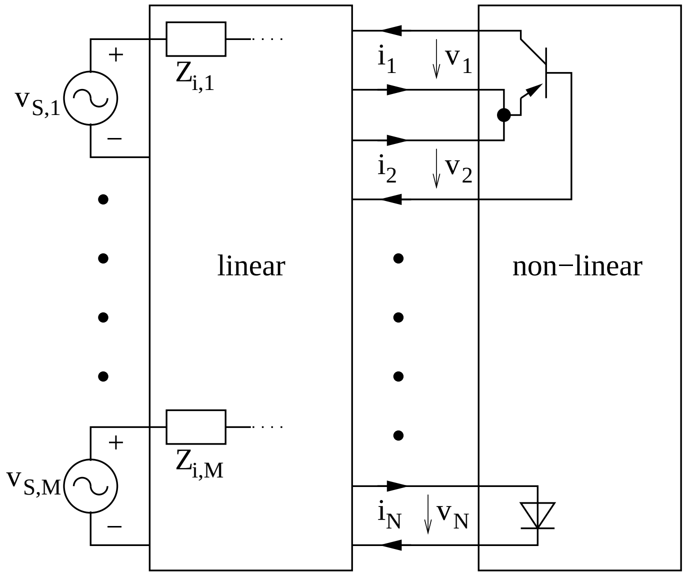


Figure 7.1: circuit partitioning in harmonic balance

The linear circuit is modeled by two transadmittance matrices: The first one $\tilde{\mathbf{Y}}$ relates the source voltages $v_{S,1}...v_{S,M}$ to the interconnection currents $i_1...i_N$ and the second one $\hat{\mathbf{Y}}$ relates the interconnection voltages $v_1...v_N$ to the interconnection currents $i_1...i_N$. Taking both, we can express the current flowing through the interconnections between linear and non-linear subcircuit:

$$\mathbf{I} = \tilde{\mathbf{Y}}_{N\times M} \cdot \mathbf{V}_S + \hat{\mathbf{Y}}_{N\times N} \cdot \mathbf{V} = \mathbf{I}_S + \hat{\mathbf{Y}} \cdot \mathbf{V} \tag{7.1}$$

Because $\mathbf{V}_S$ is known and constant, the first term can already be computed to give $\mathbf{I}_S$. Taking the whole linear network as one block is called the "piecewise" harmonic balance technique.

The non-linear circuit is modeled by its current function $i(t) = f_g(v_1, ..., v_P)$ and by the charge of its capacitances $q(t) = f_q(v_1, ..., v_Q)$. These functions must be Fourier-transformed to give the frequency-domain vectors $\mathbf{Q}$ and $\mathbf{I}$, respectively.

A simulation result is found if the currents through the interconnections are the same for the linear and the non-linear subcircuit. This principle actually gave the harmonic balance simulation its name, because through the interconnections the currents of the linear and non-linear subcircuits have to be *balanced* at every *harmonic* frequency. To be precise the described method is called Kirchhoff's current law harmonic balance (KCL-HB). Theoretically, it would also be possible to use an algorithm that tries to balance the voltages at the subcircuit interconnections. But then the Z matrix (linear subcircuit) and current-dependend voltage laws (non-linear subcircuit) have to be used. That doesn't fit the need (see other simulation types).

So, the non-linear equation system that needs to be solved writes:

$$\mathbf{F}(\mathbf{V}) = \underbrace{\boldsymbol{I}_S + \hat{\boldsymbol{Y}} \cdot \boldsymbol{V}}_{\text{linear}} + \underbrace{j \cdot \boldsymbol{\Omega} \cdot \boldsymbol{Q} + \boldsymbol{I}}_{\text{non-linear}} = \mathbf{0} \tag{7.2}$$

where matrix $\mathbf{\Omega}$ contains the angular frequencies on the first main diagonal and zeros elsewhere, $\mathbf{0}$ is the zero vector.

After each iteration step, the inverse Fourier transformation must be applied to the voltage vector $\mathbf{V}$. Then the time domain voltages $v_{0,1}...v_{K,N}$ are put into $i(t) = f_g(v_1, ..., v_P)$ and $q(t) =$

$f_q(v_1, ..., v_Q)$ again. Now, a Fourier transformation gives the vectors $\mathbf{Q}$ and $\mathbf{I}$ for the next iteration step. After repeating this several times, a simulation result has hopefully be found.

Having found this result means having got the voltages $v_1 ... v_N$ at the interconnections of the two subcircuits. With these values the voltages at all nodes can be calculated: Forget about the non-linear subcircuit, put current sources at the former interconnections (using the calculated values) and perform a normal AC simulation for each frequency. After that the simulation is complete. Note that in multi-dimensional frequency schemes, many frequency values will appear more than one time. Thus, results at equal frequencies must be added in order to get one result per frequency.

A short note to the construction of the quantities: One big difference between the HB and the conventional simulation types like a DC or an AC simulation is the structure of the matrices and vectors. A vector used in a conventional simulation contains one value for each node. In an HB simulation there are many harmonics and thus, a vector contains K values for each node. This means that within a matrix, there is a $K \times K$ diagonal submatrix for each node. Using this structure, all equations can be written in the usual way, i.e. without paying attention to the special matrix and vector structure. In a computer program, however, a special matrix class is needed in order to not waste memory for the off-diagonal zeros.

7.2 Going through each Step

7.2.1 Creating Transadmittance Matrix

It needs several steps to get the transadmittance matrices $[\tilde{Y}]$ and $[\hat{Y}]$ mentioned in equation (7.1). First the MNA matrix of the linear subcircuit (figure 7.1) is created (chapter 1.1) without the voltage sources $S_1 ... S_M$ and without the non-linear components. Note that all nodes must appear in the matrix, even those where only non-linear components are connected. Then the transimpedance matrix is derived by exciting one by one the port nodes of the MNA matrix with unity current. After that the transadmittance matrix is calculated by inverting the transimpedance matrix. Finally the matrices $[\tilde{Y}]$ and $[\hat{Y}]$ are filled with the corresponding elements of the overall transadmittance matrix.

Note: The MNA matrix of the linear subcircuit has L nodes. Every node, that is connected to the non-linear subcircuit or/and is connected to a voltage source, is called "port" in the following text. So, there are $M + N$ ports. All these ports need to be differential ones, i.e. without ground reference. Otherwise problemes may occur due to singular matrices when calculating $[\tilde{Y}]$ or $[\hat{Y}]$.

Now this should be described in more detail: By use of the MNA matrix $[A]$, the n-th column of the transimpedance matrix $[Z]$ should be calculated. The voltage source at port n is connected to node i (positive terminal) and to node j (negative terminal). This results in the following equation. (If port n is referenced to ground, the -1 is simply omitted.)

$$[A] \cdot \begin{bmatrix} V_1 \\ \vdots \\ V_L \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ -1 \\ \vdots \\ 0 \end{bmatrix} \begin{matrix} \\ \\ \leftarrow i\text{-th row} \\ \\ \leftarrow j\text{-th row} \\ \\ \\ \end{matrix} \tag{7.3}$$

After having solved it, $Z_{1,n} ... Z_{N+M,n}$ are obtained simply by subtraction of the node voltages:

$$Z_{m,n} = V_k - V_l \tag{7.4}$$

Here the voltage source at port m is connected to node k (positive terminal) and to node l (negative terminal).

The next column of $[Z]$ is obtained by changing the right-hand side of equation (7.3) appropriately. As this has to be done $N + M$ times, it is strongly recommended to use LU decomposition.

As $[\tilde{Y}]$ is not square, problems encounter by trying to build its inverse matrix. Therefore, the following procedure is recommended:

- Create the transimpedance matrix for all ports (sources and interconnections).
- Compute the inverse matrix (transadmittance matrix).
- The upper left and upper right corner contains $[\hat{Y}]$ and $[\tilde{Y}]$.
- The lower left and lower right corner contains the transadmittance matrices to calculate the currents through the sources. They can be used to simplify the AC simulation at the very end.

One further thing must be mentioned: Because the non-linear components and the sources are missing in the linear MNA matrix, there are often components that are completely disconnected from the rest of the circuit. The resulting MNA matrix cannot be solved. To avoid this problem, shunt each port with a 100Ω resistor, i.e. place a resistor in parallel to each non-linear component and to each source. The effect of these resistors can be easily removed by subtracting 10mS from the first main diagonal of the transadmittance matrix.

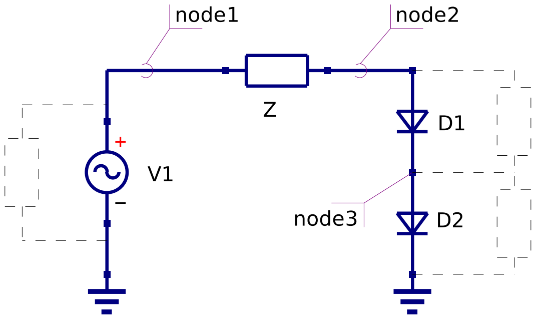


Figure 7.2: example circuit how to create the transadmittance matrices

A little example should demonstrate the creation of the transadmittance matrices. The circuit in figure 7.2 should be solved: An ac voltage source connected to a resistance that is connected to two diodes. The dotted resistances Y_{dot} = 10mS are the ones inserted temporarily to avoid disconnected nodes. The MNA matrix is:

$$[A] = \begin{bmatrix} 1/Z & -1/Z & 0 \\ -1/Z & 1/Z & 0 \\ 0 & 0 & 0 \end{bmatrix} + [Y_{dot}] = \begin{bmatrix} 1/Z + 0.01 & -1/Z & 0 \\ -1/Z & 1/Z + 0.01 & -0.01 \\ 0 & -0.01 & 0.01 + 0.01 \end{bmatrix} \tag{7.5}$$

As can be seen the MNA without Y_{dot} is singular, because it does neither contain the entries for the non-linear components, nor the entries for the sources. The next step is to build the connexion matrix for the sources and the non-linear components:

$$[C] = \begin{bmatrix} 0 & 1 & -1 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix} \begin{array}{l} \leftarrow \text{D1 connected to node 2 and to node 3} \\ \leftarrow \text{D2 connected to node 3 (and to ground)} \\ \leftarrow \text{V1 connected to node 1 (and to ground)} \end{array} \tag{7.6}$$

Now the transimpedance matrix can be calculated:

$$Z_{trans} = [C] \cdot [A]^{-1} \cdot [C]^T \tag{7.7}$$

Finally, the inverse matrix leads to the transadmittance matrix. Also, the influence of the dotted resistances in figure 7.2 is removed as the last step. With $Z = 50\Omega$ this yields:

$$Y_{trans} = Z_{trans}^{-1} - 10\text{mS} \cdot E = \begin{bmatrix} 0.02 & 0.02 & -0.02 \\ 0.02 & 0.02 & -0.02 \\ -0.02 & -0.02 & 0.02 \end{bmatrix} \tag{7.8}$$

The upper left and upper right corner contains $[\hat{Y}]$ and $[\tilde{Y}]$:

$$[\hat{Y}] = \begin{bmatrix} 0.02 & 0.02 \\ 0.02 & 0.02 \end{bmatrix} \qquad [\tilde{Y}] = \begin{bmatrix} -0.02 \\ -0.02 \end{bmatrix} \tag{7.9}$$

Compare the result with figure 1 to see that this works quite fine.

7.2.2 Starting Values

A difficult question is how to find appropriate start values for the harmonic balance simulation. Experience shows that the most simple answer is sufficient, i.e. the starting values are zero voltages. However, convergence may soon become a big problem, especially with a large signal strength and strong non-linearities. Therefore, it is recommended to perform a DC analysis in order to get a better initial guess and use the source-stepping method for all non-DC sources as a fundamental concept. This means that the elements of the current source vector $\mathbf{I}_S$ that don't refer to zero frequency should be increased step by step during the iteration until the full signal levels are reached. This procedure together with the Newton-Raphson method shows to be very robust and reliable.

7.2.3 Solution algorithm

To perform a HB simulation, the multi-dimensional, non-linear function 7.2 has to be solved. One of the best possibilities to do so is the Newton-Raphson method:

$$\mathbf{V}_{n+1} = \mathbf{V}_n - \mathbf{J}(\mathbf{V}_n)^{-1} \cdot \mathbf{F}(\mathbf{V}_n) = \mathbf{V}_n - \Delta\mathbf{V}_n \tag{7.10}$$

with $\mathbf{J}$ being the Jacobian matrix. DC and transient simulation also use this technique, but here $\mathbf{J}$ is needed in frequency domian. Starting at the HB equation 7.2, it is:

$$\boldsymbol{J}(\boldsymbol{V}_n) = \frac{d\boldsymbol{F}(\boldsymbol{V})}{d\boldsymbol{V}} = \hat{\boldsymbol{Y}}_{N\times N} + \frac{\partial \boldsymbol{I}}{\partial \boldsymbol{V}} + j \cdot \boldsymbol{\Omega} \frac{\partial \boldsymbol{Q}}{\partial \boldsymbol{V}} = \hat{\boldsymbol{Y}}_{N\times N} + \boldsymbol{G} + j \cdot \boldsymbol{\Omega} \cdot \boldsymbol{C} \tag{7.11}$$

So, two Jacobian matrices have to be built, one for the current $\boldsymbol{I}$ and one for the charge $\boldsymbol{Q}$. So, finally the following equation system must be solved at every iteration step:

$$\mathbf{J}(\mathbf{V}_n) \cdot \Delta\mathbf{V}_n = \mathbf{F}(\mathbf{V}_n) \tag{7.12}$$

Note that the Nyquist frequencies (including DC) must be kept real-valued during all iteration steps. Furthermore, for non-Nyquist frequencies the elements of the current source vector $\mathbf{I}_S$ must be set for positive and negative frequencies to half of its physical value.

7.2.4 Jacobian matrices in frequency domain

It was shown that Jacobian matrices in frequency domain are needed to solve a harmonic balance simulation. To build them, they are first created in time domain and transformed into frequency domain afterwards. To obtain a practical algorithm for this transformation, the DFT is best written as matrix equation. By having a look at equation 19.178 and 19.179, it becomes clear how this works. The harmonic factors $\exp(j\omega_k t_n)$ build the matrix $\mathbf{\Gamma}$:

$$\text{DFT:} \qquad \boldsymbol{U}(j\omega) = \mathbf{\Gamma} \cdot \boldsymbol{u}(t) \tag{7.13}$$

$$\text{IDFT:} \qquad \boldsymbol{u}(t) = \mathbf{\Gamma}^{-1} \cdot \boldsymbol{U}(j\omega) \tag{7.14}$$

with $\boldsymbol{u}$ and $\boldsymbol{U}$ being the vectors of the time and frequency values, respectively. In contrast to the standard definition of DFT and inverse DFT, for HB simulation the factor $1/N$ must be applied to the DFT and not to the inverse DFT. Now, it is possible to transform the desired Jacobian matrix into frequency domain. As the Fourier Transformation and the differentiation are both linear, they can be exchanged:

$$\boldsymbol{G} = \frac{\partial \boldsymbol{I}}{\partial \boldsymbol{V}} = \frac{\partial(\mathbf{\Gamma} \cdot \boldsymbol{i})}{\partial(\mathbf{\Gamma} \cdot \boldsymbol{v})} = \mathbf{\Gamma} \cdot \frac{\partial \boldsymbol{i}}{\partial \boldsymbol{v}} \cdot \mathbf{\Gamma}^{-1} \tag{7.15}$$

Here $\boldsymbol{i}$ is a vector with length $K \cdot N$, i.e. first all time values of the first node are inserted, then all time values of the second node etc. The Jacobi matrix of $\boldsymbol{i}$ is defined as:

$$\boldsymbol{G}(\boldsymbol{u}) = \begin{bmatrix} \dfrac{\partial i_1}{\partial u_1} & \cdots & \dfrac{\partial i_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial i_n}{\partial u_1} & \cdots & \dfrac{\partial i_n}{\partial u_n} \end{bmatrix} \tag{7.16}$$

Hence this matrix consists of $K \times K$ blocks (one for each node) that are diagonal matrices with time values of the derivatives in it. (Components exists that create non-diagonal blocks, but these are so special ones that they do not appear in this document.)

The formula 7.15 seems to be quite clear, but it has to be pointed out how this works with FFT algorithm. With $\mathbf{\Gamma}^{-1} = (\mathbf{\Gamma}^{-1})^T$ (see equation 19.179) and $(\boldsymbol{A} \cdot \boldsymbol{B})^T = \boldsymbol{B}^T \cdot \boldsymbol{A}^T$, it follows:

$$\boldsymbol{G} = \mathbf{\Gamma} \cdot \frac{\partial \boldsymbol{i}}{\partial \boldsymbol{v}} \cdot \mathbf{\Gamma}^{-1} = \left(\mathbf{\Gamma}^{-1} \cdot \left(\mathbf{\Gamma} \cdot \frac{\partial \boldsymbol{i}}{\partial \boldsymbol{v}} \right)^T \right)^T \tag{7.17}$$

So, there are two steps to perform an FFT-based transformation of the time domain Jacobian matrix into the frequency domain Jacobian:

1. Perform an FFT on every column of the Jacobian and build a new matrix $\boldsymbol{A}$ with this result, i.e. the first column of $\boldsymbol{A}$ is the FFTed first column of the Jacobian and so on.

2. Perform an IFFT on every row of the matrix $\boldsymbol{A}$ and build a new matrix $\boldsymbol{B}$ with this result, i.e. the first row of $\boldsymbol{B}$ is the IFFTed first row of $\boldsymbol{A}$ and so on.

As the Fourier transformation has to be applied to diagonal sub-matrices, the whole above-mentioned process can be performed by one single FFT. This is done by taking the $\partial \boldsymbol{i}/\partial \boldsymbol{v}$ values in a vector $\boldsymbol{J}_i$ and calculating:

$$\frac{1}{K} \cdot \text{FFT}\left(\boldsymbol{J}_i\right) \tag{7.18}$$

The result is the first column of $\boldsymbol{G}$. The second column equals the first one rotated down by one element. The third column is the second one rotated down by one element etc. A matrix obeying this structure is called circulant matrix.

As the time domain values are all real values, the above-mentioned procedure is equivalent to:

$$\text{invFFT}\left(\boldsymbol{J}_i\right) \tag{7.19}$$

The result is the first row of $\boldsymbol{G}$. The second row equals the first one rotated to the right by one element. The third row is the second one rotated to the right by one element etc.

If the matrix for a multi-dimensional FFT has to be built, the procedure is the same. The only difference is that the first row needs to be rotated block-wise for every dimension.

Solving the equation system $G \cdot V = I$ is the most time consuming task of a simulation. Therefore, any redundancy must be avoided. The frequency domain Jacobian matrix (as described above) contains for each frequency element its conjugate-complex counterpart, i.e. the negative frequency element. If each row m is added to its counterpart m', each column n can be summed up with its counterpart n' and a new row evolves that calculates the real part only. For every summed-up element i_{mn} of the current Jacobian matrix $\boldsymbol{G}$ it is:

$$\begin{aligned}
Re(i_{mn}) &= (g_{mn} + g_{m'n}) \cdot v_n + (g_{mn'} + g_{m'n'}) \cdot v_{n'} && (7.20)\\
&= (g_{mn} + g_{m'n}) \cdot v_n + (g^*_{m'n} + g^*_{mn}) \cdot v^*_n && (7.21)\\
&= 2 \cdot (Re(g_{mn}) + Re(g_{m'n})) \cdot Re(v_n) - 2 \cdot (Im(g_{mn}) + Im(g_{m'n})) \cdot Im(v_n) && (7.22)
\end{aligned}$$

If each counterpart m' is substracted from its original row m, the sum creates a row that calculates the imaginary part only:

$$\begin{aligned}
Im(i_{mn}) &= (g_{mn} - g_{m'n}) \cdot v_n + (g^*_{m'n} - g^*_{mn}) \cdot v^*_n && (7.23)\\
&= 2 \cdot (Im(g_{mn}) - Im(g_{m'n})) \cdot Re(v_n) + 2 \cdot (Re(g_{mn}) - Re(g_{m'n})) \cdot Im(v_n) && (7.24)
\end{aligned}$$

The same can be done with the Jacobian matrix of the charge $\boldsymbol{C}$. As this matrix is multiplied by $j\omega$, it yields:

$$\begin{aligned}
Re(i_{mn}) &= -2\omega \cdot (Im(c_{mn}) + Im(c_{m'n})) \cdot Re(v_n) - 2\omega \cdot (Re(c_{mn}) + Re(g_{m'n})) \cdot Im(v_n) && (7.25)\\
Im(i_{mn}) &= 2\omega \cdot (Re(c_{mn}) - Re(c_{m'n})) \cdot Re(v_n) - 2\omega \cdot (Im(c_{mn}) - Im(c_{m'n})) \cdot Im(v_n) && (7.26)
\end{aligned}$$

The linear part of Jacobian matrix $\hat{\boldsymbol{Y}}$ only contains diagonal elements. So, it is:

$$\begin{aligned}
Re(i_{nn}) &= 2 \cdot Re(y_{nn}) \cdot Re(v_n) - 2 \cdot Im(y_{nn}) \cdot Im(v_n) && (7.27)\\
Im(i_{nn}) &= 2 \cdot Im(y_{nn}) \cdot Re(v_n) + 2 \cdot Re(y_{nn}) \cdot Im(v_n) && (7.28)
\end{aligned}$$

The real and imaginary part are now separated, the resulting equation system cannot be written in matrix form. In order to derive it without first creating the standard Jacobian matrix, the indeces mn and $m'n$ must be converted into frequency indeces. This becomes straight-forward when remembering the circulant structure of the Jacobian matrix. It is:

$$\begin{aligned}
mn &\quad\rightarrow\quad n - m && (7.29)\\
m'n &\quad\rightarrow\quad n + m && (7.30)
\end{aligned}$$

Note that the new indeces correspond to the first row of the old Jacobian matrix which is the inverse Fourier transformation. This concept is also true for more-dimensional frequency domain. The index transformation must be applied to every dimension.
For every Nyquist frequency (including DC) the imaginary row and/or column, respectively, needs to be omitted. So the final result is a real-valued equation system of the same size like the original complex-valued one. Thus, the speed improvement is more than a factor of two.

7.2.5 Termination Criteria

Frequency components with very different magnitude appear in harmonic balance simulation. In order to detect when the solver has found an accurate solution, an absolute as well as relative criteria must be used on all nodes and at all frequencies. The analysis is regarded as finished if one of the criteria is satisfied.

The absolute and relative criteria write as follows:

$$\left|\tilde{I}_{n,k} + \hat{I}_{n,k}\right| < \varepsilon_{abs} \qquad \forall \quad n,k \tag{7.31}$$

$$2 \cdot \left|\frac{\tilde{I}_{n,k} + \hat{I}_{n,k}}{\tilde{I}_{n,k} - \hat{I}_{n,k}}\right| < \varepsilon_{rel} \qquad \forall \quad n,k \tag{7.32}$$

where $\tilde{I}_{n,k}$ is the current of the linear circuit partition for node n and frequency k and $\hat{I}_{n,k}$ is the current of the non-linear circuit partition.

7.2.6 Dealing with Exceptions to the Rule

Of course, there are several things that do not fit into the standard harmonic balance scheme. This subsection deals with them and shows how to solve the problems.

1. As the current source vector $\mathbf{I}_S$ is created by multiplying the source voltages and the transadmittance matrix $\tilde{\boldsymbol{Y}}$, a current source needs some extra treatment. The solution is simple: Just add a gyrator to the linear subcircuit and this way, convert the current source to a voltage source.

2. In some circuits the transadmittance between two ports may be infinite, because of a short circuit. The most simple work-around is to add the well-known $1G\Omega$ resistance at these nodes. As the absolute tolerance in the convergence check is usually about 10^{-6}, this method sacrifices no accuracy.

7.3 Speeding up Techniques

The long simulation time is the main problem of harmonic balance simulations. Solving the equation system consumes about 90% of the simulation time, because its complexity is $O(n^3)$. The rest is spent in creating the equation system (mainly matrix multiplication etc). The time used for pre and post AC analysis as well as computing the component models and the FFT can be neglected.

A possibility to reduce the simulation time is to reduce the matrix size, i.e. to reduce the number of harmonic frequencies. Of course, this also reduces the accuracy of the simulation result. Therefore, it is recommended to use a "diamond-shaped" frequency scheme: All frequencies whose mixing-order is above a specific maximum value are omitted in the equation system. This reduces the simulation time quite much but the accuracy slightly only. An even faster method is to start the iteration with a low maximum order (e.g. 4) and increase it to the user-defined value when the tolerances are near the termination criteria.

Further speeding-up techniques are the following:

- Re-use of the Jacobian matrix (Samanskii iterations) together with an update method (e.g. Broyden). The complexity is $O(n^2)$. I.e. use LU decomposition to solve equation 7.12. In the next iteration step use the same decomposition to solve equation 7.12 again, but with the new right-hand side $\mathbf{F}(\mathbf{V}_n)$. This way, the much faster substitution is sufficient. Then apply the result $\Delta\mathbf{V}_n$ together with a modified Broyden update [19]:

$$\mathbf{V}_{n+1} = \mathbf{V}_n - \frac{\Delta\mathbf{V}_n}{1 - \Delta\mathbf{V}_{n-1}^T \cdot \Delta\mathbf{V}_n / \left|\Delta\mathbf{V}_{n-1}\right|^2} \tag{7.33}$$

  with $|\mathbf{v}|$ being the length of the vector $\mathbf{v}$, i.e. the Euclidean norm. The Broyden update can be performed in several consecutive iteration steps.

- Use of iterative matrix solvers, for example Krylov algorithms like Generalized Minimal Residual (GMRES). The complexity is $O(n^2)$ for the standard method and $O(n \cdot \log n)$ for the matrix-implicit method.

Both methods tend to cause convergence problems in some circuits.

7.4 A Symbolic HB Algorithm

In this final section, a harmonic balance algorithm in symbolic language is presented.

Listing 7.1: symbolic HB algorithm

```
 init();                           // separate linear and non-linear devices
 Y = calcTransMatrix();            // transadmittance matrix of linear circuit
 Is = calcSourceCurrent();         // source current of linear subcircuit
 (v, i, q) = calculateDC();        // DC simulation as initial HB estimate
 V = FFT(v);                       // transform voltage into frequency domain
 F = Is + Y*V;                     // linear part of HB equation

do
   JG = mFFT(GJacobian(v));                // create Jacobians and transform...
   JQ = mFFT(QJacobian(v));                // ... them into frequency domain
   J = Y + JG + j*Ω*JQ;                    // calculate overall Jacobian
   V = V - invert(J) * F;                  // Newton Raphson iteration step
   v = IFFT(V);                            // voltage into time domain
   i = nonlinearCurrent(v);                // use component models to get...
   q = nonlinearCharge(v);                 // ... values for next iteration
   I = FFT(i);                             // current into frequency domain
   Q = FFT(q);                             // charge into frequency domain
   F = Is + Y*V + I + j*Ω*Q;               // HB equation
while (abs(F) > Fterm);                    // convergence reached ?

 Va = invert(Ya) * Ia;                     // AC simulation to get all voltages
```

7.5 Large-Signal S-Parameter Simulation

Using harmonic balance techniques, it is also possible to perform an S-parameter simulation in the large-signal regime. This is called LSSP (large-signal s-parameter). Figure 7.3 shows the principle. The port n excites the circuit with the simulation frequency f_0; meanwhile the power of all other ports is set to zero. Having voltage and current of the fundamental frequency f_0 at the ports, the S-parameters can be calculated:

$$\underline{S}_{mn} = \frac{\underline{U}_m(f_0) - \underline{I}_m(f_0) \cdot Z_m}{\underline{U}_n(f_0) + \underline{I}_n(f_0) \cdot Z_n} \cdot \sqrt{\frac{Z_n}{Z_m}} \tag{7.34}$$

$$= \frac{2 \cdot \underline{U}_m(f_0) - \underline{U}_{0,m}}{\underline{U}_{0,n}} \cdot \sqrt{\frac{Z_n}{Z_m}} \tag{7.35}$$

for $m = n$

$$\underline{S}_{nn} = 2 \cdot \frac{\underline{U}_n(f_0)}{\underline{U}_{0,n}} - 1 \tag{7.36}$$

and for $m \neq n$

$$\underline{S}_{mn} = 2 \cdot \frac{\underline{U}_m(f_0)}{\underline{U}_{0,n}} \cdot \sqrt{\frac{Z_n}{Z_m}} \tag{7.37}$$

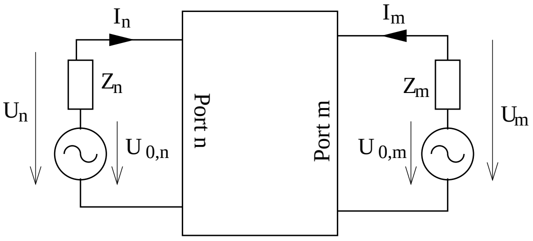


Figure 7.3: S-parameter from AC voltages and currents

An algorithm in symbolic language should describe the whole LSSP:

Listing 7.2: symbolic HB algorithm

```
for n=1 to $NumberOfPorts$ {
  Set power of port $n$ to $P_n$.
  Set power of ports $\neq n$ to 0.
  Perform Harmonic Balance.

  for m=1 to $NumberOfPorts$
    Calculate $\underline{S}_{mn}$ according to above-mentioned equation.
}
```

7.6 Autonomous Harmonic Balance

Up to here, only forced circuits were dealt with. That is, the above-mentioned methods can analyse circuits that are driven by signal sources, but do not create a signal by themselves. The typical examples are amplifiers and mixers. However, harmonic balance techniques are also capable of simulating autonomous circuits like oscillators [20].
The situation here is that the frequencies are not known. This problem can be overcome by the so-called mixed harmonic balanced formulation. This means for the vector V:

- Omit the imaginary part of the element refering to the fundamental frequency, i.e. set it to zero.
- Add the fundamental frequency ω_1 to the unknowns of the equation system.

So, the non-linear equation system of equation 7.2 is re-written to the following one:

$$\mathbf{F}(\mathbf{V},\omega_1) = \underbrace{\boldsymbol{I}_S + \hat{\boldsymbol{Y}}(\omega_1)\cdot\boldsymbol{V}}_{\text{linear}} + \underbrace{j\cdot\boldsymbol{\Omega}(\omega_1)\cdot\boldsymbol{Q}(\mathbf{V}) + \boldsymbol{I}(\mathbf{V})}_{\text{non-linear}} = \mathbf{0} \tag{7.38}$$

The next problem is the fact, that all autonomous circuits have at least one further (unwanted) solution: The DC steady state. One possibility to assure that the simulation converges towards the wanted solution is to use an appropriate starting value like the AC (small-signal) solution (frequency and magnitude). Furthermore, the user should define a node p where the oscillation can be best measured. Then, convergence towards the wanted solution is further forced by normalizing the error function element refering to node p and to the fundamental frequency ω_1 by its corresponding voltage:

$$\overline{\mathbf{F}}_{p,\omega_1}(\mathbf{V},\omega) = \frac{\mathbf{F}_{p,\omega_1}(\mathbf{V},\omega)}{V_p} \tag{7.39}$$

This leads to a partial Kurokawa condition for oscillations [20].

The Jacobian matrix is still quite similar to the one of the forced circuit. For building it the following additional steps have to be performed:

- The row corresponding to node p and to the fundamental frequency has to be divided by V_p (taken from the previous iteration).
- Subtract $\mathbf{F}_{p,\omega_1}/V_p^2$ from the element in the first main diagonal corresponding to node p and to the fundamental frequency.
- Replace the column now corresponding to oscillation frequency by the derivative with respect to the oscillation frequency:

$$\frac{\partial \mathbf{F}(\mathbf{V}, \omega_1)}{\partial \omega_1} = \frac{\partial \hat{\boldsymbol{Y}}(\omega_1)}{\partial \omega_1} \cdot \boldsymbol{V} + j \cdot \frac{\partial \boldsymbol{\Omega}(\omega_1)}{\partial \omega_1} \cdot \boldsymbol{Q}(\mathbf{V}) \tag{7.40}$$

Chapter 8

Harmonic Balance Noise Analysis

Once a harmonic balance simulation is solved a cyclostationary noise analysis can be performed. This results in the sideband noise of each harmonic (including DC, i.e. base band noise). The method described here is based on the principle of small-signal noise. That is, the noise power is assumed small enough (compared to the signal power and its harmonics) to neglect the mixing of the noise terms to each other. This procedure is the standard concept in CAE and allows for a quite simple and time-saving algorithm: Use the Jacobian to calculate a conversion matrix and then apply the noise correlation matrix to it. Two important publications for HB noise simulation exist that were used for the next subsection [21], [22].

Figure 8.1 shows the equivalent circuit for starting the HB noise analysis. At every connection between linear and non-linear subcircuit, there are two noise current sources: one stemming from the linear subcircuit and one stemming from the non-linear subcircuit.

The whole algorithm consists of three parts that all have to be performed for every noise frequency ω_R of interest.

- create the noise of the linear subcircuit
- create the noise of the non-linear subcircuit
- perform noise conversion, i.e. mixing with the signals

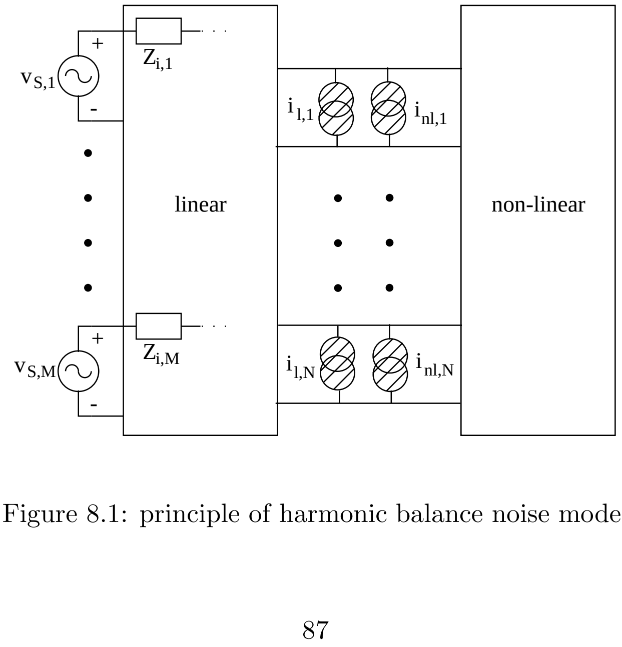


Figure 8.1: principle of harmonic balance noise model

8.1 The Linear Subcircuit

The noise stemming from the linear subcircuit is calculated in two steps:

1. An AC noise analysis (see section 3.2) is performed for the interconnecting nodes of linear and non-linear subcircuit. This results in the noise-voltage correlation matrix $\underline{\boldsymbol{C}}_{Z,lin}$. Note that the MNA matrix was already inverted during the AC analysis of the post-HB algorithm.
2. The matrix $\underline{\boldsymbol{C}}_{Z,lin}$ is converted into a noise-current correlation matrix (see section 5.5.1):

$$\underline{\boldsymbol{C}}_{Y,lin} = \hat{\boldsymbol{Y}} \cdot \underline{\boldsymbol{C}}_{Z,lin} \cdot \hat{\boldsymbol{Y}}^{+} \tag{8.1}$$

   where $\hat{\boldsymbol{Y}}$ is taken from equation 7.2.

Both steps have to be performed at every harmonic signal frequency (e.g. $k \cdot \omega_0 + \omega_R$ for the one-dimensional case). This is illustrated in figure 8.2. The final noise correlation matrix consists of $N \times N$ blocks with $K \times K$ elements each. The $K \times K$-sized blocks are diagonal submatrices, i.e. in the linear subcircuit the noise at different frequencies is uncorrelated.

Remark: If no explicit noise sources exist in the linear subcircuit, $\underline{\boldsymbol{C}}_{Z,lin}$ can be computed much faster by using Bosma's theorem (equation 5.43).

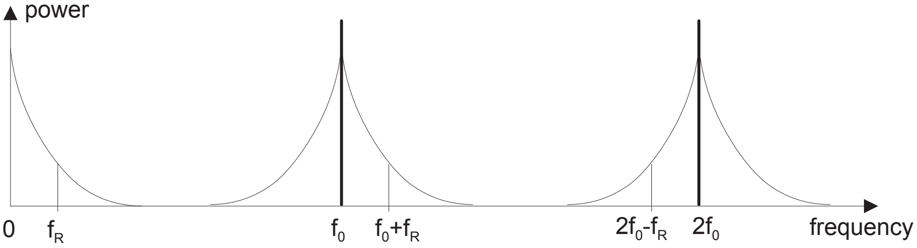


Figure 8.2: harmonics (including DC) with noise sidebands

8.2 The Non-Linear Subcircuit

The noise in the non-linear part of the circuit is calculated by using the quasi-static approach, i.e. for every moment in time the voltages and currents are regarded as a time-dependend bias point. The noise properties of these bias points are used for the noise calculation.

Remark: It is not clear whether this approach creates a valid result for noise with long-time correlation (e.g. 1/f noise), too. But up to now, no other methods were proposed and some publications reported to have achieved reasonable results with this approach and 1/f noise.

For calculating the noise-current correlation matrix $(\underline{C}_{Y,nl})_{N\times N}$ the dependencies of the noise on frequency and on bias needs to be seperated.

1. **dependency on frequency** ($\underline{C}_{Y,DC}$)
   The DC bias point $u_{DC} = u(\omega = 0)$ taken from the result of the HB simulation is chosen for building the frequency dependency. At this bias the noise power-spectral densities of the non-linear components are calculated for each harmonic frequency and are entered into the first main diagonal of the correlation matrix ($\underline{C}_{Y,DC}$). I.e. each block is a $K \times K$ diagonal submatrix with the following elements:

$$C_{Y,DC}(\omega_R) \qquad C_{Y,DC}(\omega_0 + \omega_R) \qquad C_{Y,DC}(2 \cdot \omega_0 + \omega_R) \qquad \ldots \tag{8.2}$$

   where ω_R is the desired noise frequency.

2. **dependency on bias ($\underline{\boldsymbol{M}}$)**
   The noise frequency ω_R is chosen to create the cyclostationary modulation matrix $M(t)$. Its main diagonal contains the current power-spectral density S_i of each time step normalized to its DC bias value:

$$M(t) = \sqrt{\frac{S_i\left(u(t), \omega_R\right)}{S_i(u_{DC}, \omega_R)}} = \sqrt{\frac{S_i\left(u(t), \omega_R\right)}{C_{Y,DC}(\omega_R)}} \tag{8.3}$$

   Note again that this equation only holds if the frequency dependency of S_i is the same for every bias, so that $M(t)$ is frequency independent. Anyway, this demand is fullfilled for most practical models. Now the modulation matrix $M(t)$ can be transformed into frequency domain. This is done by the procedure described in equation 7.18, resulting in the block circulant matrix ($\underline{\boldsymbol{M}}$).

The final step calculates the desired correlation matrix by applying the non-linear modulation to the frequency dependency:

$$(\underline{\boldsymbol{C}}_{\boldsymbol{Y,nl}}) = \underline{\boldsymbol{M}} \cdot (\underline{\boldsymbol{C}}_{\boldsymbol{Y,DC}}) \cdot \underline{\boldsymbol{M}}^{+} \tag{8.4}$$

8.3 Noise Conversion

As the noise of linear and non-linear components are uncorrelated, the noise-voltage correlation matrix at the interconnecting ports can now be calculated:

$$\boldsymbol{C_Z} = \boldsymbol{J}^{-1} \cdot (\boldsymbol{C}_{Y,lin} + \boldsymbol{C}_{Y,nl}) \cdot (\boldsymbol{J}^{-1})^{+} \tag{8.5}$$

here $\boldsymbol{J}^{-1}$ is the inverse of the Jacobian matrix (see equation 7.11) taken from the last HB iteration step (where it already was inverted). Note that it needs to be the precise Jacobian matrix. I.e. it must be taken from an iteration step very close to the solution, without any convergence helpers, and with a precise FFT algorithm (e.g. the multi-dimensional FFT).

Finally, the noise voltages from the interconnecting ports have to be used to compute all other noise voltages. Because the linear subcircuit may be non-reciprocal, its noise must be counted for in a final AC noise analysis, and not in the correlation matrix $\boldsymbol{C_Z}$. The rest is straight forward:

1. Convert the noise-voltage correlation matrix $\boldsymbol{C_Z}$ into the noise-current correlation matrix $\boldsymbol{C_Y}$ (see equation 8.1).
2. Remove the baseband noise (stemming from the linear subcircuit) from the noise correlation matrix $\boldsymbol{C_Y}$ in order to not double-count it in the following analysis.
3. Create the noise-current correlation matrix for the linear subcircuit and insert the elements of the matrix $\boldsymbol{C_Y}$.
4. Perform an AC noise analysis for all nodes of interest.

8.4 Phase and Amplitude Noise

The harmonic balance noise analysis calculates the noise power spectral density $S_{uu,k}(\omega_R)$ at the noise frequency ω_R of the k-th harmonic. The SSB phase and amplitude noise normalized to the carrier can be obtained by using the symmetry between positive and negative harmonic numbers:

$$\left\langle \Phi_k \Phi_{-k}^* \right\rangle = \frac{S_{uu,k} + S_{uu,-k} - 2 \cdot \mathrm{Re}\left(C_{Z,k,-k} \cdot \exp(-j \cdot 2 \cdot \phi_k)\right)}{|U_k|^2} \tag{8.6}$$

$$\left\langle A_k A_{-k}^* \right\rangle = \frac{S_{uu,k} + S_{uu,-k} + 2 \cdot \mathrm{Re}\left(C_{Z,k,-k} \cdot \exp(-j \cdot 2 \cdot \phi_k)\right)}{|U_k|^2} \tag{8.7}$$

with $U_k = |U_k| \cdot \exp(j \cdot \phi_k)$ being the k-th harmonic and $C_{Z,k,-k}$ being the correlation coefficient between the sideband noise of positive and negative harmonics.

Chapter 9

Circuit Optimization

As circuits get more and more complex, it soon becomes very hard or even impossible to understand all of its physical background. In this case a circuit optimizer is the best choice to reach a good performance. The user defines

- the circuit and the type of simulation
- variables, i.e. circuit parameters that can be varied
- goals, i.e. function values that are computed from the simulation result and that should be maximized, minimized etc.

Now the simulation engine searches for the parameter values that fit the goals most closely. Obviously, optimization is a very powerful and universal tool. That's why a lot of different algorithms exist. The most common are described in the next sections.

9.1 Grid Search

This method creates a mesh from the parameter space that consists of a user-defined number of test points. The function value of every point is calculated and the best one is taken as a result. Of course, this is a dumb algorithm that will succeed only if the number of test points is large and the number of parameters is small.
In order to get a finer grid, a restart may be performed. I.e. the neighbourhood of best parameter vector is used for a new grid search. This way the optimum can be approximated much better, because the parameter space is much smaller.

9.2 Random Method

A random optimization is perhaps the most simple algorithm. The variables are varied randomly within the user defined limits and the values with the best result are saved. This method is able to find the global minimum, but usually it needs a huge number of simulation runs. Several strategies exists. A popular one is the following:

1. Choose a starting point.
2. Update the current variables by adding (normal distributed) random numbers.
3. If the new one gives better function values, take it as new current vector.
4. Repeat from step 2 until the user-defined number of iterations is reached.

9.3 Coordinate Descent

Coordinate descent (or coordinate search) is a quite simple optimization algorithm. It works a s follows:

1. Choose a starting point.
2. Search along the first variable until a minimum is found (line search).
3. Do this step-by-step with all other variables.
4. Repeat from step 2 until the improvement is very small.

This method approximately follows the gradient of the function values, this way it finds a local minimum near the starting point.

9.4 Hooke-Jeeves Method

The optimization method by Hooke and Jeeves tries to reduce the number of simulation runs by performing two different steps. First an exploratory move tries to detect the direction where the goals gain improvements. Then a pattern move is performed that tries to further improve the result by varying the variables towards the same direction. This gives the following procedure:

- exploratory move
    - Increment the first variable and check the goals.
    - If there is no improvement, decrement this variable.
    - If there is still no improvement, keep the old value.
    - Perform these steps with all variables one-by-one.
    - If no variable was changed, repeat from beginning with a smaller step size.
    - If successful, two variable vectors are obtained:
        1. $\vec{v}_k$ contains the original variable values.
        2. $\vec{v}_{k+1}$ contains the improved variable values.
- pattern move
    - Get new variable values $\vec{v}_{k+2} = \vec{v}_{k+1} + \alpha \cdot (\vec{v}_{k+1} - \vec{v}_k)$
      Usually $\alpha = 1$, but larger values may be chosen.
    - If there is no improvement, revert to the old variable values $\vec{v}_{k+1}$.
    - Continue with a new exploratory move.

9.5 Nelder-Mead Method

The Nelder-Mead method or downhill simplex method is a good choice for large optimization problems. It is capable of finding the global minimum even for non-smooth functions containing noise.
With the number n of variables to be varied, the algorithm starts with $n + 1$ vertices (parameter vectors $\vec{v}$) that span an $n-$dimensional body. By checking the function values $f(\vec{v})$ (goals) of these vertices, the volume of this body is reduced step-by-step. This is done by four different operations (reflection with $R = 1$, expansion with $E = 2$, contraction with $K = 0.5$ and shrinkage with $S = 0.5$) in the following way:

1. Sort the vertices by its function values: $f(\vec{v}_1) < f(\vec{v}_2) < \ldots < f(\vec{v}_{n+1})$
   i.e. $f(\vec{v}_1)$ is the best value and $\vec{v}_{n+1}$ is the worst value.

2. Calculate the average value of all but the worst vector: $\vec{v}_m = \frac{1}{n} \cdot \sum_{i=1}^{n} \vec{v}_i$

3. Reflection: $\vec{v}_r = \vec{v}_m + R \cdot (\vec{v}_m - \vec{v}_{n+1})$
   If $f(\vec{v}_1) < f(\vec{v}_r) < f(\vec{v}_n)$ replace the worst value with the new vertex and start the next iteration, i.e. go back to step 1.

4. If $f(\vec{v}_r) < f(\vec{v}_1)$, then expand the new value:
   $\vec{v}_e = \vec{v}_m + E \cdot (\vec{v}_r - \vec{v}_m)$
   If $f(\vec{v}_e) < f(\vec{v}_r)$ replace the worst value with the expanded vertex, otherwise replace it with the reflected vertex. Then start the next iteration.

5. Inside contraction: If $f(\vec{v}_r) > f(\vec{v}_{n+1})$ then $\vec{v}_{ic} = \vec{v}_m + K \cdot (\vec{v}_{n+1} - \vec{v}_m)$
   If $f(\vec{v}_{ic}) < f(\vec{v}_{n+1})$, replace the worst value with the new vertex and start the next iteration.

6. Outside contraction: If $f(\vec{v}_n) < f(\vec{v}_r) < f(\vec{v}_{n+1})$ then $\vec{v}_{oc} = \vec{v}_m + K \cdot (\vec{v}_r - \vec{v}_m)$
   If $f(\vec{v}_{oc}) < f(\vec{v}_r)$, replace the worst value with the new vertex and start the next iteration.

7. Shrinkage: Replace all vertices except the best one by
   $\vec{v}_i = \vec{v}_i - S \cdot (\vec{v}_i - \vec{v}_1)$ for $i = 2, \ldots, n+1$
   Then start the next iteration.

The iteration process ends

- if the largest difference of adjacent vertices becomes less than a user defined value ϵ:

$$\max \frac{|\vec{v}_i - \vec{v}_{i+1}|}{|\vec{v}_i + \vec{v}_{i+1}|} < \epsilon \tag{9.1}$$

- and if the difference between the best and the worst function value becomes less than a user defined value ϵ:

$$\max \left| \frac{f(\vec{v}_1) - f(\vec{v}_{n+1})}{f(\vec{v}_1) + f(\vec{v}_{n+1})} \right| < \epsilon \tag{9.2}$$

9.6 Simulated Annealing

Simulated annealing (SA) is a heuristic method that emulates the statistical process of annealing in solids, i.e. slowly cooling down until the material freezes into a crystal.
The algorithm starts with a user-defined initial parameter vector. In every iteration a randomly chosen vector close to the current vector is evaluated. If the new vector gives a better function value than the current one, it replaces the current one and the next iteration starts. If the new vector gives a worse function value, it replaces the current one with a probability of

$$p = \exp\left(\frac{f_{old} - f_{new}}{T}\right) \tag{9.3}$$

with f_{old} , f_{new} being the function values of the current and the new vector, respectively, and T is the current temperature. The annealing is performed by reducing T each time a fix number of iterations are finished.

9.7 Modified controlled random search

The modified controlled random search (MCRS) is a heuristic method [23]. It starts with a number of parameter vectors (called population), that are randomly distributed over the whole parameter domain. In every iteration the algorithm randomly chooses a vertex out of the population and performs a reflection:

$$\vec{v}_r = \vec{v}_m + R \cdot (\vec{v}_x - \vec{v}_m) \tag{9.4}$$

$$\text{with} \qquad \vec{v}_m = \frac{1}{n} \cdot \sum_{i=1}^{n} \vec{v}_i \tag{9.5}$$

with $\vec{v}_x$ being an arbitrary vector from the vertex, $\vec{v}_m$ being the centroid of the vertex and R being a uniformly distributed random number between zero and α. The new vector $\vec{v}_r$ replaces the worst population vector, if its function value is better.

9.8 Differential Evolution

Differential evolution (DE) is known as one of the most powerful optimization algorithm for large and complex problems [24]. It starts with a number of parameter vectors (called population) that should be distributed over the whole parameter domain, i.e. for every parameter in every vector it is:

$$v_i = v_{i,min} + \text{rand(0,1)}_i \cdot (v_{i,max} - v_{i,min}) \tag{9.6}$$

with rand(0,1) being a uniformly distributed random number between 0 and 1, and $v_{i,min}$, $v_{i,max}$ are the minimum and maximum values for the i-th parameter. Good experiences have been made with a population size that is ten times the number of parameters.

During the optimization DE creates a new generation of the population by performing mutation, crossover and selection processes on every parameter vector. The most popular mutation strategies are as follows:

DE/rand/1	$\vec{V}_j = \vec{X}_{r1} + F \cdot (\vec{X}_{r2} - \vec{X}_{r3})$
DE/current/1	$\vec{V}_j = \vec{X}_j + F \cdot (\vec{X}_{r1} - \vec{X}_{r2})$
DE/best/1	$\vec{V}_j = \vec{X}_{best} + F \cdot (\vec{X}_{r1} - \vec{X}_{r2})$
DE/best/2	$\vec{V}_j = \vec{X}_{best} + F \cdot (\vec{X}_{r1} - \vec{X}_{r2} + \vec{X}_{r3} - \vec{X}_{r4})$
DE/rand-to-best/1	$\vec{V}_j = \vec{X}_{r1} + F \cdot (\vec{X}_{best} - \vec{X}_{r1} + \vec{X}_{r2} - \vec{X}_{r3})$
DE/pbest/1	$\vec{V}_j = \vec{X}_{pbest} + F \cdot (\vec{X}_{r1} - \vec{X}_{r2})$
DE/current-to-pbest/1	$\vec{V}_j = \vec{X}_j + F \cdot (\vec{X}_{pbest} - \vec{X}_j + \vec{X}_{r2} - \vec{X}_{r3})$
DE/best-of-rand/2	$\vec{V}_j = \vec{X}_{br} + F \cdot (\vec{X}_{r1} - \vec{X}_{r2} + \vec{X}_{r3} - \vec{X}_{r4})$

with the vector $\vec{X}_j$ being the j-th vector of the next generation, $\vec{X}_{best}$ being the best vector of the current generation, $\vec{X}_{pbest}$ being randomly chosen out of the M best vectors of the current generation (with M usually about 8), $\vec{X}_{r1}$ to $\vec{X}_{r4}$ being randomly chosen vectors of the current generation, and $\vec{X}_{br}$ being the best of the randomly chosen vectors.

DE/rand and DE/best are the traditional variants. As can be easily understood, DE/rand converges slowly but stable to the optimum. Therefore, it's a good choice for hard problems. DE/best shows the opposite behaviour. It converges fast, but tends to find local optimums instead of the global one. DE/rand-to-best, DE/pbest and DE/best-of-rand try to combine the advantages of the traditional variants. DE/best-of-rand/2 seems to be the most successful among them, but as usually this also depend on the problem.

In order to increase the diversity, the current vector $\vec{X}$ and the mutated vector $\vec{V}$ are mixed by the crossover operation. The binominal (bin) crossover is the most popular one:

$$u_i = \begin{cases} v_i & \text{if } \text{rand(0,1)}_i \leq Cr \quad \text{or} \quad i = i_{rand} \\ x_i & \text{otherwise} \end{cases} \tag{9.7}$$

with Cr being the user-defined crossover probability and i_{rand} being a randomly chosen number that makes sure that at least one crossover happens. The exponential (exp) crossover is a further method, but it usually gives worse results.

If the new vector contains parameters outside the user-defined limits, it's not recommended to set them to these limits, because the evolution process may easily get caught on them. A better way is to mirror the value on this limit:

$$u_{i,mirrored} = 2 \cdot x_{i,max} - u_i \qquad \text{if} \quad u_i > x_{i,max} \tag{9.8}$$

$$u_{i,mirrored} = 2 \cdot x_{i,min} - u_i \qquad \text{if} \quad u_i < x_{i,min} \tag{9.9}$$

Finally the selection process determines which parameter vector survives, the current one or the new generated one. I.e. the vector with the better function value is inserted into the next generation.

The whole differential evolution process can be summarized as follows:

1. Create the first population $\vec{X}_{i,1}$ randomly.
2. Perform mutation, i.e. create donor vector $\vec{V}_{i,G}$
3. Perform crossover, i.e. create trial vector $\vec{U}_{i,G}$
4. Perform selection, i.e. $\vec{X}_{i,G+1} = \begin{cases} \vec{U}_{i,G} & \text{if} \quad f(\vec{U}_{i,G}) \leq f(\vec{X}_{i,G}) \\ \vec{X}_{i,G} & \text{otherwise} \end{cases}$
5. Perform step 2 to 4 for all vectors of the population ($i = 1 \ldots N_P$).
6. Repeat from step 2 until wanted number of runs are done.

9.9 DEGL/SAW

Much work has be spent on improving the differential evolution. The most powerful attempt created an algorithm called Differential Evolution with Global and Local neighborhood-based mutations and Self Adaptive Weight factors (DEGL/SAW) [25]. It uses the mutation scheme DE/current-to-best/1/bin and performs it twice for each vector. First by choosing $\vec{X}_{r1,G}$, $\vec{X}_{r2,G}$ and $\vec{X}_{best,G}$ out of the whole population (global) and second by choosing them out of the neighborhood of the current vector (local). The neighborhood of vector $\vec{X}_{i,G}$ consists of the vectors with index $((i-n) \quad \text{modulo} \quad N_P)$ to index $((i+n) \quad \text{modulo} \quad N_P)$. Finally, the results of global mutation $\vec{G}_{i,G}$ and local mutation $\vec{L}_{i,G}$ are mixed together by a weighting factor w_i:

$$\vec{V}_{i,G} = w_i \cdot \vec{G}_{i,G} + (1 - w_i) \cdot \vec{L}_{i,G} \tag{9.10}$$

Each vector $\vec{X}_{i,G}$ has its own weighting factor w_i. All factors together build a population that evolves in the same way as the parameter vectors. I.e. at the beginning they are choosen randomly and for each generation the mutation scheme DE/current-to-best/1/no-crossover is performed. The values are limited to 0.05 to 0.95.

The following crossover and selection process equal the ones of the standard DE algorithm. The size of the neighborhood is not critical and usually chosen to be a tenth of the whole population.

Chapter 10

Linear devices

As the MNA matrix is the y-parameter matrix of the whole circuit, components that are defined by y-parameters can be easily inserted by adding these parameters to the MNA matrix elements (so-called 'stamping'), i.e. for a two-port:

$$\begin{bmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{10.1}$$

The same is true for a noise model defined with current sources, i.e. the elements of the noise current correlation matrix have simply to be added to the overall current correlation matrix.

In some cases the ground-referenced device needs to be extended to a device without ground reference. That is, the first port now has got two terminals (1 and 2) and the second port has got terminal 3 and 4:

$$\begin{bmatrix} Y_{11} & -Y_{11} & Y_{12} & -Y_{12} \\ -Y_{11} & Y_{11} & -Y_{12} & Y_{12} \\ Y_{21} & -Y_{21} & Y_{22} & -Y_{22} \\ -Y_{21} & Y_{21} & -Y_{22} & Y_{22} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.2}$$

Components that cannot be defined by y-parameters need to add additional columns and rows to the MNA matrix. Components defined by z-parameters can be added in the following way (example for a 2-port; I_{in} and I_{out} flow out of the component ports). It is easily extendable for any port number.

$$\begin{bmatrix} . & . & 1 & 0 \\ . & . & 0 & 1 \\ -1 & 0 & Z_{11} & Z_{12} \\ 0 & -1 & Z_{21} & Z_{22} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{in} \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.3}$$

For z-parmeters without ground reference, this yields:

$$\begin{bmatrix} . & . & . & . & 1 & 0 \\ . & . & . & . & -1 & 0 \\ . & . & . & . & 0 & 1 \\ . & . & . & . & 0 & -1 \\ -1 & +1 & 0 & 0 & Z_{11} & Z_{12} \\ 0 & 0 & -1 & +1 & Z_{21} & Z_{22} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_{in} \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.4}$$

A 2-port component defined by h-parameters can be added in the following way to the MNA matrix:

$$\begin{bmatrix} . & . & 1 \\ . & H_{22} & H_{21} \\ -1 & H_{12} & H_{11} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ J_1 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \end{bmatrix} \tag{10.5}$$

A 2-port component defined by g-parameters can be added in the following way to the MNA matrix:

$$\begin{bmatrix} G_{11} & . & G_{12} \\ . & . & 1 \\ G_{21} & -1 & G_{22} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ J_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \end{bmatrix} \tag{10.6}$$

A 2-port component defined by chain parameters (ABCD parameters) can be added in the following way to the MNA matrix:

$$\begin{bmatrix} . & A_{21} & A_{22} \\ . & . & -1 \\ -1 & A_{11} & A_{12} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ J_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \end{bmatrix} \tag{10.7}$$

Components that are characterized by S-parameters (normalized to Z_0) can be put into the MNA matrix and the noise current correlation matrix by the following scheme (example for a 3-port). It is easily extendable for any port number.

$$\begin{bmatrix} . & . & . & 1 & 0 & 0 \\ . & . & . & 0 & 1 & 0 \\ . & . & . & 0 & 0 & 1 \\ S_{11}-1 & S_{12} & S_{13} & Z_0 \cdot (S_{11}+1) & Z_0 \cdot S_{12} & Z_0 \cdot S_{13} \\ S_{21} & S_{22}-1 & S_{23} & Z_0 \cdot S_{21} & Z_0 \cdot (S_{22}+1) & Z_0 \cdot S_{23} \\ S_{31} & S_{32} & S_{33}-1 & Z_0 \cdot S_{31} & Z_0 \cdot S_{32} & Z_0 \cdot (S_{33}+1) \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ I_{I1} \\ I_{I2} \\ I_{I3} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.8}$$

$$\underline{C}_Y = 4 \cdot Z_0 \cdot \begin{bmatrix} . & . & . & 0 & 0 & 0 \\ . & . & . & 0 & 0 & 0 \\ . & . & . & 0 & 0 & 0 \\ 0 & 0 & 0 & c_{S,11} & c_{S,12} & c_{S,13} \\ 0 & 0 & 0 & c_{S,21} & c_{S,22} & c_{S,23} \\ 0 & 0 & 0 & c_{S,31} & c_{S,32} & c_{S,33} \end{bmatrix} \tag{10.9}$$

A 2-port component defined by T-parameters can be added in the following way to the MNA matrix:

$$\begin{bmatrix} . & . & 1 & 0 \\ . & . & 0 & 1 \\ -1 & T_{11}+T_{12} & +Z_0 & Z_0 \cdot (T_{11}-T_{12}) \\ -1 & T_{21}+T_{22} & -Z_0 & Z_0 \cdot (T_{21}-T_{22}) \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ J_1 \\ J_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \\ 0 \end{bmatrix} \tag{10.10}$$

10.1 Resistor

For DC and AC simulation an ideal resistor with resistance R yields:

$$Y = \frac{1}{R} \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{10.11}$$

The noise correlation matrix at temperature T yields:

$$(\underline{C}_Y) = \frac{4 \cdot k \cdot T}{R} \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{10.12}$$

The scattering parameters normalized to impedance Z_0 writes as follows.

$$S_{11} = S_{22} = \frac{R}{2 \cdot Z_0 + R} \tag{10.13}$$

$$S_{12} = S_{21} = 1 - S_{11} = \frac{2 \cdot Z_0}{2 \cdot Z_0 + R} \tag{10.14}$$

Being on temperature T, the noise wave correlation matrix writes as follows.

$$(\underline{C}) = k \cdot T \cdot \frac{4 \cdot R \cdot Z_0}{(2 \cdot Z_0 + R)^2} \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{10.15}$$

The noise wave correlation matrix of a parallel resistor with resistance R writes as follows.

$$(\underline{C}) = k \cdot T \cdot \frac{4 \cdot R \cdot Z_0}{(2 \cdot R + Z_0)^2} \cdot \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \tag{10.16}$$

The noise wave correlation matrix of a grounded resistor with resistance R is a matrix consisting of one element and writes as follows.

$$(\underline{C}) = k \cdot T \cdot \frac{4 \cdot R \cdot Z_0}{(R + Z_0)^2} \tag{10.17}$$

10.2 Capacitor

During DC simulation the capacitor is an open circuit. Thus, its MNA entries are all zero.

During AC simulation the y-parameter matrix of an ideal capacitor with the capacitance C writes as follows.

$$Y = \begin{pmatrix} +j\omega C & -j\omega C \\ -j\omega C & +j\omega C \end{pmatrix} \tag{10.18}$$

The scattering parameters (normalized to Z_0) of an ideal capacitor with capacitance C writes as follows.

$$S_{11} = S_{22} = \frac{1}{2 \cdot Z_0 \cdot j\omega C + 1} \tag{10.19}$$

$$S_{12} = S_{21} = 1 - S_{11} \tag{10.20}$$

An ideal capacitor is noise free. Its noise correlation matrices are, therefore, zero.

10.3 Inductor

During DC simulation an inductor is a short circuit, thus, its MNA matrix entries need an additional row and column.

$$\begin{bmatrix} . & . & +1 \\ . & . & -1 \\ +1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{br} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.21}$$

During AC simulation the Y-parameter matrix of an ideal inductor with the inductance L writes as follows.

$$Y = \begin{pmatrix} +\frac{1}{j\omega L} & -\frac{1}{j\omega L} \\ -\frac{1}{j\omega L} & +\frac{1}{j\omega L} \end{pmatrix} \tag{10.22}$$

The scattering parameters of an ideal inductor with inductance L writes as follows.

$$S_{11} = S_{22} = \frac{j\omega L}{2 \cdot Z_0 + j\omega L} \tag{10.23}$$

$$S_{12} = S_{21} = 1 - S_{11} \tag{10.24}$$

An ideal inductor is noise free.

10.4 DC Block

A DC block is a capacitor with an infinite capacitance. During DC simulation the DC block is an open circuit. Thus, its MNA entries are all zero.

The MNA matrix entries of a DC block correspond to an ideal short circuit during AC analysis which is modeled by a voltage source with zero voltage.

$$\begin{bmatrix} . & . & +1 \\ . & . & -1 \\ +1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{br} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.25}$$

The scattering parameters writes as follows.

$$(S) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{10.26}$$

A DC block is noise free. A model for transient simulation does not exist. It is common practice to model it as a capacitor with finite capacitance whose value is entered by the user.

10.5 DC Feed

A DC feed is an inductor with an infinite inductance. The MNA matrix entries of a DC feed correspond to an ideal short circuit during DC analysis:

$$\begin{bmatrix} . & . & +1 \\ . & . & -1 \\ +1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{br} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.27}$$

During AC simulation the DC feed is an open circuit. Thus, its MNA entries are all zero.

The scattering parameters writes as follows.

$$(S) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{10.28}$$

A DC feed is noise free. A model for transient simulation does not exist. It is common practice to model it as an inductor with finite inductance whose value is entered by the user.

10.6 Bias T

An ideal bias t is a combination of a DC block and a DC feed (fig. 10.1). During DC simulation the MNA matrix of an ideal bias t writes as follows:

$$\begin{bmatrix} . & . & . & 0 \\ . & . & . & 1 \\ . & . & . & -1 \\ 0 & 1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.29}$$

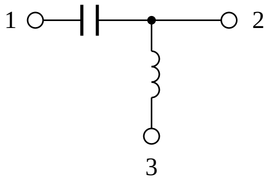


Figure 10.1: bias t

The MNA entries of the bias t during AC analysis write as follows.

$$\begin{bmatrix} . & . & . & -1 \\ . & . & . & 1 \\ . & . & . & 0 \\ -1 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.30}$$

The scattering parameters writes as follows.

$$(S) = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{10.31}$$

A bias t is noise free. A model for transient simulation does not exist. It is common practice to model it as an inductor and a capacitance with finite values which are entered by the user.

10.7 Transformer

The two winding ideal transformer, as shown in fig. 10.2, is determined by the following equation which introduces one more unknown in the MNA matrix.

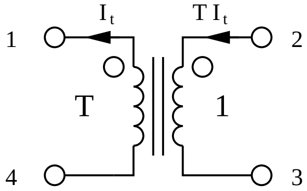


Figure 10.2: ideal two winding transformer

$$T \cdot (V_2 - V_3) = V_1 - V_4 \quad \rightarrow \quad V_1 - T \cdot V_2 + T \cdot V_3 - V_4 = 0 \tag{10.32}$$

The new unknown variable I_t must be considered by the four remaining simple equations.

$$I_1 = -I_t \quad I_2 = T \cdot I_t \quad I_3 = -T \cdot I_t \quad I_4 = I_t \tag{10.33}$$

And in matrix representation this is for DC and for AC simulation:

$$\begin{bmatrix} . & . & . & . & -1 \\ . & . & . & . & T \\ . & . & . & . & -T \\ . & . & . & . & 1 \\ 1 & -T & T & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_t \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.34}$$

It is noticeable that the additional row (part of the C matrix) and the corresponding column (part of the B matrix) are transposed to each other. When considering the turns ratio T being complex introducing an additional phase the transformer can be used as phase-shifting transformer. Both the vectors must be conjugated complex transposed in this case.

Using the port numbers depicted in fig. 10.2, the scattering parameters of an ideal transformer with voltage transformation ratio T (number of turns) writes as follows.

$$S_{14} = S_{22} = S_{33} = S_{41} = \frac{1}{T^2+1} \tag{10.35}$$

$$S_{12} = -S_{13} = S_{21} = -S_{24} = -S_{31} = S_{34} = -S_{42} = S_{43} = T \cdot S_{22} \tag{10.36}$$

$$S_{11} = S_{23} = S_{32} = S_{44} = T \cdot S_{12} \tag{10.37}$$

An ideal transformer is noise free.

10.8 Symmetrical transformer

The ideal symmetrical transformer, as shown in fig. 10.3, is determined by the following equations which introduce two more unknowns in the MNA matrix.

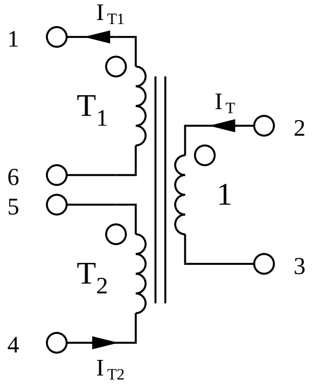


Figure 10.3: ideal three winding transformer

$$T_1 \cdot (V_2 - V_3) = V_1 - V_6 \quad \rightarrow \quad V_1 - T_1 \cdot V_2 + T_1 \cdot V_3 - V_6 = 0 \tag{10.38}$$

$$T_2 \cdot (V_2 - V_3) = V_5 - V_4 \quad \rightarrow \quad -T_2 \cdot V_2 + T_2 \cdot V_3 - V_4 + V_5 = 0 \tag{10.39}$$

The new unknown variables I_{T1} and I_{T2} must be considered by the six remaining simple equations.

$$I_2 = T_1 \cdot I_{T1} + T_2 \cdot I_{T2} \quad I_3 = -T_1 \cdot I_{T1} - T_2 \cdot I_{T2} \tag{10.40}$$

$$I_1 = -I_{T1} \quad I_4 = I_{T2} \quad I_5 = -I_{T2} \quad I_6 = I_{T1} \tag{10.41}$$

The matrix representation needs to be augmented by two more new rows and their corresponding columns. For DC and AC simulation it is:

$$\begin{bmatrix} . & . & . & . & . & . & -1 & 0 \\ . & . & . & . & . & . & T_1 & T_2 \\ . & . & . & . & . & . & -T_1 & -T_2 \\ . & . & . & . & . & . & 0 & 1 \\ . & . & . & . & . & . & 0 & -1 \\ . & . & . & . & . & . & 1 & 0 \\ 1 & -T_1 & T_1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -T_2 & T_2 & -1 & 1 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ V_5 \\ V_6 \\ I_{T1} \\ I_{T2} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \\ I_6 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.42}$$

Using the port numbers depicted in fig. 10.3, the scattering parameters of an ideal, symmetrical transformer with voltage transformation ratio (number of turns) T_1 and T_2, respectively, writes as follows.

$$denom = 1 + T_1^2 + T_2^2 \tag{10.43}$$

$$S_{11} = S_{66} = \frac{T_1^2}{denom} \qquad S_{16} = S_{61} = 1 - S_{11} \tag{10.44}$$

$$S_{44} = S_{55} = \frac{T_2^2}{denom} \qquad S_{45} = S_{54} = 1 - S_{44} \tag{10.45}$$

$$S_{22} = S_{33} = \frac{1}{denom} \qquad S_{23} = S_{32} = 1 - S_{22} \tag{10.46}$$

$$S_{12} = S_{21} = -S_{13} = -S_{31} = -S_{26} = -S_{62} = S_{36} = S_{63} = \frac{T_1}{denom} \tag{10.47}$$

$$-S_{24} = -S_{42} = S_{25} = S_{52} = S_{34} = S_{43} = -S_{35} = -S_{53} = \frac{T_2}{denom} \tag{10.48}$$

$$-S_{14} = -S_{41} = S_{15} = S_{51} = S_{46} = S_{64} = -S_{56} = -S_{65} = \frac{T_1 \cdot T_2}{denom} \tag{10.49}$$

An ideal symmetrical transformer is noise free.

10.9 Non-ideal transformer

Many simulators support non-ideal transformers (e.g. mutual inductor in SPICE). An often used model consists of finite inductances and an imperfect coupling (straw inductance). This model has three parameters: Inductance of the primary coil L_1, inductance of the secondary coil L_2 and the coupling factor $k = 0...1$.

This model can be replaced by the equivalent circuit depicted in figure 10.4. The values are calculated as follows.

$$\text{turn ratio:} \qquad T = \sqrt{\frac{L_1}{L_2}} \tag{10.50}$$

$$\text{mutual inductance:} \qquad M = k \cdot L_1 \tag{10.51}$$

$$\text{primary inductance:} \qquad L_{1,new} = L_1 - M = L_1 \cdot (1 - k) \tag{10.52}$$

$$\text{secondary inductance:} \qquad L_{2,new} = L_2 - \frac{M}{T^2} = L_2 \cdot (1 - k) \tag{10.53}$$

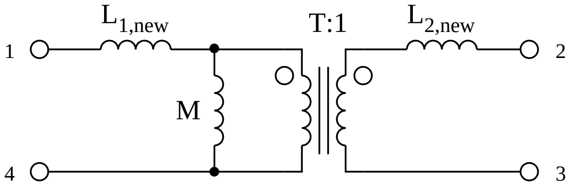


Figure 10.4: equivalent circuit of non-ideal transformer

The Y-parameters of this component are:

$$Y_{11} = Y_{44} = -Y_{41} = -Y_{14} = \frac{1}{j\omega \cdot L_1 \cdot (1-k^2)} \tag{10.54}$$

$$Y_{22} = Y_{33} = -Y_{23} = -Y_{32} = \frac{1}{j\omega \cdot L_2 \cdot (1-k^2)} \tag{10.55}$$

$$Y_{13} = Y_{31} = Y_{24} = Y_{42} = -Y_{12} = -Y_{21} = -Y_{34} = -Y_{43} = \frac{k}{j\omega \cdot \sqrt{L_1 \cdot L_2} \cdot (1-k^2)} \tag{10.56}$$

Furthermore, its S-parameters are:

$$D = (k^2-1) \cdot \frac{\omega^2 \cdot L_1 \cdot L_2}{2 \cdot Z_0} + j\omega L_1 + j\omega L_2 + 2 \cdot Z_0 \tag{10.57}$$

$$S_{14} = S_{41} = \frac{j\omega L_2 + 2 \cdot Z_0}{D} \tag{10.58}$$

$$S_{11} = S_{44} = 1 - S_{14} \tag{10.59}$$

$$S_{23} = S_{32} = \frac{j\omega L_1 + 2 \cdot Z_0}{D} \tag{10.60}$$

$$S_{22} = S_{33} = 1 - S_{23} \tag{10.61}$$

$$S_{12} = -S_{13} = S_{21} = -S_{24} = -S_{31} = S_{34} = -S_{42} = S_{43} = \frac{j\omega \cdot k \cdot \sqrt{L_1 \cdot L_2}}{D} \tag{10.62}$$

Also including an ohmic resistance R_1 and R_2 for each coil, leads to the following Y-parameters:

$$Y_{11} = Y_{44} = -Y_{41} = -Y_{14} = \frac{1}{j\omega \cdot L_1 \cdot \left(1 - k^2 \cdot \dfrac{j\omega L_2}{j\omega L_2 + R_2}\right) + R_1} \tag{10.63}$$

$$Y_{22} = Y_{33} = -Y_{23} = -Y_{32} = \frac{1}{j\omega \cdot L_2 \cdot \left(1 - k^2 \cdot \dfrac{j\omega L_1}{j\omega L_1 + R_1}\right) + R_2} \tag{10.64}$$

$$Y_{13} = Y_{31} = Y_{24} = Y_{42} = -Y_{12} = -Y_{21} = -Y_{34} = -Y_{43} = k \cdot \frac{j\omega\sqrt{L_1 \cdot L_2}}{j\omega \cdot L_2 + R_2} \cdot Y_{11} \tag{10.65}$$

Building the S-parameters leads to too large equations. Numerically converting the Y-parameters into S-parameters is therefore recommended.

The MNA matrix entries during DC analysis and the noise correlation matrices of this transformer are:

$$(\underline{Y}) = \begin{pmatrix} 1/R_1 & 0 & 0 & -1/R_1 \\ 0 & 1/R_2 & -1/R_2 & 0 \\ 0 & -1/R_2 & 1/R_2 & 0 \\ -1/R_1 & 0 & 0 & 1/R_1 \end{pmatrix} \tag{10.66}$$

$$(\underline{C}_Y) = 4 \cdot k \cdot T \cdot \begin{pmatrix} 1/R_1 & 0 & 0 & -1/R_1 \\ 0 & 1/R_2 & -1/R_2 & 0 \\ 0 & -1/R_2 & 1/R_2 & 0 \\ -1/R_1 & 0 & 0 & 1/R_1 \end{pmatrix} \tag{10.67}$$

$$(\underline{C}_S) = 4 \cdot k \cdot T \cdot Z_0 \cdot \begin{pmatrix} \frac{R_1}{(2\cdot Z_0+R_1)^2} & 0 & 0 & -\frac{R_1}{(2\cdot Z_0+R_1)^2} \\ 0 & \frac{R_2}{(2\cdot Z_0+R_2)^2} & -\frac{R_2}{(2\cdot Z_0+R_2)^2} & 0 \\ 0 & -\frac{R_2}{(2\cdot Z_0+R_2)^2} & \frac{R_2}{(2\cdot Z_0+R_2)^2} & 0 \\ -\frac{R_1}{(2\cdot Z_0+R_1)^2} & 0 & 0 & \frac{R_1}{(2\cdot Z_0+R_1)^2} \end{pmatrix} \tag{10.68}$$

A transformer with three coupled inductors has three coupling factors k_{12}, k_{13} and k_{23}. Its Y-parameters write as follows (port numbers are according to figure 10.3).

$$A = j\omega \cdot (1 - k_{12}^2 - k_{13}^2 - k_{23}^2 + 2 \cdot k_{12} \cdot k_{13} \cdot k_{23}) \tag{10.69}$$

$$Y_{11} = Y_{66} = -Y_{16} = -Y_{61} = \frac{1 - k_{23}^2}{L_1 \cdot A} \tag{10.70}$$

$$Y_{22} = Y_{33} = -Y_{23} = -Y_{32} = \frac{1 - k_{12}^2}{L_3 \cdot A} \tag{10.71}$$

$$Y_{44} = Y_{55} = -Y_{45} = -Y_{54} = \frac{1 - k_{13}^2}{L_2 \cdot A} \tag{10.72}$$

$$Y_{12} = Y_{21} = Y_{36} = Y_{63} = -Y_{13} = -Y_{31} = -Y_{26} = -Y_{62} = \frac{k_{12} \cdot k_{23} - k_{13}}{\sqrt{L_1 \cdot L_3} \cdot A} \tag{10.73}$$

$$Y_{15} = Y_{51} = Y_{46} = Y_{64} = -Y_{14} = -Y_{41} = -Y_{56} = -Y_{65} = \frac{k_{13} \cdot k_{23} - k_{12}}{\sqrt{L_1 \cdot L_2} \cdot A} \tag{10.74}$$

$$Y_{25} = Y_{52} = Y_{43} = Y_{34} = -Y_{24} = -Y_{42} = -Y_{53} = -Y_{35} = \frac{k_{12} \cdot k_{13} - k_{23}}{\sqrt{L_2 \cdot L_3} \cdot A} \tag{10.75}$$

A more general approach for coupled inductors can be obtained by using the induction law:

$$V_L = j\omega L \cdot I_L + j\omega \cdot \sum_{n=1}^{N} k_n \cdot \sqrt{L \cdot L_n} \cdot I_{L,n} \tag{10.76}$$

where V_L and I_L is the voltage across and the current through the inductor, respectively. L is its inductance. The inductor is coupled with N other inductances L_n. The corresponding coupling factors are k_n and $I_{L,n}$ are the currents through the inductors.

Realizing this approach with the MNA matrix is straight forward: Every inductance L needs an additional matrix row. The corresponding element in the D matrix is $j\omega L$. If two inductors are coupled the cross element in the D matrix is $j\omega k \cdot \sqrt{L_1 \cdot L_2}$. For two coupled inductors this yields:

$$\begin{bmatrix} . & . & . & . & +1 & 0 \\ . & . & . & . & -1 & 0 \\ . & . & . & . & 0 & +1 \\ . & . & . & . & 0 & -1 \\ +1 & -1 & 0 & 0 & j\omega L_1 & j\omega k \cdot \sqrt{L_1 \cdot L_2} \\ 0 & 0 & +1 & -1 & j\omega k \cdot \sqrt{L_1 \cdot L_2} & j\omega L_2 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_{br1} \\ I_{br2} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.77}$$

Obviously, this approach has an advantage: It also works for zero inductances and for unity coupling factors. It has the disadvantage that it enlarges the MNA matrix.

10.10 Attenuator

The ideal attenuator with (power) attenuation L is frequency independent and the model is valid for DC and for AC simulation. It is determined by the following Z parameters.

$$Z_{11} = Z_{22} = Z_{ref} \cdot \frac{L+1}{L-1} \tag{10.78}$$

$$Z_{12} = Z_{21} = Z_{ref} \cdot \frac{2 \cdot \sqrt{L}}{L-1} \tag{10.79}$$

The Z parameter representation is not very practical as new lines in the MNA matrix have to be added. More useful are the Y parameters.

$$(\underline{Y}) = \frac{1}{Z_{ref} \cdot (L-1)} \cdot \begin{pmatrix} L+1 & -2 \cdot \sqrt{L} \\ -2 \cdot \sqrt{L} & L+1 \end{pmatrix} \tag{10.80}$$

Attenuator with (power) attenuation L, reference impedance Z_{ref} and temperature T:

$$(\underline{C}_Y) = 4 \cdot k \cdot T \cdot \text{Re}\,(\underline{Y}) = \frac{4 \cdot k \cdot T}{Z_{ref} \cdot (L-1)} \cdot \begin{pmatrix} L+1 & -2 \cdot \sqrt{L} \\ -2 \cdot \sqrt{L} & L+1 \end{pmatrix} \tag{10.81}$$

The scattering parameters and noise wave correlation matrix of an ideal attenuator with (power) attenuation L (loss) (or power gain $G = 1/L$) in reference to the impedance Z_{ref} writes as follows.

$$S_{11} = S_{22} = \frac{r \cdot (L-1)}{L - r^2} = \frac{r \cdot (1-G)}{1 - r^2 \cdot G} \tag{10.82}$$

$$S_{12} = S_{21} = \frac{\sqrt{L} \cdot (1 - r^2)}{L - r^2} = \frac{\sqrt{G} \cdot (1 - r^2)}{1 - r^2 \cdot G} \tag{10.83}$$

$$(\underline{C}) = k \cdot T \cdot \frac{(L-1) \cdot (r^2 - 1)}{(L - r^2)^2} \cdot \begin{pmatrix} -r^2 - L & 2 \cdot r\sqrt{L} \\ 2 \cdot r\sqrt{L} & -r^2 - L \end{pmatrix} \tag{10.84}$$

with

$$r = \frac{Z_{ref} - Z_0}{Z_{ref} + Z_0} \tag{10.85}$$

10.11 Amplifier

An ideal amplifier increases signal strength from input to output and blocks all signals flowing into the output. The ideal amplifier is an isolator with voltage gain G and is determined by the following Z or Y parameters (valid for DC and AC simulation).

$$Z_{11} = Z_1 \qquad Z_{12} = 0 \tag{10.86}$$

$$Z_{21} = 2 \cdot \sqrt{Z_1 \cdot Z_2} \cdot G \qquad Z_{22} = Z_2 \tag{10.87}$$

$$Y_{11} = \frac{1}{Z_1} \qquad Y_{12} = 0 \tag{10.88}$$

$$Y_{21} = -\frac{2 \cdot G}{\sqrt{Z_1 \cdot Z_2}} \qquad Y_{22} = \frac{1}{Z_2} \tag{10.89}$$

With the reference impedance of the input Z_1 and the one of the output Z_2 and the voltage amplification G, the scattering parameters of an ideal amplifier write as follows.

$$S_{11} = \frac{Z_1 - Z_0}{Z_1 + Z_0} \tag{10.90}$$

$$S_{12} = 0 \tag{10.91}$$

$$S_{22} = \frac{Z_2 - Z_0}{Z_2 + Z_0} \tag{10.92}$$

$$S_{21} = \frac{4 \cdot Z_0 \cdot \sqrt{Z_1 \cdot Z_2} \cdot G}{(Z_1 + Z_0) \cdot (Z_2 + Z_0)} \tag{10.93}$$

Being on temperature T, the noise wave correlation matrix with reference to impedance Z_1 and Z_2 (input and output) writes as follows.

$$\underline{c}_{11} = \underline{c}_{12} = \underline{c}_{21} = 0 \tag{10.94}$$

$$\underline{c}_{22} = 4 \cdot k \cdot T \cdot Z_0 \cdot Z_2 \cdot (F - 1) \cdot \left(\frac{G}{Z_0 + Z_2}\right)^2 \tag{10.95}$$

And for y parameters the noise correlation matrix writes:

$$\underline{c}_{y11} = \underline{c}_{y12} = \underline{c}_{y21} = 0 \tag{10.96}$$

$$\underline{c}_{y22} = 4 \cdot k \cdot T \cdot (F - 1) \cdot \frac{G^2}{Z_2} \tag{10.97}$$

10.12 Isolator

An isolator is a one-way two-port, transporting incoming waves lossless from the input (port 1) to the output (port 2), but dissipating all waves flowing into the output. The ideal isolator with reference impedances Z_1 (input) and Z_2 (output) is determined by the following Z parameters (for DC and AC simulation).

$$Z_{11} = Z_1 \qquad Z_{12} = 0 \tag{10.98}$$

$$Z_{21} = 2 \cdot \sqrt{Z_1 \cdot Z_2} \qquad Z_{22} = Z_2 \tag{10.99}$$

A more useful MNA representation is with Y parameters.

$$(\underline{Y}) = \begin{pmatrix} \frac{1}{Z_1} & 0 \\ \frac{-2}{\sqrt{Z_1 \cdot Z_2}} & \frac{1}{Z_2} \end{pmatrix} \tag{10.100}$$

Isolator with reference impedance Z_1 (input) and Z_2 (output) and temperature T:

$$(\underline{C}_Y) = 4 \cdot k \cdot T \cdot \begin{pmatrix} \frac{1}{Z_1} & 0 \\ \frac{-2}{\sqrt{Z_1 \cdot Z_2}} & \frac{1}{Z_2} \end{pmatrix} \tag{10.101}$$

With the reference impedance of the input Z_1 and the one of the output Z_2, the scattering parameters of an ideal isolator writes as follows.

$$S_{11} = \frac{Z_1 - Z_0}{Z_1 + Z_0} \tag{10.102}$$

$$S_{12} = 0 \tag{10.103}$$

$$S_{22} = \frac{Z_2 - Z_0}{Z_2 + Z_0} \tag{10.104}$$

$$S_{21} = \sqrt{1 - (S_{11})^2} \cdot \sqrt{1 - (S_{22})^2} \tag{10.105}$$

Being on temperature T, the noise wave correlation matrix of an ideal isolator with reference impedance Z_1 and Z_2 (input and output) writes as follows.

$$(\underline{C}) = \frac{4 \cdot k \cdot T \cdot Z_0}{(Z_1 + Z_0)^2} \cdot \begin{pmatrix} Z_1 & \sqrt{Z_1 \cdot Z_2} \cdot \dfrac{Z_0 - Z_1}{Z_0 + Z_2} \\ \sqrt{Z_1 \cdot Z_2} \cdot \dfrac{Z_0 - Z_1}{Z_0 + Z_2} & Z_2 \cdot \left(\dfrac{Z_1 - Z_0}{Z_2 + Z_0}\right)^2 \end{pmatrix} \tag{10.106}$$

10.13 Circulator

A circulator is a 3-port device, transporting incoming waves lossless from port 1 to port 2, from port 2 to port 3 and from port 3 to port 1. In all other directions, there is no energy flow. The ideal circulator cannot be characterized with Z or Y parameters, because their values are partly infinite. But implementing with S parameters is practical (see equation 10.8).

With the reference impedances Z_1, Z_2 and Z_3 for the ports 1, 2 and 3 the scattering matrix of an ideal circulator writes as follows.

$$denom = 1 - r_1 \cdot r_2 \cdot r_3 \tag{10.107}$$

$$r_1 = \frac{Z_0 - Z_1}{Z_0 + Z_1} \quad , \quad r_2 = \frac{Z_0 - Z_2}{Z_0 + Z_2} \quad , \quad r_3 = \frac{Z_0 - Z_3}{Z_0 + Z_3} \tag{10.108}$$

$$S_{11} = \frac{r_2 \cdot r_3 - r_1}{denom} \quad , \quad S_{22} = \frac{r_1 \cdot r_3 - r_2}{denom} \quad , \quad S_{33} = \frac{r_1 \cdot r_2 - r_3}{denom} \tag{10.109}$$

$$S_{12} = \sqrt{\frac{Z_2}{Z_1}} \cdot \frac{Z_1 + Z_0}{Z_2 + Z_0} \cdot \frac{r_3 \cdot (1 - r_1^2)}{denom} \quad , \quad S_{13} = \sqrt{\frac{Z_3}{Z_1}} \cdot \frac{Z_1 + Z_0}{Z_3 + Z_0} \cdot \frac{1 - r_1^2}{denom} \tag{10.110}$$

$$S_{21} = \sqrt{\frac{Z_1}{Z_2}} \cdot \frac{Z_2 + Z_0}{Z_1 + Z_0} \cdot \frac{1 - r_2^2}{denom} \quad , \quad S_{23} = \sqrt{\frac{Z_3}{Z_2}} \cdot \frac{Z_2 + Z_0}{Z_3 + Z_0} \cdot \frac{r_1 \cdot (1 - r_2^2)}{denom} \tag{10.111}$$

$$S_{31} = \sqrt{\frac{Z_1}{Z_3}} \cdot \frac{Z_3 + Z_0}{Z_1 + Z_0} \cdot \frac{r_2 \cdot (1 - r_3^2)}{denom} \quad , \quad S_{32} = \sqrt{\frac{Z_2}{Z_3}} \cdot \frac{Z_3 + Z_0}{Z_2 + Z_0} \cdot \frac{1 - r_3^2}{denom} \tag{10.112}$$

An ideal circulator is noise free.

10.14 Phase shifter

A phase shifter has the same model equations like a transmission line. Its Z-parameters therefore writes as follows.

$$Z_{11} = Z_{22} = \frac{j \cdot Z_{ref}}{\tan(\phi)} = Z_{12} \cdot \cos(\phi) \tag{10.113}$$

$$Z_{12} = Z_{21} = \frac{j \cdot Z_{ref}}{\sin(\phi)} \tag{10.114}$$

The admittance parameters required for the AC analysis result in

$$Y_{11} = Y_{22} = \frac{j}{Z_{ref} \cdot \tan(\phi)} = -Y_{12} \cdot \cos(\phi) \tag{10.115}$$

$$Y_{12} = Y_{21} = \frac{1}{j \cdot Z_{ref} \cdot \sin(\phi)} \tag{10.116}$$

where ϕ denotes the actual phase shift of the device. Note that these parameters are not defined for some phase shifts. The cosine term should therefore be preferred instead of the tangen term. For zero phase shift ($\phi = 0$) an ideal short circuit can be taken. For $\phi = \pi$ the chain parameters must be taken instead of Y-parameters.

The scattering parameters of an ideal phase shifter with phase shift ϕ and reference impedance Z_{ref} writes as follows.

$$r = \frac{Z_{ref} - Z_0}{Z_{ref} + Z_0} \tag{10.117}$$

$$S_{11} = S_{22} = \frac{r \cdot (1 - \exp(j \cdot 2\phi))}{1 - r^2 \cdot \exp(j \cdot 2\phi)} \tag{10.118}$$

$$S_{12} = S_{21} = \frac{(1 - r^2) \cdot \exp(j \cdot \phi)}{1 - r^2 \cdot \exp(j \cdot 2\phi)} \tag{10.119}$$

An ideal phase shifter is noise free. Note that there's no model for DC and transient analysis, because the equations exhibit imaginary numbers and the impulse response isn't finite.

10.15 Coupler (Hybrid)

According to the port numbers in fig. 10.5 the Y-parameters of a coupler write as follows.

$$Y_{11} = Y_{22} = Y_{33} = Y_{44} = \frac{A \cdot (2 - A)}{D} \tag{10.120}$$

$$Y_{12} = Y_{21} = Y_{34} = Y_{43} = \frac{-A \cdot B}{D} \tag{10.121}$$

$$Y_{13} = Y_{31} = Y_{24} = Y_{42} = \frac{C \cdot (A - 2)}{D} \tag{10.122}$$

$$Y_{14} = Y_{41} = Y_{23} = Y_{32} = \frac{B \cdot C}{D} \tag{10.123}$$

$$\tag{10.124}$$

with

$$A = k^2 \cdot (1 + \exp(j \cdot 2\phi)) \tag{10.125}$$

$$B = 2 \cdot \sqrt{1 - k^2} \tag{10.126}$$

$$C = 2 \cdot k \cdot \exp(j \cdot \phi) \tag{10.127}$$

$$D = Z_{ref} \cdot \left(A^2 - C^2\right) \tag{10.128}$$

$$\tag{10.129}$$

whereas $0 < k < 1$ denotes the coupling factor, ϕ the phase shift of the coupling path and Z_{ref} the reference impedance. The coupler becomes a hybrid by setting $k = 1/\sqrt{2}$.
The noise wave correlation matrix contains zeros in the first main diagonale, whereas there are some non-diagonale elements being $2k\sqrt{1 - k^2} \cdot \cos\phi$. Thus, this component shows physical behaviour for $\phi = 0.5 \cdot \pi + n \cdot \pi$ only (90 degree).
Note that there's no model for DC and transient analysis, because the equations exhibit imaginary numbers and the impulse response isn't finite. Thus, it's recommended to model the coupler by just making a short between port 1 and port 2 and between port 3 and port 4.

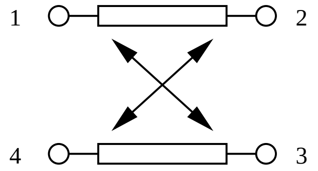


Figure 10.5: ideal coupler device

The scattering parameters of a coupler are:

$$S_{11} = S_{22} = S_{33} = S_{44} = 0 \tag{10.130}$$

$$S_{14} = S_{23} = S_{32} = S_{41} = 0 \tag{10.131}$$

$$S_{12} = S_{21} = S_{34} = S_{43} = \sqrt{1-k^2} \tag{10.132}$$

$$S_{13} = S_{31} = S_{24} = S_{42} = k \cdot \exp\left(j\phi\right) \tag{10.133}$$

whereas $0 < k < 1$ denotes the coupling factor, ϕ the phase shift of the coupling path. Extending them for an arbitrary reference impedance Z_{ref}, they already become quite complex:

$$\begin{align}
r &= \frac{Z_0 - Z_{ref}}{Z_0 + Z_{ref}} \tag{10.134}\\
A &= k^2 \cdot \left(\exp\left(j \cdot 2\phi\right) + 1\right) \tag{10.135}\\
B &= r^2 \cdot \left(1 - A\right) \tag{10.136}\\
C &= k^2 \cdot \left(\exp\left(j \cdot 2\phi\right) - 1\right) \tag{10.137}\\
D &= 1 - 2 \cdot r^2 \cdot \left(1 + C\right) + B^2 \tag{10.138}\\
& \tag{10.139}
\end{align}$$

$$S_{11} = S_{22} = S_{33} = S_{44} = r \cdot \frac{A \cdot B + C + 2 \cdot r^2 \cdot k^2 \cdot \exp\left(j \cdot 2\phi\right)}{D} \tag{10.140}$$

$$S_{12} = S_{21} = S_{34} = S_{43} = \sqrt{1-k^2} \cdot \frac{\left(1 - r^2\right) \cdot \left(1 - B\right)}{D} \tag{10.141}$$

$$S_{13} = S_{31} = S_{24} = S_{42} = k \cdot \exp\left(j\phi\right) \cdot \frac{\left(1 - r^2\right) \cdot \left(1 + B\right)}{D} \tag{10.142}$$

$$S_{14} = S_{23} = S_{32} = S_{41} = 2 \cdot \sqrt{1-k^2} \cdot k \cdot \exp\left(j\phi\right) \cdot r \cdot \frac{\left(1 - r^2\right)}{D} \tag{10.143}$$

An ideal coupler is noise free.

10.16 Unsymmetrical Hybrid

The previous chapter deals with symmetrical hybrids. This chapter describes the model of unsymmetrical hybrids , i.e. hybrids where one path has a phase shift only. Figure 10.6 shows a symbol of these devices. Its s-parameters writes as follows.

$$S = \frac{1}{\sqrt{2}} \cdot \begin{bmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & e^{j\phi} \\ 1 & 1 & 0 & 0 \\ 1 & e^{j\phi} & 0 & 0 \end{bmatrix} \tag{10.144}$$

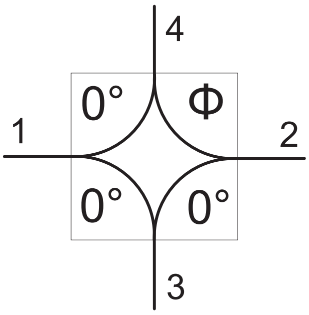


Figure 10.6: unsymmetrical hybrid (3dB-coupler)

The noise wave correlation matrix contains zeros in the first main diagonale, whereas there are some non-diagonale elements being $1 + \exp(\pm j\phi)$. Thus, this component shows physical behaviour for $\phi = \pi + n \cdot 2\pi$ only (180 degree).
The y-parameters have a pole at $\phi = \pi + n \cdot 2\pi$. Thus the component has to be modeled with s-parameters.

10.17 Gyrator

A gyrator is an impedance inverter. Thus, for example, it converts a capacitance into an inductance and vice versa. The ideal gyrator with the ratio $G = 1/R$, as shown in fig. 10.7, is determined by the following equations which introduce two more unknowns in the MNA matrix.

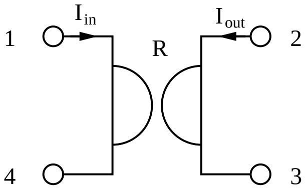


Figure 10.7: ideal gyrator

$$I_{in} = G \cdot (V_2 - V_3) \quad \rightarrow \quad G \cdot V_2 - G \cdot V_3 - I_{in} = 0 \tag{10.145}$$

$$I_{out} = -G \cdot (V_1 - V_4) \quad \rightarrow \quad -G \cdot V_1 + G \cdot V_4 - I_{out} = 0 \tag{10.146}$$

The new unknown variables I_{out} and I_{in} must be considered by the four remaining simple equations.

$$I_1 = I_{in} \quad I_2 = I_{out} \quad I_3 = -I_{out} \quad I_4 = -I_{in} \tag{10.147}$$

As can be seen, a gyrator consists of two cross-connected VCCS (section 10.20.1). Hence, its y-parameter matrix is:

$$(\underline{Y}) = \begin{bmatrix} 0 & G & -G & 0 \\ -G & 0 & 0 & G \\ G & 0 & 0 & -G \\ 0 & -G & G & 0 \end{bmatrix} \tag{10.148}$$

The scattering matrix of an ideal gyrator with the ratio $G = 1/R$ writes as follows.

$$r = \frac{R}{Z_0} = \frac{1}{G \cdot Z_0} \tag{10.149}$$

$$S_{11} = S_{22} = S_{33} = S_{44} = \frac{r^2}{r^2 + 4} \tag{10.150}$$

$$S_{14} = S_{23} = S_{32} = S_{41} = 1 - S_{11} \tag{10.151}$$

$$S_{12} = -S_{13} = -S_{21} = S_{24} = S_{31} = -S_{34} = -S_{42} = S_{43} = \frac{2 \cdot r}{r^2 + 4} \tag{10.152}$$

10.18 Voltage and current sources

For an AC analysis, DC sources are short circuit (voltage source) or open circuit (current source), respectively. Accordingly, for a DC analysis, AC sources are short circuit (voltage source) or open circuit (current source), respectively. As these sources have no internal resistance, they are noisefree.

The MNA matrix of a current source is (with short circuit current I_0 flowing into node 1 and out of node 2):

$$\begin{bmatrix} . & . \\ . & . \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} I_0 \\ -I_0 \end{bmatrix} \tag{10.153}$$

The MNA matrix of a voltage source is (with open circuit voltage U_0 across node 1 to node 2):

$$\begin{bmatrix} . & . & 1 \\ . & . & -1 \\ 1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{in} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ U_0 \end{bmatrix} \tag{10.154}$$

The MNA matrix of a power source is (with internal resistance R and available power P):

$$\begin{bmatrix} \frac{1}{R} & -\frac{1}{R} \\ -\frac{1}{R} & \frac{1}{R} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \end{bmatrix} = \begin{bmatrix} \sqrt{\frac{8 \cdot P}{R}} \\ -\sqrt{\frac{8 \cdot P}{R}} \end{bmatrix} \tag{10.155}$$

The factor "8" is because of:

- transforming peak current value into effective value (factor two)
- at power matching the internal resistance dissipates the same power as the load (gives factor four).

The noise current correlation matrix of a power source equals the one of a resistor with resistance R.

All voltage sources (AC and DC) are short circuits and therefore their S-parameter matrix equals the one of the DC block. All current sources are open circuits and therefore their S-parameter matrix equals the one of the DC feed.

10.19 Noise sources

To implement the frequency dependencies of all common noise PSDs the following equation can be used.

$$PSD = \frac{PSD_0}{a + b \cdot f^c} \tag{10.156}$$

Where f is frequency and a, b, c are the parameters. The following PSDs appear in electric devices.

white noise (thermal noise, shot noise):	$a = 0,\ b = 1,\ c = 0$
pink noise (flicker noise):	$a = 0,\ b = 1,\ c = 1$
Lorentzian PSD (generation-recombination noise):	$a = 1,\ b = 1/f_c^2,\ c = 2$

10.19.1 Noise current source

Noise current source with a current power spectral density of $cPSD$:

$$(\underline{C}_Y) = cPSD \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{10.157}$$

The MNA matrix entries for DC and AC analysis are all zero.

The noise wave correlation matrix of a noise current source with current power spectral density $cPSD$ and its S parameter matrix write as follows.

$$(\underline{C}) = cPSD \cdot Z_0 \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \qquad (\underline{S}) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \tag{10.158}$$

10.19.2 Noise voltage source

A noise voltage source (voltage power spectral density $vPSD$) cannot be modeled with the noise current matrix. That is why one has to use a noise current source (current power spectral density $cPSD$) connected to a gyrator (transimpedance R) satisfying the equation

$$vPSD = cPSD \cdot R^2 \tag{10.159}$$

Figure 10.8 shows an example.

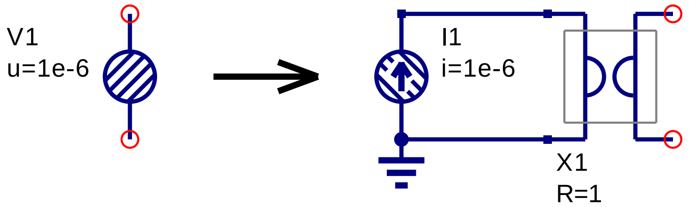


Figure 10.8: noise voltage source (left-hand side) and its equivalent circuit (right-hand side)

The MNA matrix entries of the above construct (gyrator ratio $R = 1$) is similiar to a voltage source with zero voltage.

$$\begin{bmatrix} . & . & -1 \\ . & . & 1 \\ 1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_x \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.160}$$

The appropriate noise current correlation matrix yields:

$$(\underline{C}_Y) = cPSD \cdot \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{10.161}$$

The noise wave correlation matrix of a noise voltage source with voltage power spectral density $vPSD$ and its S parameter matrix write as follows.

$$(\underline{C}) = \frac{vPSD}{4 \cdot Z_0} \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \qquad (\underline{S}) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \tag{10.162}$$

10.19.3 Correlated noise sources

For two correlated noise current sources the (normalized) correlation coefficient K must be known (with $|K| = 0 \ldots 1$). If the first noise source has the current power spectral density S_{i1} and is connected to node 1 and 2, and if furthermore the second noise source has the spectral density S_{i2} and is connected to node 3 and 4, then the correlation matrix writes:

$$(\underline{C}_Y) = \begin{pmatrix} S_{i1} & -S_{i1} & K \cdot \sqrt{S_{i1} \cdot S_{i2}} & -K \cdot \sqrt{S_{i1} \cdot S_{i2}} \\ -S_{i1} & S_{i1} & -K \cdot \sqrt{S_{i1} \cdot S_{i2}} & K \cdot \sqrt{S_{i1} \cdot S_{i2}} \\ K \cdot \sqrt{S_{i1} \cdot S_{i2}} & -K \cdot \sqrt{S_{i1} \cdot S_{i2}} & S_{i2} & -S_{i2} \\ -K \cdot \sqrt{S_{i1} \cdot S_{i2}} & K \cdot \sqrt{S_{i1} \cdot S_{i2}} & -S_{i2} & S_{i2} \end{pmatrix} \tag{10.163}$$

The MNA matrix entries for DC and AC analysis are all zero.

The noise wave correlation matrix of two correlated noise current sources with current power spectral densities S_{i1} and S_{i2} and correlation coefficient K writes as follows.

$$(\underline{C}) = Z_0 \cdot \begin{pmatrix} S_{i1} & -S_{i1} & K \cdot \sqrt{S_{i1} \cdot S_{i2}} & -K \cdot \sqrt{S_{i1} \cdot S_{i2}} \\ -S_{i1} & S_{i1} & -K \cdot \sqrt{S_{i1} \cdot S_{i2}} & K \cdot \sqrt{S_{i1} \cdot S_{i2}} \\ K \cdot \sqrt{S_{i1} \cdot S_{i2}} & -K \cdot \sqrt{S_{i1} \cdot S_{i2}} & S_{i2} & -S_{i2} \\ -K \cdot \sqrt{S_{i1} \cdot S_{i2}} & K \cdot \sqrt{S_{i1} \cdot S_{i2}} & -S_{i2} & S_{i2} \end{pmatrix} \tag{10.164}$$

$$(\underline{S}) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{10.165}$$

For two correlated noise voltage sources two extra rows and columns are needed in the MNA matrix:

$$\begin{bmatrix} . & . & . & . & -1 & 0 \\ . & . & . & . & 1 & 0 \\ . & . & . & . & 0 & -1 \\ . & . & . & . & 0 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_{x1} \\ I_{x2} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.166}$$

The appropriate noise current correlation matrix (with the noise voltage power spectral densities S_{v1} and S_{v2} and the correlation coefficient K) yields:

$$(\underline{C}_Y) = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & S_{v1} & K \cdot \sqrt{S_{v1} \cdot S_{v2}} \\ 0 & 0 & 0 & 0 & K \cdot \sqrt{S_{v1} \cdot S_{v2}} & S_{v2} \end{pmatrix} \tag{10.167}$$

The noise wave correlation matrix of two correlated noise voltage sources with voltage power spectral densities S_{v1} and S_{v2} and correlation coefficient K and its S parameter matrix write as follows.

$$(\underline{C}) = \frac{1}{4 \cdot Z_0} \cdot \begin{pmatrix} S_{v1} & -S_{v1} & K \cdot \sqrt{S_{v1} \cdot S_{v2}} & -K \cdot \sqrt{S_{v1} \cdot S_{v2}} \\ -S_{v1} & S_{v1} & -K \cdot \sqrt{S_{v1} \cdot S_{v2}} & K \cdot \sqrt{S_{v1} \cdot S_{v2}} \\ K \cdot \sqrt{S_{v1} \cdot S_{v2}} & -K \cdot \sqrt{S_{v1} \cdot S_{v2}} & S_{v2} & -S_{v2} \\ -K \cdot \sqrt{S_{v1} \cdot S_{v2}} & K \cdot \sqrt{S_{v1} \cdot S_{v2}} & -S_{v2} & S_{v2} \end{pmatrix} \tag{10.168}$$

$$(\underline{S}) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{10.169}$$

If a noise current source (ports 1 and 2) and a noise voltage source (ports 3 and 4) are correlated, the MNA matrix entries are as follows.

$$\begin{bmatrix} . & . & . & . & 0 \\ . & . & . & . & 0 \\ . & . & . & . & -1 \\ . & . & . & . & 1 \\ 0 & 0 & 1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_x \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.170}$$

The appropriate noise current correlation matrix (with the noise power spectral densities S_{i1} and S_{v2} and the correlation coefficient K) yields:

$$(\underline{C}_Y) = \begin{pmatrix} S_{i1} & -S_{i1} & 0 & 0 & K \cdot \sqrt{S_{i1} \cdot S_{v2}} \\ -S_{i1} & S_{i1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ K \cdot \sqrt{S_{i1} \cdot S_{v2}} & 0 & 0 & 0 & S_{v2} \end{pmatrix} \tag{10.171}$$

Note: Because the gyrator factor (It is unity.) has been omitted in the above matrix the units are not correct. This can be ignored.

The noise wave correlation matrix of one correlated noise current source S_{i1} and one noise voltage source S_{v2} with correlation coefficient K writes as follows.

$$(\underline{C}) = \begin{pmatrix} Z_0 \cdot S_{i1} & -Z_0 \cdot S_{i1} & K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} & -K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} \\ -Z_0 \cdot S_{i1} & Z_0 \cdot S_{i1} & -K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} & K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} \\ K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} & -K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} & S_{v2}/4/Z_0 & -S_{v2}/4/Z_0 \\ -K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} & K/2 \cdot \sqrt{S_{i1} \cdot S_{v2}} & -S_{v2}/4/Z_0 & S_{v2}/4/Z_0 \end{pmatrix} \tag{10.172}$$

$$(\underline{S}) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{10.173}$$

10.20 Controlled sources

The models of the controlled sources contain the transfer factor G. It is complex because of the delay time T and frequency f.

$$\underline{G} = G \cdot e^{j\omega T} = G \cdot e^{j \cdot 2\pi f \cdot T} \tag{10.174}$$

During a DC analysis (frequency zero) it becomes real because the exponent factor is unity.

10.20.1 Voltage controlled current source

The voltage-dependent current source (VCCS), as shown in fig. 10.9, is determined by the following equation which introduces one more unknown in the MNA matrix.

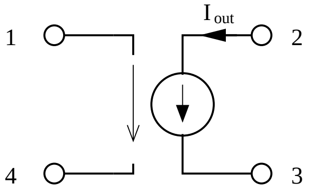


Figure 10.9: voltage controlled current source

$$I_{out} = G \cdot (V_1 - V_4) \quad \rightarrow \quad V_1 - V_4 - \frac{1}{G} \cdot I_{out} = 0 \tag{10.175}$$

The new unknown variable I_{out} must be considered by the four remaining simple equations.

$$I_1 = 0 \quad I_2 = I_{out} \quad I_3 = -I_{out} \quad I_4 = 0 \tag{10.176}$$

And in matrix representation this is:

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ G & 0 & 0 & -G \\ -G & 0 & 0 & G \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.177}$$

The scattering matrix of the voltage controlled current source writes as follows (τ is time delay).

$$S_{11} = S_{22} = S_{33} = S_{44} = 1 \tag{10.178}$$

$$S_{12} = S_{13} = S_{14} = S_{23} = S_{32} = S_{41} = S_{42} = S_{43} = 0 \tag{10.179}$$

$$S_{21} = S_{34} = -2 \cdot G \cdot \exp\left(-j\omega\tau\right) \tag{10.180}$$

$$S_{24} = S_{31} = 2 \cdot G \cdot \exp\left(-j\omega\tau\right) \tag{10.181}$$

10.20.2 Current controlled current source

The current-dependent current source (CCCS), as shown in fig. 10.10, is determined by the following equation which introduces one more unknown in the MNA matrix.

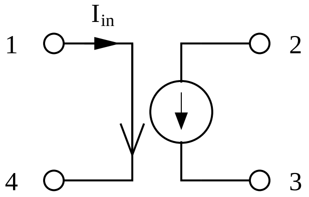


Figure 10.10: current controlled current source

$$V_1 - V_4 = 0 \tag{10.182}$$

The new unknown variable I_{in} must be considered by the four remaining simple equations.

$$I_1 = I_{in} \quad I_2 = G \cdot I_{in} \quad I_3 = -G \cdot I_{in} \quad I_4 = -I_{in} \tag{10.183}$$

And in matrix representation this is:

$$\begin{bmatrix} . & . & . & . & 1 \\ . & . & . & . & G \\ . & . & . & . & -G \\ . & . & . & . & -1 \\ 1 & 0 & 0 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_{in} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.184}$$

The scattering matrix of the current controlled current source writes as follows (τ is time delay).

$$S_{14} = S_{22} = S_{33} = S_{41} = 1 \tag{10.185}$$

$$S_{11} = S_{12} = S_{13} = S_{23} = S_{32} = S_{42} = S_{43} = S_{44} = 0 \tag{10.186}$$

$$S_{21} = S_{34} = -G \cdot \exp\left(-j\omega\tau\right) \tag{10.187}$$

$$S_{24} = S_{31} = G \cdot \exp\left(-j\omega\tau\right) \tag{10.188}$$

10.20.3 Voltage controlled voltage source

The voltage-dependent voltage source (VCVS), as shown in fig. 10.11, is determined by the following equation which introduces one more unknown in the MNA matrix.

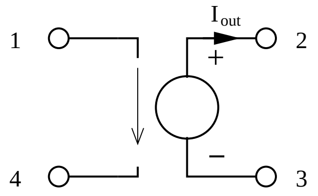


Figure 10.11: voltage controlled voltage source

$$V_2 - V_3 = G \cdot (V_1 - V_4) \quad \rightarrow \quad V_1 \cdot G - V_2 + V_3 - V_4 \cdot G = 0 \tag{10.189}$$

The new unknown variable I_{out} must be considered by the four remaining simple equations.

$$I_1 = 0 \quad I_2 = -I_{out} \quad I_3 = I_{out} \quad I_4 = 0 \tag{10.190}$$

And in matrix representation this is:

$$\begin{bmatrix} . & . & . & . & 0 \\ . & . & . & . & -1 \\ . & . & . & . & 1 \\ . & . & . & . & 0 \\ G & -1 & 1 & -G & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.191}$$

The scattering matrix of the voltage controlled voltage source writes as follows (τ is time delay).

$$S_{11} = S_{23} = S_{32} = S_{44} = 1 \tag{10.192}$$

$$S_{12} = S_{13} = S_{14} = S_{22} = S_{33} = S_{41} = S_{42} = S_{43} = 0 \tag{10.193}$$

$$S_{21} = S_{34} = G \cdot \exp\left(-j\omega\tau\right) \tag{10.194}$$

$$S_{24} = S_{31} = -G \cdot \exp\left(-j\omega\tau\right) \tag{10.195}$$

10.20.4 Current controlled voltage source

The current-dependent voltage source (CCVS), as shown in fig. 10.12, is determined by the following equations which introduce two more unknowns in the MNA matrix.

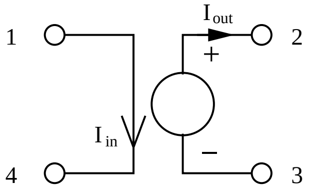


Figure 10.12: current controlled voltage source

$$V_1 - V_4 = 0 \tag{10.196}$$

$$V_2 - V_3 = G \cdot I_{in} \quad \rightarrow \quad V_2 - V_3 - I_{in} \cdot G = 0 \tag{10.197}$$

The new unknown variables I_{out} and I_{in} must be considered by the four remaining simple equations.

$$I_1 = I_{in} \quad I_2 = -I_{out} \quad I_3 = I_{out} \quad I_4 = -I_{in} \tag{10.198}$$

The matrix representation needs to be augmented by two more new rows (for the new unknown variables) and their corresponding columns.

$$\begin{bmatrix} . & . & . & . & 1 & 0 \\ . & . & . & . & 0 & 1 \\ . & . & . & . & 0 & -1 \\ . & . & . & . & -1 & 0 \\ 0 & 1 & -1 & 0 & -G & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ V_4 \\ I_{in} \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{10.199}$$

The scattering matrix of the current controlled voltage source writes as follows (τ is time delay).

$$S_{14} = S_{23} = S_{32} = S_{41} = 1 \tag{10.200}$$

$$S_{11} = S_{12} = S_{13} = S_{22} = S_{33} = S_{42} = S_{43} = S_{44} = 0 \tag{10.201}$$

$$S_{21} = S_{34} = \frac{G}{2} \cdot \exp\left(-j\omega\tau\right) \tag{10.202}$$

$$S_{24} = S_{31} = -\frac{G}{2} \cdot \exp\left(-j\omega\tau\right) \tag{10.203}$$

10.21 AM modulated AC source

An AC voltage source in the time-domain is characterized by its frequency f, the initial phase ϕ and the amplitude A. During amplitude modulation the modulation level M must be considered. The output voltage of the source is determined by the following equation.

$$V_1(t) - V_2(t) = (1 + M \cdot V_3(t)) \cdot A \cdot \sin(\omega \cdot t + \phi) \tag{10.204}$$

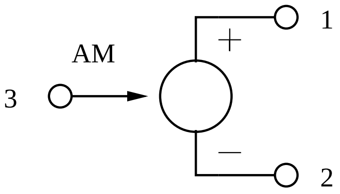


Figure 10.13: AM modulated AC source

The appropriate MNA matrix entries during the transient analysis decribing a simple linear operation can be written as

$$\begin{bmatrix} . & . & . & 1 \\ . & . & . & -1 \\ . & . & . & 0 \\ 1 & -1 & -M \cdot A \cdot \sin(\omega \cdot t + \phi) & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ V_3(t) \\ J_1(t) \end{bmatrix} = \begin{bmatrix} I_1(t) \\ I_2(t) \\ I_3(t) \\ A \cdot \sin(\omega \cdot t + \phi) \end{bmatrix} \tag{10.205}$$

10.22 PM modulated AC source

The phase modulated AC source is also characterized by the frequency f, the amplidude A and by an initial phase ϕ. The output voltage in the time-domain is determinded by the following equation

$$V_1(t) - V_2(t) = A \cdot \sin(\omega \cdot t + \phi + 2\pi \cdot M \cdot V_3(t)) \tag{10.206}$$

whereas M denotes the modulation index and V_3 the modulating voltage.

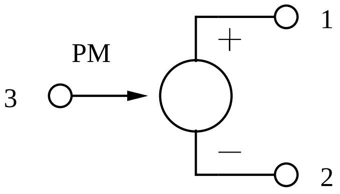


Figure 10.14: PM modulated AC source

The component is non-linear in the frequency- as well in the time-domain. In order to prepare the source for subsequent Newton-Raphson iterations the derivative

$$g = \frac{\partial(V_1 - V_2)}{\partial V_3} = 2\pi \cdot M \cdot A \cdot \cos(\omega \cdot t + \phi + 2\pi \cdot M \cdot V_3) \tag{10.207}$$

is required. With this at hand the MNA matrix entries of the PM modulated AC voltage source during the transient analysis can be written as

$$\begin{bmatrix} . & . & . & +1 \\ . & . & . & -1 \\ . & . & . & 0 \\ +1 & -1 & g & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1(t) \\ V_2(t) \\ V_3(t) \\ J_1(t) \end{bmatrix} = \begin{bmatrix} I_1(t) \\ I_2(t) \\ I_3(t) \\ g \cdot V_3 - A \cdot \sin(\omega \cdot t + \phi + 2\pi \cdot M \cdot V_3) \end{bmatrix} \tag{10.208}$$

10.23 Transmission Line

A transmission line is usually described by its ABCD-matrix. (Note that in ABCD-matrices, i.e. the chain matrix representation, the current i_2 is defined to flow out of the output port.)

$$(\underline{A}) = \begin{pmatrix} \cosh(\gamma \cdot l) & Z_L \cdot \sinh(\gamma \cdot l) \\ \sinh(\gamma \cdot l)/Z_L & \cosh(\gamma \cdot l) \end{pmatrix} \tag{10.209}$$

These can easily be recalculated into impedance parameters.

$$Z_{11} = Z_{22} = \frac{Z_L}{\tanh(\gamma \cdot l)} \tag{10.210}$$

$$Z_{12} = Z_{21} = \frac{Z_L}{\sinh(\gamma \cdot l)} \tag{10.211}$$

Or in admittance parameter representation it yields

$$\begin{aligned} Y_{11} = Y_{22} &= \frac{1}{Z_L \cdot \tanh(\gamma \cdot l)} \\ Y_{12} = Y_{21} &= \frac{-1}{Z_L \cdot \sinh(\gamma \cdot l)} \end{aligned} \tag{10.212}$$

whence γ denotes the propagation constant $\alpha + j\beta$ and l is the length of the transmission line. Z_L represents the characteristic impedance of the transmission line. The Y-parameters as defined by eq. (10.212) can be used for the microstrip line. For an ideal, i.e. lossless, transmission lines they write accordingly.

$$Z_{11} = Z_{22} = \frac{Z_L}{j \cdot \tan(\beta \cdot l)} \tag{10.213}$$

$$Z_{12} = Z_{21} = \frac{Z_L}{j \cdot \sin(\beta \cdot l)} \tag{10.214}$$

$$Y_{11} = Y_{22} = \frac{1}{j \cdot Z_L \cdot \tan(\beta \cdot l)} \tag{10.215}$$

$$Y_{12} = Y_{21} = \frac{j}{Z_L \cdot \sin(\beta \cdot l)} \tag{10.216}$$

The scattering matrix of an ideal, lossless transmission line with impedance Z and electrical length l writes as follows.

$$r = \frac{Z - Z_0}{Z + Z_0} \tag{10.217}$$

$$p = \exp\left(-j\omega \frac{l}{c_0}\right) \tag{10.218}$$

$$S_{11} = S_{22} = \frac{r \cdot (1 - p^2)}{1 - r^2 \cdot p^2} \quad , \quad S_{12} = S_{21} = \frac{p \cdot (1 - r^2)}{1 - r^2 \cdot p^2} \tag{10.219}$$

With $c_0 = 299\ 792\ 458$ m/s being the vacuum light velocity. Adding attenuation to the transmission line, the quantity p extends to:

$$p = \exp\left(-j\omega \frac{l}{c_0} - \alpha \cdot l\right) \tag{10.220}$$

Another equivalent equation set for the calculation of the scattering parameters is the following: With the physical length l of the component, its impedance Z_L and propagation constant γ, the complex propagation constant γ is given by

$$\gamma = \alpha + j\beta \tag{10.221}$$

where α is the attenuation factor and β is the (real) propagation constant given by

$$\beta = \sqrt{\varepsilon_{r_{eff}}(\omega)} \cdot k_0 \tag{10.222}$$

where $\varepsilon_{r_{eff}}(\omega)$ is the effective dielectric constant and k_0 is the TEM propagation constant k_0 for the equivalent transmission line with an air dielectric.

$$k_0 = \omega\sqrt{\varepsilon_0\mu_0} \tag{10.223}$$

The expressions used to calculate the scattering parameters are given by

$$S_{11} = S_{22} = \frac{(z-y)\sinh\gamma l}{2\cosh\gamma l + (z+y)\sinh\gamma l} \tag{10.224}$$

$$S_{12} = S_{21} = \frac{2}{2\cosh\gamma l + (z+y)\sinh\gamma l} \tag{10.225}$$

with z being the normalized impedance and y is the normalized admittance.
A lossy transmission line creates noise whose noise current correlation matrix can be calculated by Bosma's theorem (see eqn. 5.45 on page 49).

10.24 Differential Transmission Line

A differential (4-port) transmission line is not referenced to ground potential, i.e. the wave from the input (port 1 and 4) is distributed to the output (port 2 and 3). Its chain parameter representation writes (for the ABCD matrix elements see equation 10.209):

$$\begin{bmatrix} . & . & A_{21} & -A_{21} & A_{22} \\ . & . & -A_{21} & A_{21} & -A_{22} \\ . & . & . & . & -1 \\ . & . & . & . & +1 \\ -1 & +1 & A_{11} & -A_{11} & A_{12} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_4 \\ V_2 \\ V_3 \\ J_2 \end{bmatrix} = \begin{bmatrix} I_1 \\ I_4 \\ I_2 \\ I_3 \\ 0 \end{bmatrix} \tag{10.226}$$

Its admittance parameters are:

$$Y_{11} = Y_{22} = Y_{33} = Y_{44} = -Y_{14} = -Y_{41} = -Y_{23} = -Y_{32} = \frac{1}{Z_L \cdot \tanh(\gamma \cdot l)} \tag{10.227}$$

$$Y_{13} = Y_{31} = Y_{24} = Y_{42} = -Y_{12} = -Y_{21} = -Y_{34} = -Y_{43} = \frac{1}{Z_L \cdot \sinh(\gamma \cdot l)} \tag{10.228}$$

The scattering parameters writes:

$$S_{11} = S_{22} = S_{33} = S_{44} = Z_L \cdot \frac{(2 \cdot Z_0 + Z_L) \cdot \exp(2 \cdot \gamma \cdot l) + (2 \cdot Z_0 - Z_L)}{(2 \cdot Z_0 + Z_L)^2 \cdot \exp(2 \cdot \gamma \cdot l) - (2 \cdot Z_0 - Z_L)^2} \tag{10.229}$$

$$S_{14} = S_{41} = S_{23} = S_{32} = 1 - S_{11} \tag{10.230}$$

$$S_{12} = S_{21} = S_{34} = S_{43} = -S_{13} = -S_{31} = -S_{24} = -S_{42} \tag{10.231}$$

$$= \frac{4 \cdot Z_L \cdot Z_0 \cdot \exp(\gamma \cdot l)}{(2 \cdot Z_0 + Z_L)^2 \cdot \exp(2 \cdot \gamma \cdot l) - (2 \cdot Z_0 - Z_L)^2} \tag{10.232}$$

Note: As already stated, this is a pure differential transmission line without ground reference. It is *not* a three-wire system. I.e. there is only one mode. The next section describes a differential line with ground reference.

10.25 Coupled transmission line

A coupled transmission line (see fig. 10.15) is described by two identical transmission line ABCD-matrices, one for the even mode (or common mode) and one for the odd mode (or differential mode).

$$V_1 = V_e + V_o \qquad V_4 = V_e - V_o \tag{10.233}$$

$$I_1 = I_e + I_o \qquad I_4 = I_e - I_o \tag{10.234}$$

or

$$V_e = \frac{1}{2} \cdot (V_1 + V_4) \qquad V_o = \frac{1}{2} \cdot (V_1 - V_4) \tag{10.235}$$

$$I_e = \frac{1}{2} \cdot (I_1 + I_4) \qquad I_o = \frac{1}{2} \cdot (I_1 - I_4) \tag{10.236}$$

Because the coupled lines are a symmetrical 3-line system, the matrices are completely independent of each other. Therefore, it gives (for the ABCD matrix elements see equation 10.209):

$$\begin{bmatrix} . & . & \frac{1}{2} \cdot (A_{21e} + A_{21o}) & \frac{1}{2} \cdot (A_{21e} - A_{21o}) & A_{22o} & A_{22e} \\ . & . & \frac{1}{2} \cdot (A_{21e} - A_{21o}) & \frac{1}{2} \cdot (A_{21e} + A_{21o}) & -A_{22o} & A_{22e} \\ . & . & . & . & -1 & -1 \\ . & . & . & . & +1 & -1 \\ -\frac{1}{2} & +\frac{1}{2} & \frac{1}{2} \cdot A_{11o} & -\frac{1}{2} \cdot A_{11o} & A_{12o} & . \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} \cdot A_{11e} & \frac{1}{2} \cdot A_{11e} & . & A_{12e} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_4 \\ V_2 \\ V_3 \\ J_o \\ J_e \end{bmatrix} = \begin{bmatrix} I_1 \\ I_4 \\ I_2 \\ I_3 \\ 0 \\ 0 \end{bmatrix} \tag{10.237}$$

Its Y-parameters write as follows.

$$Y_{11} = Y_{22} = Y_{33} = Y_{44} = \frac{1}{2 \cdot Z_{L,e} \cdot \tanh(\gamma_e \cdot l)} + \frac{1}{2 \cdot Z_{L,o} \cdot \tanh(\gamma_o \cdot l)} \tag{10.238}$$

$$Y_{12} = Y_{21} = Y_{34} = Y_{43} = \frac{-1}{2 \cdot Z_{L,e} \cdot \sinh(\gamma_e \cdot l)} + \frac{-1}{2 \cdot Z_{L,o} \cdot \sinh(\gamma_o \cdot l)} \tag{10.239}$$

$$Y_{13} = Y_{31} = Y_{24} = Y_{42} = \frac{-1}{2 \cdot Z_{L,e} \cdot \sinh(\gamma_e \cdot l)} + \frac{1}{2 \cdot Z_{L,o} \cdot \sinh(\gamma_o \cdot l)} \tag{10.240}$$

$$Y_{14} = Y_{41} = Y_{23} = Y_{32} = \frac{1}{2 \cdot Z_{L,e} \cdot \tanh(\gamma_e \cdot l)} + \frac{-1}{2 \cdot Z_{L,o} \cdot \tanh(\gamma_o \cdot l)} \tag{10.241}$$

The S-parameters (according to the port numbering in fig. 10.15) are as followed [26].

reflection coefficients

$$S_{11} = S_{22} = S_{33} = S_{44} = X_e + X_o \tag{10.242}$$

through paths

$$S_{12} = S_{21} = S_{34} = S_{43} = Y_e + Y_o \tag{10.243}$$

coupled paths

$$S_{14} = S_{41} = S_{23} = S_{32} = X_e - X_o \tag{10.244}$$

isolated paths

$$S_{13} = S_{31} = S_{24} = S_{42} = Y_e - Y_o \tag{10.245}$$

with the denominator

$$D_{e,o} = 2 \cdot Z_{L,e,o} \cdot Z_0 \cdot \cosh(\gamma_{e,o} \cdot l) + \left(Z_{L,e,o}^2 + Z_0^2\right) \cdot \sinh(\gamma_{e,o} \cdot l) \tag{10.246}$$

and

$$X_{e,o} = \frac{\left(Z_{L,e,o}^2 - Z_0^2\right) \cdot \sinh(\gamma_{e,o} \cdot l)}{2 \cdot D_{e,o}} \tag{10.247}$$

$$Y_{e,o} = \frac{Z_{L,e,o} \cdot Z_0}{D_{e,o}} \tag{10.248}$$

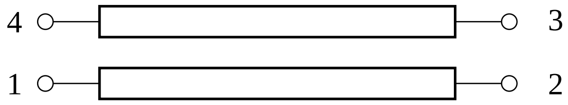


Figure 10.15: coupled transmission line

10.26 S-parameter container with additional reference port

The Y-parameters of a multi-port component defined by its S-parameters required for a small signal AC analysis can be obtained by converting the S-parameters into Y-parameters.

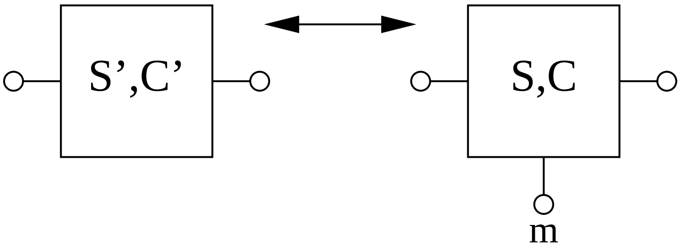


Figure 10.16: S-parameter container with noise wave correlation matrix

In order to extend a $m-1$-port to have a S-parameter device with m ports assuming that the original reference port had a reflection coefficient Γ_m the new S-parameters are according to T. O. Grosch and L. A. Carpenter [27]:

$$S_{mm} = \frac{2 - \Gamma_m - m + \sum\limits_{i=1}^{m-1} \sum\limits_{j=1}^{m-1} S'_{ij}}{1 - m \cdot \Gamma_m - \sum\limits_{i=1}^{m-1} \sum\limits_{j=1}^{m-1} S'_{ij}} \tag{10.249}$$

$$S_{im} = \left(\frac{1 - \Gamma_m \cdot S_{mm}}{1 - \Gamma_m}\right) \cdot \left(1 - \sum_{j=1}^{m-1} S'_{ij}\right) \qquad \text{for } i = 1, 2 \dots m-1 \tag{10.250}$$

$$S_{mj} = \left(\frac{1 - \Gamma_m \cdot S_{mm}}{1 - \Gamma_m}\right) \cdot \left(1 - \sum_{i=1}^{m-1} S'_{ij}\right) \qquad \text{for } j = 1, 2 \dots m-1 \tag{10.251}$$

$$S_{ij} = S'_{ij} - \left(\frac{\Gamma_m \cdot S_{im} \cdot S_{mj}}{1 - \Gamma_m \cdot S_{mm}}\right) \qquad \text{for } i, j = 1, 2 \dots m-1 \tag{10.252}$$

If the reference port has been ground potential, then Γ_m simply folds to -1. The reverse transformation by connecting a termination with a reflection coefficient of Γ_m to the mth port writes as follows.

$$S'_{ij} = S_{ij} + \left(\frac{\Gamma_m \cdot S_{im} \cdot S_{mj}}{1 - \Gamma_m \cdot S_{mm}}\right) \qquad \text{for } i, j = 1, 2 \dots m-1 \tag{10.253}$$

With the S-parameter transformation done the m-port noise wave correlation matrix is

$$C_m = \left|\frac{1}{1 - \Gamma_m}\right|^2 \cdot \left(K \cdot C_{m-1} \cdot K^+ - T_s \cdot k_B \cdot \left|1 - |\Gamma_m|^2\right| \cdot D \cdot D^+\right) \tag{10.254}$$

with

$$K = \begin{bmatrix} 1+\Gamma_m\left(S_{1m}-1\right) & \Gamma_m S_{1m} & \dots & \Gamma_m S_{1m} \\ \Gamma_m S_{2m} & 1+\Gamma_m\left(S_{2m}-1\right) & \dots & \Gamma_m S_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ \Gamma_m S_{(m-1)m} & \Gamma_m S_{(m-1)m} & \dots & 1+\Gamma_m\left(S_{(m-1)m}-1\right) \\ \Gamma_m S_{mm}-1 & \Gamma_m S_{mm}-1 & \dots & \Gamma_m S_{mm}-1 \end{bmatrix} \tag{10.255}$$

$$D = \begin{bmatrix} S_{1m} \\ S_{2m} \\ \vdots \\ S_{(m-1)m} \\ S_{mm}-1 \end{bmatrix} \tag{10.256}$$

whence T_s denotes the equivalent noise temperature of the original reference port and the $^+$ operator indicates the transposed conjugate matrix (also called adjoint or adjugate).

The reverse transformation can be written as

$$C_{m-1} = K' \cdot C_m \cdot K'^{+} + T_s \cdot k_B \cdot \frac{\left|1-\left|\Gamma_m\right|^2\right|}{\left|1-\Gamma_m S_{mm}\right|^2} \cdot D' \cdot D'^{+} \tag{10.257}$$

with

$$K' = \begin{bmatrix} 1 & 0 & \dots & 0 & \dfrac{\Gamma_m S_{1m}}{1-\Gamma_m S_{mm}} \\ 0 & 1 & \dots & 0 & \dfrac{\Gamma_m S_{2m}}{1-\Gamma_m S_{mm}} \\ . & . & \ddots & . & \vdots \\ 0 & 0 & \dots & 1 & \dfrac{\Gamma_m S_{(m-1)m}}{1-\Gamma_m S_{mm}} \end{bmatrix} \tag{10.258}$$

$$D' = \begin{bmatrix} S_{1m} \\ S_{2m} \\ \vdots \\ S_{(m-1)m} \end{bmatrix} \tag{10.259}$$

10.27 Loudspeaker

An electrodynamic loudspeaker consists of a voice-coil within the field of a magnet. An AC current flowing through this coil creates a motion that is transfered into an acoustic pressure by the diaphragm (or cone) to which the coil is attached. A complete model of a loudspeaker is very complex, but the small-signal impedance can be simulated by the following equation [28].

$$Z_{VC} = R_E + Z_L + Z_M \tag{10.260}$$

with R_E being the voice-coil DC resistance, Z_L being the impedance of the lossy inductor and Z_M models the low-frequency resonance caused by the mechanical system.

The impedance Z_L stems from the coil inductance together with the hysteresis losses of the magnetic material. It is calculated as follows.

$$Z_L = K \cdot (j \cdot \omega)^n = K \cdot \omega^n \cdot \exp\left(j \cdot \frac{n\pi}{2}\right) \tag{10.261}$$

It is $0 \leq n \leq 1$, with $n = 1$ being a lossless inductor. The parameter K has no general relation to n, but usually systems with a low value of n exhibit higher values for K. The impedance Z_M creates a resonance peak and is modeled with the following equation.

$$Z_M = \frac{R_{ES} \cdot s/\omega_S}{Q_{MS} \cdot (s/\omega_S)^2 + s/\omega_S + Q_{MS}} \tag{10.262}$$

with $s = j\omega$, ω_S is the (fundamental) resonance frequency of the driver, Q_{MS} is the mechanical quality factor and R_{ES} determines the height of the resonance peak.

As an examples a typical 10 inch loudspeaker exhibits the parameter values: $R_E = 5.08\Omega$, $n = 0.688$, $K = 0.0235$, $f_S = 35.2\text{Hz}$, $R_{ES} = 32\Omega$, $Q_{MS} = 2.8$.

10.28 Real-Life Models

Non-ideal electronic components exhibit parasitic effects. Depending on the usage, they may show a very different behaviour than the ideal ones. More precise models can sometimes be obtained from their manufacturers or vendors. However, first oder approximations exists that can give satisfactory result in many cases. A few of these simple models are presented in this chapter.

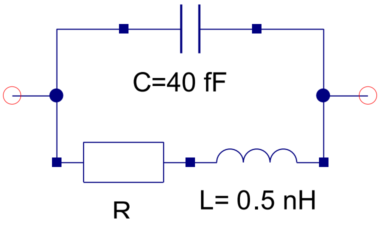


Figure 10.17: simple equivalent circuit of a 0603 resistor

A model for a resistor (case 0603) is depicted in figure 10.17. Conclusion:

- useful up to 1GHz
- values between 50Ω and 150Ω are useful up to 20GHz

The parasitics for standard SMD resistors are approximately:

- case 0201 → C=20fF, L=0.2nH
- case 0402 → C=30fF, L=0.3nH
- case 0603 → C=40fF, L=0.5nH
- case 0805 → C=50fF, L=0.6nH

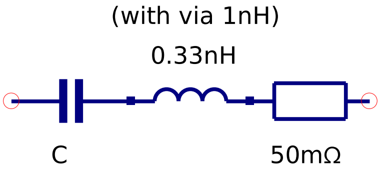


Figure 10.18: simple equivalent circuit of a 0603 ceramic capacitor

A model for a (ceramic) capacitor (case 0603) is depicted in figure 10.18. Conclusion:

- as coupling capacitor useful wide into GHz band
- as blocking capacitor a via is necessary, i.e. 10nF has resonance at about 50MHz

The parasitic inductances for SMD capacitors are approximately:

- case 0201 → 0.2nH
- case 0402 → 0.25nH
- case 0603 → 0.33nH
- case 0805 → 0.43nH

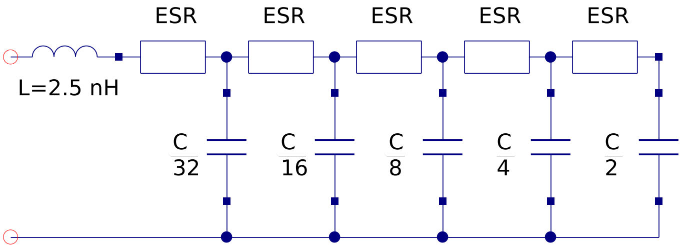


Figure 10.19: simple equivalent circuit of an electrolyte capacitor

Electrolyte capacitors are quite complicate to model. They also show the biggest differences from sample to sample. Nonetheless, figure 10.19 gives an idea how a model may look like. Conclusion:

- very broad resonance
- useful up to about 10MHz (strongly depending on capacitance)

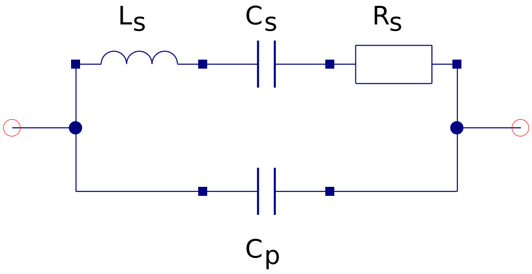


Figure 10.20: first order model of a (fundamental tone) crystal

Figure 10.20 shows a simple equivalent circuit of a crystal. The always existing but unwanted side resonances are not included here. The datasheets of crystals usually specify the quality factor Q, the equivalent series resistance R_s and the parallel capacitance C_p (and of course the frequency f_0). In most cases the quality factor can be approximated to $Q \cdot f_0 \approx 16 \cdot 10^{12}$. It corresponds to the series resonance, i.e.

$$Q = \sqrt{\frac{L_s}{C_s}} \cdot \frac{1}{R_s} \tag{10.263}$$

and therefore

$$C_s = \frac{1}{2 \cdot \pi \cdot f_0 \cdot Q \cdot R_s} \tag{10.264}$$

$$L_s = \frac{R_s \cdot Q}{2 \cdot \pi \cdot f_0} \tag{10.265}$$

Chapter 11

Non-linear devices

11.1 Operational amplifier

The ideal operational amplifier, as shown in fig. 11.1, is determined by the following equation which introduces one more unknown in the MNA matrix.

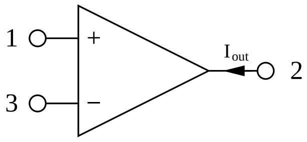


Figure 11.1: ideal operational amplifier

$$V_1 - V_3 = 0 \tag{11.1}$$

The new unknown variable I_{out} must be considered by the three remaining simple equations.

$$I_1 = 0 \quad I_2 = I_{out} \quad I_3 = 0 \tag{11.2}$$

And in matrix representation this is (for DC and AC simulation):

$$\begin{bmatrix} . & . & . & 0 \\ . & . & . & 1 \\ . & . & . & 0 \\ 1 & 0 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ 0 \end{bmatrix} \tag{11.3}$$

The operational amplifier could be considered as a special case of a voltage controlled current source with infinite forward transconductance G. Please note that the presented matrix form is only valid in cases where there is a finite feedback impedance between the output and the inverting input port.

To allow a feedback circuit to the non-inverting input (e.g. for a Schmitt trigger), one needs a limited output voltage swing. The following equations are often used to model the transmission characteristic of operational amplifiers.

$$I_1 = 0 \qquad\qquad I_3 = 0 \tag{11.4}$$

$$V_2 = V_{max} \cdot \frac{2}{\pi} \arctan\left(\frac{\pi}{2 \cdot V_{max}} \cdot G \cdot (V_1 - V_3)\right) \tag{11.5}$$

with V_{max} being the maximum output voltage swing and G the voltage amplification. To model the small-signal behaviour (AC analysis), it is necessary to differentiate:

$$g = \frac{\partial V_2}{\partial (V_1 - V_3)} = \frac{G}{1 + \left(\dfrac{\pi}{2 \cdot V_{max}} \cdot G \cdot (V_1 - V_3) \right)^2} \tag{11.6}$$

This leads to the following matrix representation being a specialised three node voltage controlled voltage source (see section 10.20.3 on page 116).

$$\begin{bmatrix} . & . & . & 0 \\ . & . & . & 1 \\ . & . & . & 0 \\ g & -1 & -g & 0 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ I_3 \\ 0 \end{bmatrix} \tag{11.7}$$

The above MNA matrix entries are also used during the non-linear DC analysis with the 0 in the right hand side vector replaced by an equivalent voltage

$$V_{eq} = g \cdot (V_1 - V_3) - V_{out} \tag{11.8}$$

with V_{out} computed using eq. (11.5).

With the given small-signal matrix representation, building the S-parameters is easy.

$$(\underline{S}) = \begin{bmatrix} 1 & 0 & 0 \\ 4g & -1 & -4g \\ 0 & 0 & 1 \end{bmatrix} \tag{11.9}$$

11.2 PN-Junction Diode

The following table contains the model parameters for the pn-junction diode model.

Name	Symbol	Description	Unit	Default
Is	I_S	saturation current	A	10^{-14}
N	N	emission coefficient		1.0
Isr	I_{SR}	recombination current parameter	A	0.0
Nr	N_R	emission coefficient for Isr		2.0
Rs	R_S	ohmic resistance	Ω	0.0
Cj0	C_{j0}	zero-bias junction capacitance	F	0.0
M	M	grading coefficient		0.5
Vj	V_j	junction potential	V	0.7
Fc	F_c	forward-bias depletion capacitance coefficient		0.5
Cp	C_p	linear capacitance	F	0.0
Tt	τ	transit time	s	0.0
Bv	B_v	reverse breakdown voltage	V	∞
Ibv	I_{bv}	current at reverse breakdown voltage	A	0.001
Ikf	I_{kf}	high-injection knee current	A	∞
Kf	K_F	flicker noise coefficient		0.0
Af	A_F	flicker noise exponent		1.0
Ffe	F_{FE}	flicker noise frequency exponent		1.0
Temp	T	device temperature	°C	26.85
Xti	X_{TI}	saturation current exponent		3.0
Eg	E_G	energy bandgap	eV	1.11
EgA	α_{EG}	temperature coefficient 1 for energy bandgap	eV/K	$7.02 \cdot 10^{-4}$

Name	Symbol	Description	Unit	Default
EgB	β_{EG}	temperature coefficient 2 for energy bandgap	K	1108
Tbv	T_{Bv}	Bv linear temperature coefficient	1/°C	0.0
Trs	T_{RS}	Rs linear temperature coefficient	1/°C	0.0
Ttt1	$T_{\tau 1}$	Tt linear temperature coefficient	1/°C	0.0
Ttt2	$T_{\tau 2}$	Tt quadratic temperature coefficient	$1/°C^2$	0.0
Tm1	T_{M1}	M linear temperature coefficient	1/°C	0.0
Tm2	T_{M2}	M quadratic temperature coefficient	$1/°C^2$	0.0
Tnom	T_{NOM}	temperature at which parameters were extracted	°C	26.85
Area	A	default area for diode		1.0

11.2.1 Large signal model

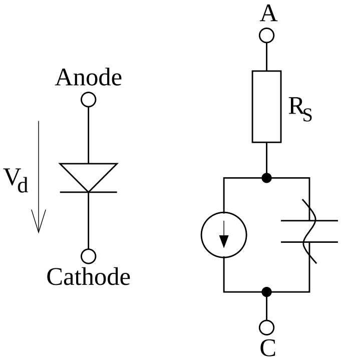


Figure 11.2: pn-junction diode symbol and large signal model

The current equation of the diode and its derivative writes as follows:

$$I_d = I_{df} + I_{dr} = I_S \cdot \left(e^{\frac{V_d}{N \cdot V_T}} - 1\right) + I_{SR} \cdot \left(e^{\frac{V_d}{N_R \cdot V_T}} - 1\right) \tag{11.10}$$

$$g_d = \frac{\partial I_d}{\partial V_d} = \frac{I_S}{N \cdot V_T} \cdot e^{\frac{V_d}{N \cdot V_T}} + \frac{I_{SR}}{N_R \cdot V_T} \cdot e^{\frac{V_d}{N_R \cdot V_T}} \tag{11.11}$$

The first term I_{df} is the normal diode current and the second term is the recombination current I_{dr}. In high-injection region the normal diode current and conductance have to be multiplied as follows.

$$I_{df,h} = I_{df} \cdot \sqrt{\frac{I_{kf}}{I_{kf} + I_{df}}} \tag{11.12}$$

$$g_{df,h} = g_{df} \cdot \left(1 - \frac{0.5 \cdot I_{df}}{I_{kf} + I_{df}}\right) \cdot \sqrt{\frac{I_{kf}}{I_{kf} + I_{df}}} \tag{11.13}$$

The complete MNA matrix entries for a dc iteration are:

$$\begin{bmatrix} g_d & -g_d \\ -g_d & g_d \end{bmatrix} \cdot \begin{bmatrix} V_C \\ V_A \end{bmatrix} = \begin{bmatrix} +I_d - g_d \cdot V_d \\ -I_d + g_d \cdot V_d \end{bmatrix} \tag{11.14}$$

11.2.2 Small signal model

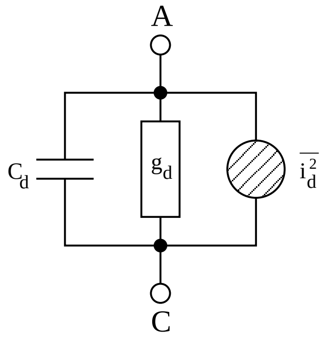


Figure 11.3: small signal model of intrinsic diode including noise

The voltage dependent capacitance (or charge) consists of a linear capacitance C_p, a diffusion capacitance $\tau \cdot g_d$ and a junction (depletion) capacitance and is usually modeled by the following equations.

$$C_d = C_p + \tau \cdot g_d + \begin{cases} C_{j0} \cdot \left(1 - \dfrac{V_d}{V_j}\right)^{-M} & \text{for } V_d/V_j \le F_c \\ \dfrac{C_{j0}}{(1-F_c)^M} \cdot \left(1 + \dfrac{M \cdot (V_d/V_j - F_c)}{1-F_c}\right) & \text{for } V_d/V_j > F_c \end{cases} \tag{11.15}$$

$$Q_d = C_p \cdot V_d + \tau \cdot I_d + \begin{cases} \dfrac{C_{j0} \cdot V_j}{1-M} \cdot \left(1 - \left(1 - \dfrac{V_d}{V_j}\right)^{1-M}\right) & \text{for } V_d/V_j \le F_c \\ C_{j0} \cdot V_j \cdot X_d & \text{for } V_d/V_j > F_c \end{cases} \tag{11.16}$$

with

$$\begin{aligned} X_d = & \frac{1-(1-F_c)^{1-M}}{1-M} \\ & + \frac{1-F_c \cdot (1+M)}{(1-F_c)^{1+M}} \cdot (V_d/V_j - F_c) + \frac{M}{2 \cdot (1-F_c)^{1+M}} \cdot \left((V_d/V_j)^2 - F_c^2\right) \end{aligned} \tag{11.17}$$

The S-parameters of the passive circuit shown in fig. 11.3 can be written as

$$S_{11} = S_{22} = \frac{1}{1 + 2 \cdot y} \tag{11.18}$$

$$S_{12} = S_{21} = 1 - S_{11} = \frac{2 \cdot y}{1 + 2 \cdot y} \tag{11.19}$$

with

$$y = Z_0 \cdot (g_d + j\omega C_d) \tag{11.20}$$

11.2.3 Noise model

The parasitic ohmic resistance in a non-ideal diode, of course, creates thermal noise that is characterized by the following spectral density.

$$\frac{\overline{i_{R_S}^2}}{\Delta f} = \frac{4k_BT}{R_S} \tag{11.21}$$

The intrinsic diode (pn- or schottky-diode) generates shot noise. Both types of current (field and diffusion) contribute independently to it. That is, even though the two currents flow in different directions (”minus” in dc current equation), they have to be added in the noise equation (current is proportional to noise power spectral density). So, it is:

$$\frac{\overline{i_d^2}}{\Delta f} = 2 \cdot e \cdot I_S \cdot \left(\exp\left(\frac{V_d}{N \cdot V_T}\right) + 1\right) = 2 \cdot e \cdot (I_d + 2 \cdot I_S) \tag{11.22}$$

Where e is charge of an electron and V_T is the temperature voltage.

To be very precise, the equation above only holds for diodes whose field and diffusion current dominate absolutely (diffusion limited diode), i.e. $N = 1$. Many diodes also generate a generation/recombination current ($N \approx 2$), which produces shot noise, too. But depending on where and how the charge carriers generate or recombine, their effective charge is somewhat smaller than e. To take this into account, one needs a further factor K. Several opinions exist according the value of K. Some say 1 and 2/3 are common values, others say $K = 1/N$ with K and N being bias dependent. Altogether the noise current correlation matrix is:

$$(\underline{C}_Y) = 2 \cdot e \cdot K \cdot (I_d + 2 \cdot I_S) \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \tag{11.23}$$

Remark: Believing the diode equation $I_D = I_S \cdot (\exp(V/(N \cdot V_T)) - 1)$ is the whole truth, it is logical to define $K = 1/N$, because at $V = 0$ the conductance g_d of the diode must create thermal noise.

Some special diodes have additional current or noise components (tunnel diodes, avalanche diodes etc.). All these mechanisms are not taken into account in equation above.

The flicker noise generated by the DC current flowing through the diode is characterized by the following spectral density.

$$\frac{\overline{i_d^2}}{\Delta f} = K_F \frac{I_d^{A_F}}{f^{F_{FE}}} \tag{11.24}$$

Taking into account the dynamic conductance g_d in parallel to the noise current source, the noise wave correlation matrix writes as follows.

$$\begin{aligned} (\underline{C}) = & \left|\frac{0.5 \cdot Y_0}{g_d + j\omega C_d + 0.5 \cdot Y_0}\right|^2 \cdot 2 \cdot e \cdot K \cdot I_S \cdot \left(\exp\left(\frac{V_d}{N \cdot V_T}\right) + 1\right) \cdot Z_0 \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \\ & = 2 \cdot e \cdot K \cdot Z_0 \cdot (I_d + 2 \cdot I_S) \cdot \left|\frac{1}{2 \cdot Z_0 \cdot (g_d + j\omega C_d) + 1}\right|^2 \cdot \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \qquad (11.25) \\ & \qquad \text{with} \quad \frac{1}{2} \le K \le 1 \end{aligned}$$

Where e is charge of an electron, V_T the temperature voltage, g_d the (dynamic) conductance of the diode and C_d its junction capacitance.

11.2.4 Reverse Breakdown

Several models were developed for the reverse breakdown region of a diode. A very common one is to use an exponential behaviour.

$$I_{br} = I_{bv} \cdot \exp\left(\frac{-B_v}{V_T}\right) \cdot \left(1 - \exp\left(\frac{-V_d}{V_T}\right)\right) \tag{11.26}$$

$$g_{br} = \frac{I_{bv}}{V_T} \cdot \exp\left(-\frac{B_v + V_d}{V_T}\right) \tag{11.27}$$

The current I_{br} is simply added to the forward current I_d of the diode.

11.2.5 Temperature model

This section mathematically describes the dependencies of the diode characterictics on temperature. For a junction diode a typical value for X_{TI} is 3.0, for a Schottky barrier diode it is 2.0. The energy band gap at zero temperature E_G is by default 1.11eV. For other materials than Si, 0.69eV (for a Schottky barrier diode), 0.67eV (for Ge) and 1.43eV (for GaAs) should be used.

$$n_i^2\left(T\right) = B \cdot T^3 \cdot e^{-E_G(T)/k_B T} \tag{11.28}$$

$$n_i\left(T\right) = 1.45 \cdot 10^{10} \cdot \left(\frac{T}{300K}\right)^{1.5} \cdot \exp\left(\frac{e \cdot E_G\left(300K\right)}{2 \cdot k_B \cdot 300K} - \frac{e \cdot E_G\left(T\right)}{2 \cdot k_B \cdot T}\right) \tag{11.29}$$

$$E_G\left(T\right) = E_G - \frac{\alpha_{EG} \cdot T^2}{\beta_{EG} + T} \tag{11.30}$$

with experimental values for Si given by

$$\alpha_{EG} = 7.02 \cdot 10^{-4}$$
$$\beta_{EG} = 1108$$
$$E_G = 1.16eV$$

The following equations show the temperature dependencies of the diode parameters. The reference temperature T_1 in these equations denotes the nominal temperature T_{NOM} specified by the diode model. (Note that V_j may become zero or negative which is not allowed. Thus, its lower boundary must be limited to a small positive value.)

$$I_S\left(T_2\right) = I_S\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{X_{TI}/N} \cdot \exp\left[-\frac{e \cdot E_G}{N \cdot k_B \cdot T_2} \cdot \left(1 - \frac{T_2}{T_1}\right)\right] \tag{11.31}$$

$$I_{SR}\left(T_2\right) = I_{SR}\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{X_{TI}/N_R} \cdot \exp\left[-\frac{e \cdot E_G}{N \cdot k_B \cdot T_2} \cdot \left(1 - \frac{T_2}{T_1}\right)\right] \tag{11.32}$$

$$V_j\left(T_2\right) = \frac{T_2}{T_1} \cdot V_j\left(T_1\right) + \frac{2 \cdot k_B \cdot T_2}{e} \cdot \ln\left(\frac{n_i\left(T_1\right)}{n_i\left(T_2\right)}\right) \tag{11.33}$$

$$= \frac{T_2}{T_1} \cdot V_j\left(T_1\right) - \frac{3 \cdot k_B \cdot T_2}{e} \cdot \ln\left(\frac{T_2}{T_1}\right) - \left(\frac{T_2}{T_1} \cdot E_G\left(T_1\right) - E_G\left(T_2\right)\right) \tag{11.34}$$

$$C_{j0}\left(T_2\right) = C_{j0}\left(T_1\right) \cdot \left(1 + M \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{V_j\left(T_2\right) - V_j\left(T_1\right)}{V_j\left(T_1\right)}\right)\right) \tag{11.35}$$

Some additionial temperature coefficients determine the temperature dependence of even more model parameters.

$$B_v\left(T_2\right) = B_v\left(T_1\right) - T_{Bv} \cdot \left(T_2 - T_1\right) \tag{11.36}$$

$$\tau\left(T_2\right) = \tau\left(T_1\right) \cdot \left(1 + T_{\tau 1} \cdot \left(T_2 - T_1\right) + T_{\tau 2} \cdot \left(T_2 - T_1\right)^2\right) \tag{11.37}$$

$$M\left(T_2\right) = M\left(T_1\right) \cdot \left(1 + T_{M1} \cdot \left(T_2 - T_1\right) + T_{M2} \cdot \left(T_2 - T_1\right)^2\right) \tag{11.38}$$

$$R_S\left(T_2\right) = R_S\left(T_1\right) \cdot \left(1 + T_{RS} \cdot \left(T_2 - T_1\right)\right) \tag{11.39}$$

11.2.6 Area dependence of the model

The area factor A used in the diode model determines the number of equivalent parallel devices of the specified model. The diode model parameters affected by the A factor are:

$$I_S\left(A\right) = I_S \cdot A \tag{11.40}$$

$$I_{bv}\left(A\right) = I_{bv} \cdot A \tag{11.41}$$

$$C_{j0}\left(A\right) = C_{j0} \cdot A \tag{11.42}$$

$$R_S\left(A\right) = \frac{R_S}{A} \tag{11.43}$$

11.3 Bipolar Junction Transistor

The following table contains the model parameters for the BJT (Spice Gummel-Poon) model.

Name	Symbol	Description	Unit	Default
Is	I_S	saturation current	A	10^{-16}
Nf	N_F	forward emission coefficient		1.0
Nr	N_R	reverse emission coefficient		1.0
Ikf	I_{KF}	high current corner for forward beta	A	∞
Ikr	I_{KR}	high current corner for reverse beta	A	∞
Vaf	V_{AF}	forward early voltage	V	∞
Var	V_{AR}	reverse early voltage	V	∞
Ise	I_{SE}	base-emitter leakage saturation current	A	0
Ne	N_E	base-emitter leakage emission coefficient		1.5
Isc	I_{SC}	base-collector leakage saturation current	A	0
Nc	N_C	base-collector leakage emission coefficient		2.0
Bf	B_F	forward beta		100
Br	B_R	reverse beta		1
Rbm	R_{Bm}	minimum base resistance for high currents	Ω	0.0
Irb	I_{RB}	current for base resistance midpoint	A	∞
Rc	R_C	collector ohmic resistance	Ω	0.0
Re	R_E	emitter ohmic resistance	Ω	0.0
Rb	R_B	zero-bias base resistance (may be high-current dependent)	Ω	0.0
Cje	C_{JE}	base-emitter zero-bias depletion capacitance	F	0.0
Vje	V_{JE}	base-emitter junction built-in potential	V	0.75
Mje	M_{JE}	base-emitter junction exponential factor		0.33
Cjc	C_{JC}	base-collector zero-bias depletion capacitance	F	0.0
Vjc	V_{JC}	base-collector junction built-in potential	V	0.75
Mjc	M_{JC}	base-collector junction exponential factor		0.33
Xcjc	X_{CJC}	fraction of Cjc that goes to internal base pin		1.0
Cjs	C_{JS}	zero-bias collector-substrate capacitance	F	0.0
Vjs	V_{JS}	substrate junction built-in potential	V	0.75
Mjs	M_{JS}	substrate junction exponential factor		0.0
Fc	F_C	forward-bias depletion capacitance coefficient		0.5
Tf	T_F	ideal forward transit time	s	0.0
Xtf	X_{TF}	coefficient of bias-dependence for Tf		0.0
Vtf	V_{TF}	voltage dependence of Tf on base-collector voltage	V	∞
Itf	I_{TF}	high-current effect on Tf	A	0.0
Ptf	φ_{TF}	excess phase at the frequency $1/(2\pi T_F)$	°	0.0
Tr	T_R	ideal reverse transit time	s	0.0
Kf	K_F	flicker noise coefficient		0.0
Af	A_F	flicker noise exponent		1.0
Ffe	F_{FE}	flicker noise frequency exponent		1.0
Kb	K_B	burst noise coefficient		0.0
Ab	A_B	burst noise exponent		1.0
Fb	F_B	burst noise corner frequency	Hz	1.0
Temp	T	device temperature	°C	26.85
Xti	X_{TI}	saturation current exponent		3.0
Xtb	X_{TB}	temperature exponent for forward- and reverse-beta		0.0
Eg	E_G	energy bandgap	eV	1.11
Tnom	T_{NOM}	temperature at which parameters were extracted	°C	26.85
Area	A	default area for bipolar transistor		1.0

11.3.1 Large signal model

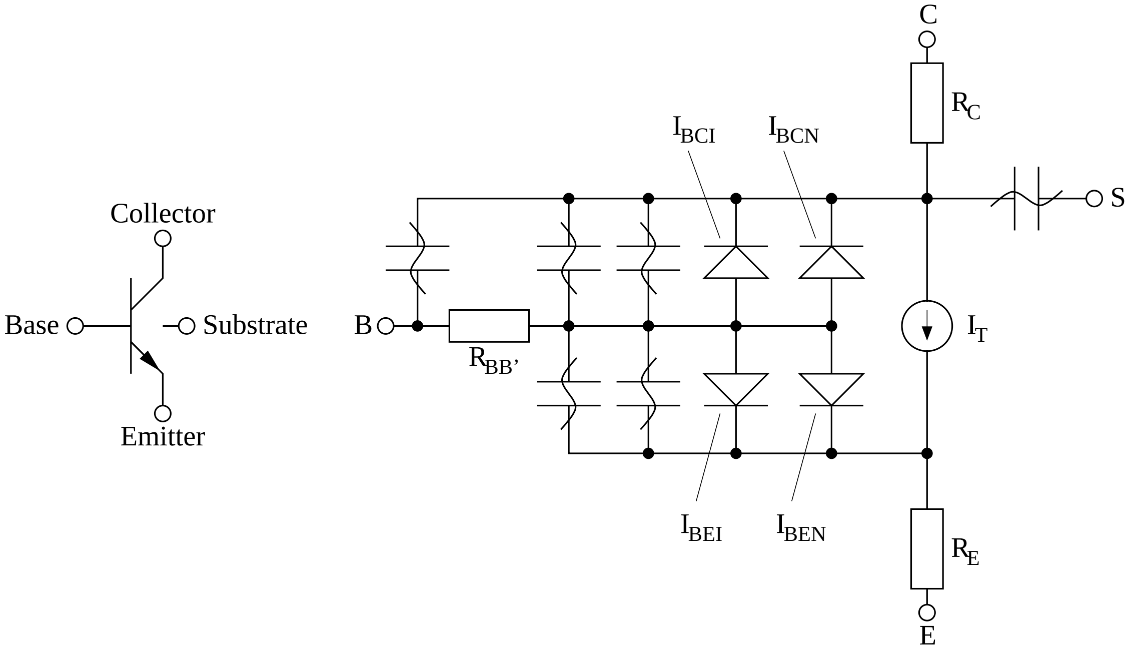


Figure 11.4: bipolar transistor symbol and large signal model for vertical device

The SGP (**S**PICE **G**ummel-**P**oon) model is basically a transport model, i.e. the voltage dependent ideal transfer currents (forward I_F and backward I_R) are reference currents in the model. The ideal base current parts are defined dependent on the ideal transfer currents. The ideal forward transfer current starts flowing when applying a positive control voltage at the base-emitter junction. It is defined by:

$$I_F = I_S \cdot \left(e^{\frac{V_{BE}}{N_F \cdot V_T}} - 1\right) \tag{11.44}$$

The ideal base current components are defined by the ideal transfer currents. The non-ideal components are independently defined by dedicated saturation currents and emission coefficients.

$$I_{BEI} = \frac{I_F}{B_F} \qquad g_{BEI} = \frac{\partial I_{BEI}}{\partial V_{BE}} = \frac{I_S}{N_F \cdot V_T \cdot B_F} \cdot e^{\frac{V_{BE}}{N_F \cdot V_T}} \tag{11.45}$$

$$I_{BEN} = I_{SE} \cdot \left(e^{\frac{V_{BE}}{N_E \cdot V_T}} - 1\right) \qquad g_{BEN} = \frac{\partial I_{BEN}}{\partial V_{BE}} = \frac{I_{SE}}{N_E \cdot V_T} \cdot e^{\frac{V_{BE}}{N_E \cdot V_T}} \tag{11.46}$$

$$I_{BE} = I_{BEI} + I_{BEN} \tag{11.47}$$

$$g_\pi = g_{BE} = g_{BEI} + g_{BEN} \tag{11.48}$$

The ideal backward transfer current arises when applying a positive control voltage at the base-collector junction (e.g. in the active inverse mode). It is defined by:

$$I_R = I_S \cdot \left(e^{\frac{V_{BC}}{N_R \cdot V_T}} - 1\right) \tag{11.49}$$

Again, the ideal base current component through the base-collector junction is defined in reference to the ideal backward transfer current and the non-ideal component is defined by a dedicated saturation current and emission coefficient.

$$I_{BCI} = \frac{I_R}{B_R} \qquad g_{BCI} = \frac{\partial I_{BCI}}{\partial V_{BC}} = \frac{I_S}{N_R \cdot V_T \cdot B_R} \cdot e^{\frac{V_{BC}}{N_R \cdot V_T}} \tag{11.50}$$

$$I_{BCN} = I_{SC} \cdot \left(e^{\frac{V_{BC}}{N_C \cdot V_T}} - 1 \right) \qquad g_{BCN} = \frac{\partial I_{BCN}}{\partial V_{BC}} = \frac{I_{SC}}{N_C \cdot V_T} \cdot e^{\frac{V_{BC}}{N_C \cdot V_T}} \tag{11.51}$$

$$I_{BC} = I_{BCI} + I_{BCN} \tag{11.52}$$

$$g_\mu = g_{BC} = g_{BCI} + g_{BCN} \tag{11.53}$$

With these definitions it is possible to calculate the overall base current flowing into the device using all the base current components.

$$I_B = I_{BE} + I_{BC} = I_{BEI} + I_{BEN} + I_{BCI} + I_{BCN} \tag{11.54}$$

The overall transfer current I_T can be calculated using the normalized base charge Q_B and the ideal forward and backward transfer currents.

$$I_T = \frac{I_F - I_R}{Q_B} \tag{11.55}$$

The normalized base charge Q_B has no dimension and has the value 1 for $V_{BE} = V_{BC} = 0$. It is used to model two effects: the influence of the base width modulation on the transfer current (Early effect) and the ideal transfer currents deviation at high currents, i.e. the decreasing current gain at high currents.

$$Q_B = \frac{Q_1}{2} \cdot \left(1 + \sqrt{1 + 4 \cdot Q_2} \right) \tag{11.56}$$

The Q_1 term is used to describe the Early effect and Q_2 is responsible for the high current effects.

$$Q_1 = \frac{1}{1 - \dfrac{V_{BC}}{V_{AF}} - \dfrac{V_{BE}}{V_{AR}}} \quad \text{and} \quad Q_2 = \frac{I_F}{I_{KF}} + \frac{I_R}{I_{KR}} \tag{11.57}$$

The transfer current I_T depends on V_{BE} and V_{BC} by the normalized base charge Q_B and the forward transfer current I_F and the backward transfer current I_R. That is why both of the partial derivatives are required.

The forward transconductance g_{mf} of the transfer current I_T is obtained by differentiating it with respect to V_{BE}. The reverse transconductance g_{mr} can be calculated by differentiating the transfer current with respect to V_{BC}.

$$g_{mf} = \frac{\partial I_T}{\partial V_{BE}} = \frac{\partial I_{TF}}{\partial V_{BE}} - \frac{\partial I_{TR}}{\partial V_{BE}} = \frac{1}{Q_B} \cdot \left(+g_{IF} - I_T \cdot \frac{\partial Q_B}{\partial V_{BE}} \right) \tag{11.58}$$

$$g_{mr} = \frac{\partial I_T}{\partial V_{BC}} = \frac{\partial I_{TF}}{\partial V_{BC}} - \frac{\partial I_{TR}}{\partial V_{BC}} = \frac{1}{Q_B} \cdot \left(-g_{IR} - I_T \cdot \frac{\partial Q_B}{\partial V_{BC}} \right) \tag{11.59}$$

With g_{IF} being the forward conductance of the ideal forward transfer current and g_{IR} being the reverse conductance of the ideal backward transfer current.

$$g_{IF} = \frac{\partial I_F}{\partial V_{BE}} = g_{BEI} \cdot B_F \tag{11.60}$$

$$g_{IR} = \frac{\partial I_R}{\partial V_{BC}} = g_{BCI} \cdot B_R \tag{11.61}$$

The remaining derivatives in eq. (11.58) and (11.59) can be written as

$$\frac{\partial Q_B}{\partial V_{BE}} = Q_1 \cdot \left(\frac{Q_B}{V_{AR}} + \frac{g_{IF}}{I_{KF} \cdot \sqrt{1 + 4 \cdot Q_2}}\right) \tag{11.62}$$

$$\frac{\partial Q_B}{\partial V_{BC}} = Q_1 \cdot \left(\frac{Q_B}{V_{AF}} + \frac{g_{IR}}{I_{KR} \cdot \sqrt{1 + 4 \cdot Q_2}}\right) \tag{11.63}$$

For the calculation of the bias dependent base resistance $R_{BB'}$ there are two different ways within the SGP model. If the model parameter I_{RB} is not given it is determined by the normalized base charge Q_B. Otherwise I_{RB} specifies the base current at which the base resistance drops half way to the minimum (i.e. the constant component) base resistance R_{Bm}. The base resistance is linear (bias independent), if R_{Bm} equals (or is greater than) R_B.

$$R_{BB'} = \begin{cases} R_{Bm} + \dfrac{R_B - R_{Bm}}{Q_B} & \text{for } I_{RB} = \infty \\ R_{Bm} + 3 \cdot (R_B - R_{Bm}) \cdot \dfrac{\tan z - z}{z \cdot \tan^2 z} & \text{for } I_{RB} \neq \infty \end{cases} \tag{11.64}$$

$$\text{with} \qquad z = \frac{\sqrt{1 + \dfrac{144}{\pi^2} \cdot \dfrac{I_B}{I_{RB}}} - 1}{\dfrac{24}{\pi^2} \cdot \sqrt{\dfrac{I_B}{I_{RB}}}} \tag{11.65}$$

The MNA matrix entries and the current vector entries are as follows.

$$\begin{bmatrix} g_\mu + g_\pi & -g_\mu & -g_\pi & 0 \\ -g_\mu + g_{mf} - g_{mr} & g_\mu + g_{mr} & -g_{mf} & 0 \\ -g_\pi - g_{mf} + g_{mr} & -g_{mr} & g_\pi + g_{mf} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} V_B \\ V_C \\ V_E \\ V_S \end{bmatrix} = \begin{bmatrix} -I_{BE_{eq}} - I_{BC_{eq}} \\ +I_{BC_{eq}} - I_{CE_{eq}} \\ +I_{BE_{eq}} + I_{CE_{eq}} \\ 0 \end{bmatrix} \tag{11.66}$$

$$I_{BE_{eq}} = I_{BE} - g_\pi \cdot V_{BE} \tag{11.67}$$

$$I_{BC_{eq}} = I_{BC} - g_\mu \cdot V_{BC} \tag{11.68}$$

$$I_{CE_{eq}} = I_T - g_{mf} \cdot V_{BE} + g_{mr} \cdot V_{BC} \tag{11.69}$$

In order to implement the influence of the excess phase parameter φ_{TF} – denoting the phase shift of the current gain at the transit frequency – the method developed by P.B. Weil and L.P. McNamee [29] can be used. They propose to use a second-order Bessel polynomial to modify the forward transfer current:

$$I_{Tx} = I_T \cdot \Phi\left(s\right) = I_T \cdot \frac{3 \cdot \omega_0^2}{s^2 + 3 \cdot \omega_0 \cdot s + 3 \cdot \omega_0^2} \tag{11.70}$$

This polynomial is formulated to closely resemble a time domain delay for a Gaussian curve which is similar to the physical phenomenon exhibited by bipolar transistor action.

Applying the inverse Laplace transformation to eq. (11.70) and using finite difference methods the transfer current can be written as

$$I_{Tx}^{n+1} = C_1 \cdot I_T^{n+1} + C_2 \cdot I_{Tx}^n - C_3 \cdot I_{Tx}^{n-1} \tag{11.71}$$

with

$$C_1 = \frac{3 \cdot \omega_0^2 \cdot \Delta t^2}{1 + 3 \cdot \omega_0 \cdot \Delta t + 3 \cdot \omega_0^2 \cdot \Delta t^2} \tag{11.72}$$

$$C_2 = \frac{2 + 3 \cdot \omega_0 \cdot \Delta t}{1 + 3 \cdot \omega_0 \cdot \Delta t + 3 \cdot \omega_0^2 \cdot \Delta t^2} \tag{11.73}$$

$$C_3 = \frac{1}{1 + 3 \cdot \omega_0 \cdot \Delta t + 3 \cdot \omega_0^2 \cdot \Delta t^2} \tag{11.74}$$

and

$$\omega_0 = \frac{\pi}{180} \cdot \frac{1}{\varphi_{TF} \cdot T_F} \tag{11.75}$$

The appropiate modified derivative writes as

$$g_{mx}^{n+1} = C_1 \cdot g_m^{n+1} \tag{11.76}$$

It should be noted that the excess phase implementation during the transient analysis (and thus in the AC analysis as well) holds for the forward part of the transfer current only.

With non-equidistant inegration time steps during transient analysis present eqs. (11.73) and (11.74) yield

$$C_2 = \frac{1 + \Delta t/\Delta t_1 + 3 \cdot \omega_0 \cdot \Delta t}{1 + 3 \cdot \omega_0 \cdot \Delta t + 3 \cdot \omega_0^2 \cdot \Delta t^2} \tag{11.77}$$

$$C_3 = \frac{\Delta t/\Delta t_1}{1 + 3 \cdot \omega_0 \cdot \Delta t + 3 \cdot \omega_0^2 \cdot \Delta t^2} \tag{11.78}$$

whereas Δt denotes the current time step and Δt_1 the previous one.

11.3.2 Small signal model

Equations for the real valued conductances in both equivalent circuits for the intrinsic BJT have already been given.

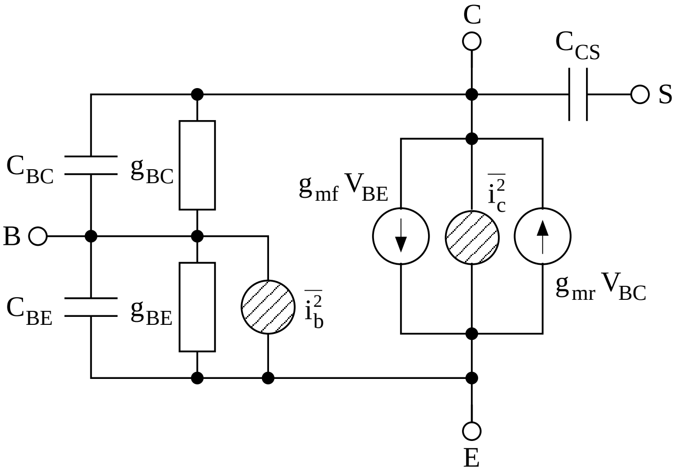


Figure 11.5: small signal model of intrinsic BJT including noise sources

The junctions depletion capacitances in the SGP model write as follows:

$$C_{BE_{dep}} = \begin{cases} C_{JE} \cdot \left(1 - \dfrac{V_{BE}}{V_{JE}}\right)^{-M_{JE}} & \text{for } V_{BE} \le F_C \cdot V_{JE} \\ \dfrac{C_{JE}}{(1 - F_C)^{M_{JE}}} \cdot \left(1 + \dfrac{M_{JE} \cdot (V_{BE} - F_C \cdot V_{JE})}{V_{JE} \cdot (1 - F_C)}\right) & \text{for } V_{BE} > F_C \cdot V_{JE} \end{cases} \tag{11.79}$$

$$C_{BC_{dep}} = \begin{cases} C_{JC} \cdot \left(1 - \dfrac{V_{BC}}{V_{JC}}\right)^{-M_{JC}} & \text{for } V_{BC} \le F_C \cdot V_{JC} \\ \dfrac{C_{JC}}{(1 - F_C)^{M_{JC}}} \cdot \left(1 + \dfrac{M_{JC} \cdot (V_{BC} - F_C \cdot V_{JC})}{V_{JC} \cdot (1 - F_C)}\right) & \text{for } V_{BC} > F_C \cdot V_{JC} \end{cases} \tag{11.80}$$

$$C_{SC_{dep}} = \begin{cases} C_{JS} \cdot \left(1 - \dfrac{V_{SC}}{V_{JS}}\right)^{-M_{JS}} & \text{for } V_{SC} \le 0 \\ C_{JS} \cdot \left(1 + M_{JS} \cdot \dfrac{V_{SC}}{V_{JS}}\right) & \text{for } V_{SC} > 0 \end{cases} \tag{11.81}$$

The base-collector depletion capacitance is split into two components: an external and an internal.

$$C_{BCI_{dep}} = X_{CJC} \cdot C_{BC_{dep}} \tag{11.82}$$

$$C_{BCX_{dep}} = (1 - X_{CJC}) \cdot C_{BC_{dep}} \tag{11.83}$$

The base-emitter diffusion capacitance can be obtained using the following equation.

$$C_{BE_{diff}} = \frac{\partial Q_{BE}}{\partial V_{BE}} \quad \text{with} \quad Q_{BE} = \frac{I_F}{Q_B} \cdot T_{FF} \tag{11.84}$$

Thus the diffusion capacitance depends on the bias-dependent effective forward transit time T_{FF} which is defined as:

$$T_{FF} = T_F \cdot \left(1 + X_{TF} \cdot \left(\frac{I_F}{I_F + I_{TF}}\right)^2 \cdot \exp\left(\frac{V_{BC}}{1.44 \cdot V_{TF}}\right)\right) \tag{11.85}$$

With

$$\frac{\partial T_{FF}}{\partial V_{BE}} = \frac{T_F \cdot X_{TF} \cdot 2 \cdot g_{IF} \cdot I_F \cdot I_{TF}}{(I_F + I_{TF})^3} \cdot \exp\left(\frac{V_{BC}}{1.44 \cdot V_{TF}}\right) \tag{11.86}$$

the base-emitter diffusion capacitance can finally be written as:

$$C_{BE_{diff}} = \frac{\partial Q_{BE}}{\partial V_{BE}} = \frac{1}{Q_B} \cdot \left(I_F \cdot \frac{\partial T_{FF}}{\partial V_{BE}} + T_{FF} \cdot \left(g_{IF} - \frac{I_F}{Q_B} \cdot \frac{\partial Q_B}{\partial V_{BE}}\right)\right) \tag{11.87}$$

Because the base-emitter charge Q_{BE} in eq. (11.84) also depends on the voltage across the base-collector junction, it is necessary to find the appropriate derivative as well:

$$C_{BE_{BC}} = \frac{\partial Q_{BE}}{\partial V_{BC}} = \frac{I_F}{Q_B} \cdot \left(\frac{\partial T_{FF}}{\partial V_{BC}} - \frac{T_{FF}}{Q_B} \cdot \frac{\partial Q_B}{\partial V_{BC}}\right) \tag{11.88}$$

which turns out to be a so called transcapacitance. It additionally requires:

$$\frac{\partial T_{FF}}{\partial V_{BC}} = \frac{T_F \cdot X_{TF}}{1.44 \cdot V_{TF}} \cdot \left(\frac{I_F}{I_F + I_{TF}}\right)^2 \cdot \exp\left(\frac{V_{BC}}{1.44 \cdot V_{TF}}\right) \tag{11.89}$$

The base-collector diffusion capacitance writes as follows:

$$C_{BC_{diff}} = \frac{\partial Q_{BC}}{\partial V_{BC}} = T_R \cdot g_{IR} \quad \text{with} \quad Q_{BC} = T_R \cdot I_R \tag{11.90}$$

To take the excess phase parameter φ_{TF} into account the forward transconductance is going to be a complex quantity.

$$g_{mf} = g_{mf} \cdot e^{-j\varphi_{ex}} \qquad \text{with} \qquad \varphi_{ex} = \left(\frac{\pi}{180} \cdot \varphi_{TF}\right) \cdot T_F \cdot 2\pi f \tag{11.91}$$

With these calculations made it is now possible to define the small signal Y-parameters of the intrinsic BJT. The Y-parameter matrix can be converted to S-parameters.

$$Y = \begin{bmatrix} Y_{BC} + Y_{BE} + Y_{BE_{BC}} & -Y_{BC} - Y_{BE_{BC}} & -Y_{BE} & 0 \\ g_{mf} - Y_{BC} - g_{mr} & Y_{CS} + Y_{BC} + g_{mr} & -g_{mf} & -Y_{CS} \\ g_{mr} - g_{mf} - Y_{BE} - Y_{BE_{BC}} & -g_{mr} + Y_{BE_{BC}} & Y_{BE} + g_{mf} & 0 \\ 0 & -Y_{CS} & 0 & Y_{CS} \end{bmatrix} \tag{11.92}$$

with

$$Y_{BC} = g_\mu + j\omega\left(C_{BCI_{dep}} + C_{BC_{diff}}\right) \tag{11.93}$$

$$Y_{BE} = g_\pi + j\omega\left(C_{BE_{dep}} + C_{BE_{diff}}\right) \tag{11.94}$$

$$Y_{CS} = j\omega \cdot C_{CS_{dep}} \tag{11.95}$$

$$Y_{BE_{BC}} = j\omega \cdot C_{BE_{BC}} \tag{11.96}$$

The external capacitance C_{BCX} connected between the internal collector node and the external base node is separately modeled if it is non-zero and if there is a non-zero base resistance.

11.3.3 Noise model

The ohmic resistances $R_{BB'}$, R_C and R_E generate thermal noise characterized by the following spectral densities.

$$\frac{\overline{i_{R_{BB'}}^2}}{\Delta f} = \frac{4k_BT}{R_{BB'}} \quad \text{and} \quad \frac{\overline{i_{R_C}^2}}{\Delta f} = \frac{4k_BT}{R_C} \quad \text{and} \quad \frac{\overline{i_{R_E}^2}}{\Delta f} = \frac{4k_BT}{R_E} \tag{11.97}$$

Shot noise, flicker noise and burst noise generated by the DC base current is characterized by the spectral density

$$\frac{\overline{i_b^2}}{\Delta f} = 2eI_{BE} + K_F\frac{I_{BE}^{A_F}}{f^{F_{FE}}} + K_B\frac{I_{BE}^{A_B}}{1 + \left(\dfrac{f}{F_B}\right)^2} \tag{11.98}$$

The shot noise generated by the DC collector to emitter current flow is characterized by the spectral density

$$\frac{\overline{i_c^2}}{\Delta f} = 2eI_T \tag{11.99}$$

The noise current correlation matrix of the four port intrinsic bipolar transistor can then be written as

$$\underline{C}_Y = \Delta f \begin{bmatrix} +\overline{i_b^2} & 0 & -\overline{i_b^2} & 0 \\ 0 & +\overline{i_c^2} & -\overline{i_c^2} & 0 \\ -\overline{i_b^2} & -\overline{i_c^2} & +\overline{i_c^2} + \overline{i_b^2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{11.100}$$

This matrix representation can be converted to the noise wave correlation matrix representation $\underline{C}_S$ using the formulas given in section 5.5.1 on page 48.

11.3.4 Temperature model

Temperature appears explicitly in the exponential term of the bipolar transistor model equations. In addition, the model parameters are modified to reflect changes in the temperature. The reference temperature T_1 in these equations denotes the nominal temperature T_{NOM} specified by the bipolar transistor model.

$$I_S\left(T_2\right) = I_S\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{X_{TI}} \cdot \exp\left[-\frac{e \cdot E_G\left(300K\right)}{k_B \cdot T_2} \cdot \left(1 - \frac{T_2}{T_1}\right)\right] \tag{11.101}$$

$$V_{JE}\left(T_2\right) = \frac{T_2}{T_1} \cdot V_{JE}\left(T_1\right) - \frac{3 \cdot k_B \cdot T_2}{e} \cdot \ln\left(\frac{T_2}{T_1}\right) - \left(\frac{T_2}{T_1} \cdot E_G\left(T_1\right) - E_G\left(T_2\right)\right) \tag{11.102}$$

$$V_{JC}\left(T_2\right) = \frac{T_2}{T_1} \cdot V_{JC}\left(T_1\right) - \frac{3 \cdot k_B \cdot T_2}{e} \cdot \ln\left(\frac{T_2}{T_1}\right) - \left(\frac{T_2}{T_1} \cdot E_G\left(T_1\right) - E_G\left(T_2\right)\right) \tag{11.103}$$

$$V_{JS}\left(T_2\right) = \frac{T_2}{T_1} \cdot V_{JS}\left(T_1\right) - \frac{3 \cdot k_B \cdot T_2}{e} \cdot \ln\left(\frac{T_2}{T_1}\right) - \left(\frac{T_2}{T_1} \cdot E_G\left(T_1\right) - E_G\left(T_2\right)\right) \tag{11.104}$$

where the $E_G\left(T\right)$ dependency has already been described in section 11.2.5 on page 132. The temperature dependence of B_F and B_R is determined by

$$B_F\left(T_2\right) = B_F\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{X_{TB}} \tag{11.105}$$

$$B_R\left(T_2\right) = B_R\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{X_{TB}} \tag{11.106}$$

Through the parameters I_{SE} and I_{SC}, respectively, the temperature dependence of the non-ideal saturation currents is determined by

$$I_{SE}\left(T_2\right) = I_{SE}\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{-X_{TB}} \cdot \left[\frac{I_S\left(T_2\right)}{I_S\left(T_1\right)}\right]^{1/N_E} \tag{11.107}$$

$$I_{SC}\left(T_2\right) = I_{SC}\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{-X_{TB}} \cdot \left[\frac{I_S\left(T_2\right)}{I_S\left(T_1\right)}\right]^{1/N_C} \tag{11.108}$$

The temperature dependence of the zero-bias depletion capacitances C_{JE}, C_{JC} and C_{JS} are determined by

$$C_{JE}\left(T_2\right) = C_{JE}\left(T_1\right) \cdot \left(1 + M_{JE} \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{V_{JE}\left(T_2\right) - V_{JE}\left(T_1\right)}{V_{JE}\left(T_1\right)}\right)\right) \tag{11.109}$$

$$C_{JC}\left(T_2\right) = C_{JC}\left(T_1\right) \cdot \left(1 + M_{JC} \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{V_{JC}\left(T_2\right) - V_{JC}\left(T_1\right)}{V_{JC}\left(T_1\right)}\right)\right) \tag{11.110}$$

$$C_{JS}\left(T_2\right) = C_{JS}\left(T_1\right) \cdot \left(1 + M_{JS} \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{V_{JS}\left(T_2\right) - V_{JS}\left(T_1\right)}{V_{JS}\left(T_1\right)}\right)\right) \tag{11.111}$$

11.3.5 Area dependence of the model

The area factor A used in the bipolar transistor model determines the number of equivalent parallel devices of a specified model. The bipolar transistor model parameters affected by the A factor are:

$$I_S\left(A\right) = I_S \cdot A \tag{11.112}$$

$$I_{SE}\left(A\right) = I_{SE} \cdot A \qquad I_{SC}\left(A\right) = I_{SC} \cdot A \tag{11.113}$$

$$I_{KF}\left(A\right) = I_{KF} \cdot A \qquad I_{KR}\left(A\right) = I_{KR} \cdot A \tag{11.114}$$

$$I_{RB}\left(A\right) = I_{RB} \cdot A \qquad I_{TF}\left(A\right) = I_{TF} \cdot A \tag{11.115}$$

$$C_{JE}\left(A\right) = C_{JE} \cdot A \qquad C_{JC}\left(A\right) = C_{JC} \cdot A \tag{11.116}$$
$$C_{JS}\left(A\right) = C_{JS} \cdot A \tag{11.117}$$

$$R_B\left(A\right) = \frac{R_B}{A} \qquad R_{Bm}\left(A\right) = \frac{R_{Bm}}{A} \tag{11.118}$$
$$R_E\left(A\right) = \frac{R_E}{A} \qquad R_C\left(A\right) = \frac{R_C}{A} \tag{11.119}$$

11.4 Junction FET

The following table contains the model parameters for the JFET model.

Name	Symbol	Description	Unit	Default
Vt0	V_{Th}	zero -bias threshold voltage	V	-2.0
Beta	β	transconductance parameter	A/V^2	10^{-4}
Lambda	λ	channel-length modulation parameter	1/V	0.0
Rd	R_D	drain ohmic resistance	Ω	0.0
Rs	R_S	source ohmic resistance	Ω	0.0
Is	I_S	gate-junction saturation current	A	10^{-14}
N	N	gate P-N emission coefficient		1.0
Isr	I_{SR}	gate-junction recombination current parameter	A	0.0
Nr	N_R	Isr emission coefficient		2.0
Cgs	C_{gs}	zero-bias gate-source junction capacitance	F	0.0
Cgd	C_{gd}	zero-bias gate-drain junction capacitance	F	0.0
Pb	P_b	gate-junction potential	V	1.0
Fc	F_c	forward-bias junction capacitance coefficient		0.5
M	M	gate P-N grading coefficient		0.5
Kf	K_F	flicker noise coefficient		0.0
Af	A_F	flicker noise exponent		1.0
Ffe	F_{FE}	flicker noise frequency exponent		1.0
Temp	T	device temperature	°C	26.85
Xti	X_{TI}	saturation current exponent		3.0
Vt0tc	$V_{Th_{TC}}$	Vt0 temperature coefficient	V/°C	0.0
Betatce	β_{TCE}	Beta exponential temperature coefficient	%/°C	0.0
Tnom	T_{NOM}	temperature at which parameters were extracted	°C	26.85
Area	A	default area for JFET		1.0

11.4.1 Large signal model

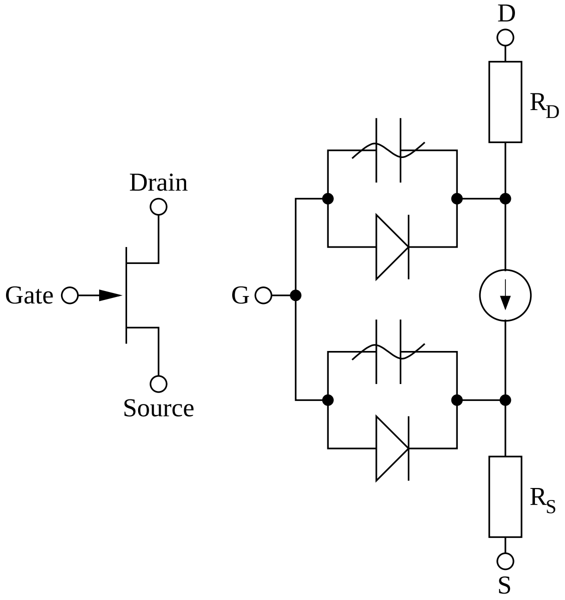


Figure 11.6: junction FET symbol and large signal model

The current equation of the gate source diode and its derivative writes as follows:

$$I_{GS} = I_S \cdot \left(e^{\frac{V_{GS}}{N \cdot V_T}} - 1\right) + I_{SR} \cdot \left(e^{\frac{V_{GS}}{N_R \cdot V_T}} - 1\right) \tag{11.120}$$

$$g_{gs} = \frac{\partial I_{GS}}{\partial V_{GS}} = \frac{I_S}{N \cdot V_T} \cdot e^{\frac{V_{GS}}{N \cdot V_T}} + \frac{I_{SR}}{N_R \cdot V_T} \cdot e^{\frac{V_{GS}}{N_R \cdot V_T}} \tag{11.121}$$

The current equation of the gate drain diode and its derivative writes as follows:

$$I_{GD} = I_S \cdot \left(e^{\frac{V_{GD}}{N \cdot V_T}} - 1\right) + I_{SR} \cdot \left(e^{\frac{V_{GD}}{N_R \cdot V_T}} - 1\right) \tag{11.122}$$

$$g_{gd} = \frac{\partial I_{GD}}{\partial V_{GD}} = \frac{I_S}{N \cdot V_T} \cdot e^{\frac{V_{GD}}{N \cdot V_T}} + \frac{I_{SR}}{N_R \cdot V_T} \cdot e^{\frac{V_{GD}}{N_R \cdot V_T}} \tag{11.123}$$

Both equations contain the gate-junction saturation current I_S, the gate P-N emission coefficient N and the temperature voltage V_T with the Boltzmann's constant k_B and the electron charge q. The operating temperature T must be specified in Kelvin.

$$V_T = \frac{k_B \cdot T}{q} \tag{11.124}$$

The controlled drain currents have been defined by Shichman and Hodges [30] for different modes of operations.

$$g_m = \frac{\partial I_d}{\partial V_{GS}} \quad \text{and} \quad g_{ds} = \frac{\partial I_d}{\partial V_{DS}} \quad \text{with} \quad V_{GD} = V_{GS} - V_{DS} \tag{11.125}$$

- normal mode: $V_{DS} > 0$
  - normal mode, cutoff region: $V_{GS} - V_{Th} < 0$

$$I_d = 0 \tag{11.126}$$

$$g_m = 0 \tag{11.127}$$

$$g_{ds} = 0 \tag{11.128}$$

- normal mode, saturation region: $0 < V_{GS} - V_{Th} < V_{DS}$

$$I_d = \beta \cdot (1 + \lambda V_{DS}) \cdot (V_{GS} - V_{Th})^2 \tag{11.129}$$

$$g_m = \beta \cdot (1 + \lambda V_{DS}) \cdot 2\,(V_{GS} - V_{Th}) \tag{11.130}$$

$$g_{ds} = \beta \cdot \lambda\,(V_{GS} - V_{Th})^2 \tag{11.131}$$

- normal mode, linear region: $V_{DS} < V_{GS} - V_{Th}$

$$I_d = \beta \cdot (1 + \lambda V_{DS}) \cdot (2\,(V_{GS} - V_{Th}) - V_{DS}) \cdot V_{DS} \tag{11.132}$$

$$g_m = \beta \cdot (1 + \lambda V_{DS}) \cdot 2 \cdot V_{DS} \tag{11.133}$$

$$g_{ds} = \beta \cdot (1 + \lambda V_{DS}) \cdot 2\,(V_{GS} - V_{Th} - V_{DS}) + \beta \cdot \lambda V_{DS} \cdot (2\,(V_{GS} - V_{Th}) - V_{DS}) \tag{11.134}$$

- inverse mode: $V_{DS} < 0$
  In inverse mode the same equations as in normal mode holds, but the following things must be changed: Replace V_{GS} with V_{GD}, change the sign of I_{DS} and of g_m, place g_m according to its new dependency (V_{GD}) into the MNA matrix.

Applying the rules for creating the MNA matrix of an arbitrary network the complete MNA matrix entries (admittance matrix and current vector) for the intrinsic junction FET are:

$$\begin{bmatrix} g_{gd} + g_{gs} & -g_{gd} & -g_{gs} \\ -g_{gd} + g_m & g_{ds} + g_{gd} & -g_{ds} - g_m \\ -g_{gs} - g_m & -g_{ds} & g_{gs} + g_{ds} + g_m \end{bmatrix} \cdot \begin{bmatrix} V_G \\ V_D \\ V_S \end{bmatrix} = \begin{bmatrix} -I_{GD_{eq}} - I_{GS_{eq}} \\ +I_{GD_{eq}} - I_{DS_{eq}} \\ +I_{GS_{eq}} + I_{DS_{eq}} \end{bmatrix} \tag{11.135}$$

with

$$I_{GS_{eq}} = I_{GS} - g_{gs} \cdot V_{GS} \tag{11.136}$$

$$I_{GD_{eq}} = I_{GD} - g_{gd} \cdot V_{GD} \tag{11.137}$$

$$I_{DS_{eq}} = I_d - g_m \cdot V_{GS} - g_{ds} \cdot V_{DS} \tag{11.138}$$

11.4.2 Small signal model

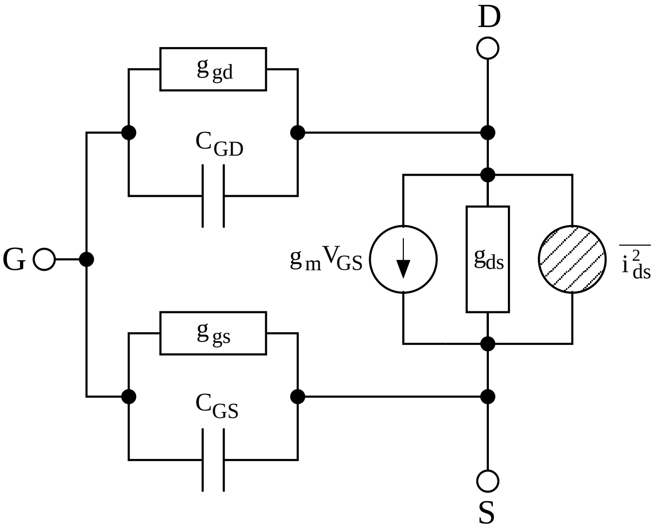


Figure 11.7: small signal model of intrinsic junction FET with noise source

The small signal Y-parameter matrix of the intrinsic junction FET writes as follows. It can be converted to S-parameters.

$$Y = \begin{bmatrix} Y_{GD} + Y_{GS} & -Y_{GD} & -Y_{GS} \\ g_m - Y_{GD} & Y_{GD} + Y_{DS} & -Y_{DS} - g_m \\ -g_m - Y_{GS} & -Y_{DS} & Y_{GS} + Y_{DS} + g_m \end{bmatrix} \tag{11.139}$$

with

$$Y_{GD} = g_{gd} + j\omega C_{GD} \tag{11.140}$$

$$Y_{GS} = g_{gs} + j\omega C_{GS} \tag{11.141}$$

$$Y_{DS} = g_{ds} \tag{11.142}$$

The junction capacitances are modeled with the following equations.

$$C_{GD} = \begin{cases} C_{gd} \cdot \left(1 - \dfrac{V_{GD}}{P_b}\right)^{-M} & \text{for } V_{GD} \le F_c \cdot P_b \\ \dfrac{C_{gd}}{(1 - F_c)^M} \cdot \left(1 + \dfrac{M \cdot (V_{GD} - F_c \cdot P_b)}{P_b \cdot (1 - F_c)}\right) & \text{for } V_{GD} > F_c \cdot P_b \end{cases} \tag{11.143}$$

$$C_{GS} = \begin{cases} C_{gs} \cdot \left(1 - \dfrac{V_{GS}}{P_b}\right)^{-M} & \text{for } V_{GS} \le F_c \cdot P_b \\ \dfrac{C_{gs}}{(1 - F_c)^M} \cdot \left(1 + \dfrac{M \cdot (V_{GS} - F_c \cdot P_b)}{P_b \cdot (1 - F_c)}\right) & \text{for } V_{GS} > F_c \cdot P_b \end{cases} \tag{11.144}$$

11.4.3 Noise model

Both the drain and source resistance R_D and R_S generate thermal noise characterized by the following spectral density.

$$\frac{\overline{i_{R_D}^2}}{\Delta f} = \frac{4k_BT}{R_D} \quad \text{and} \quad \frac{\overline{i_{R_S}^2}}{\Delta f} = \frac{4k_BT}{R_S} \tag{11.145}$$

Channel noise and flicker noise generated by the DC transconductance g_m and current flow from drain to source is characterized by the following spectral density.

$$\frac{\overline{i_{ds}^2}}{\Delta f} = \frac{8 k_B T g_m}{3} + K_F \frac{I_{DS}^{A_F}}{f^{F_{FE}}} \tag{11.146}$$

The noise current correlation matrix (admittance representation) of the intrinsic junction FET can be expressed by

$$\underline{C}_Y = \Delta f \begin{bmatrix} 0 & 0 & 0 \\ 0 & +\overline{i_{ds}^2} & -\overline{i_{ds}^2} \\ 0 & -\overline{i_{ds}^2} & +\overline{i_{ds}^2} \end{bmatrix} \tag{11.147}$$

This matrix representation can be easily converted to the noise-wave representation $\underline{C}_S$ if the small signal S-parameter matrix is known.

11.4.4 Temperature model

Temperature appears explicitly in the exponential terms of the JFET model equations. In addition, saturation current, gate-junction potential and zero-bias junction capacitances have built-in temperature dependence.

$$I_S\left(T_2\right) = I_S\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{X_{TI}/N} \cdot \exp\left[-\frac{e \cdot E_G\left(300K\right)}{N \cdot k_B \cdot T_2} \cdot \left(1 - \frac{T_2}{T_1}\right)\right] \tag{11.148}$$

$$I_{SR}\left(T_2\right) = I_{SR}\left(T_1\right) \cdot \left(\frac{T_2}{T_1}\right)^{X_{TI}/N_R} \cdot \exp\left[-\frac{e \cdot E_G\left(300K\right)}{N_R \cdot k_B \cdot T_2} \cdot \left(1 - \frac{T_2}{T_1}\right)\right] \tag{11.149}$$

$$P_b\left(T_2\right) = \frac{T_2}{T_1} \cdot P_b\left(T_1\right) - \frac{3 \cdot k_B \cdot T_2}{e} \cdot \ln\left(\frac{T_2}{T_1}\right) - \left(\frac{T_2}{T_1} \cdot E_G\left(T_1\right) - E_G\left(T_2\right)\right) \tag{11.150}$$

$$C_{gs}\left(T_2\right) = C_{gs}\left(T_1\right) \cdot \left(1 + M \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{P_b\left(T_2\right) - P_b\left(T_1\right)}{P_b\left(T_1\right)}\right)\right) \tag{11.151}$$

$$C_{gd}\left(T_2\right) = C_{gd}\left(T_1\right) \cdot \left(1 + M \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{P_b\left(T_2\right) - P_b\left(T_1\right)}{P_b\left(T_1\right)}\right)\right) \tag{11.152}$$

where the $E_G\left(T\right)$ dependency has already been described in section 11.2.5 on page 132. Also the threshold voltage as well as the transconductance parameter have a temperature dependence determined by

$$V_{Th}\left(T_2\right) = V_{Th}\left(T_1\right) + V_{Th_{TC}} \cdot \left(T_2 - T_1\right) \tag{11.153}$$

$$\beta\left(T_2\right) = \beta\left(T_1\right) \cdot 1.01^{\beta_{TCE} \cdot \left(T_2 - T_1\right)} \tag{11.154}$$

11.4.5 Area dependence of the model

The area factor A used for the JFET model determines the number of equivalent parallel devices of a specified model. The following parameters are affected by the area factor.

$$\beta\left(A\right) = \beta \cdot A \qquad I_S\left(A\right) = I_S \cdot A \tag{11.155}$$

$$R_D\left(A\right) = \frac{R_D}{A} \qquad R_S\left(A\right) = \frac{R_S}{A} \tag{11.156}$$

$$C_{gs}\left(A\right) = C_{gs} \cdot A \qquad C_{gd}\left(A\right) = C_{gd} \cdot A \tag{11.157}$$

11.5 MOS Field-Effect Transistor

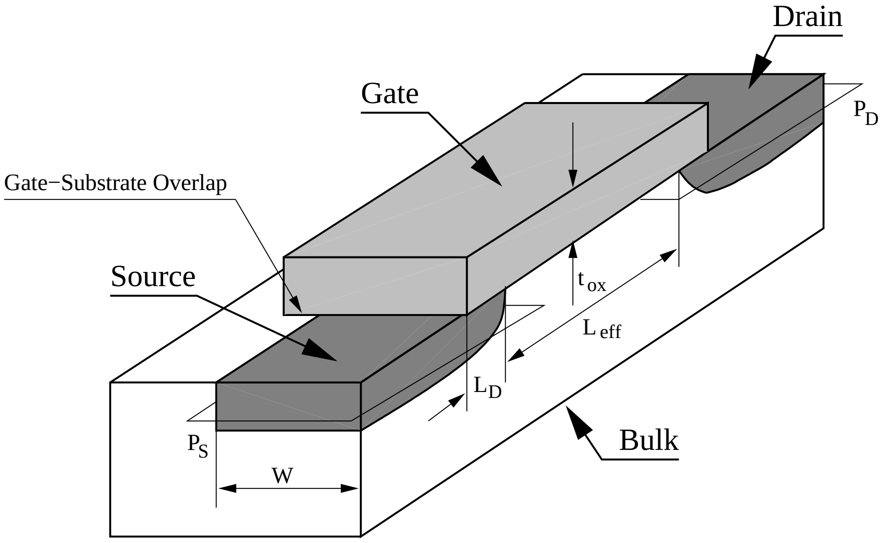


Figure 11.8: vertical section of integrated MOSFET

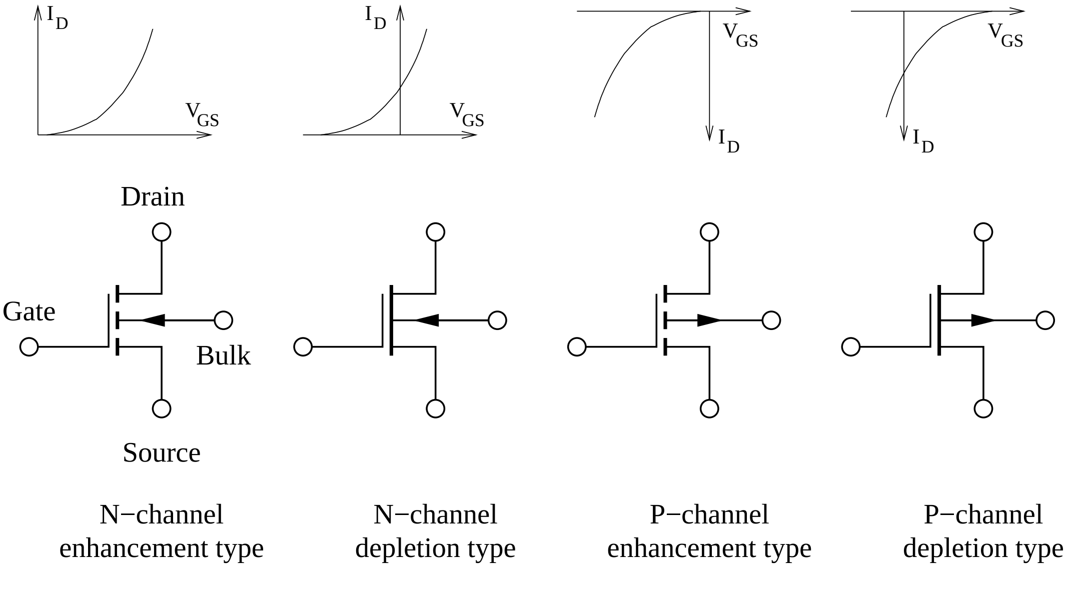


Figure 11.9: four types of MOS field effect transistors and their symbols

There are four different types of MOS field effect transistors as shown in fig. 11.9 all covered by the model going to be explained here. The "First Order Model" is a physical model with the drain current equations according to Harold Shichman and David A. Hodges [30].

The following table contains the model and device parameters for the MOSFET level 1.

Name	Symbol	Description	Unit	Default	Typical
Is	I_S	bulk junction saturation current	A	10^{-14}	10^{-15}
N	N	bulk junction emission coefficient		1.0	
Vt0	V_{T0}	zero-bias threshold voltage	V	0.0	0.7
Lambda	λ	channel-length modulation parameter	1/V	0.0	0.02

Name	Symbol	Description	Unit	Default	Typical
Kp	K_P	transconductance coefficient	A/V^2	$2 \cdot 10^{-5}$	$6 \cdot 10^{-5}$
Gamma	γ	bulk threshold	$\sqrt{V}$	0.0	0.37
Phi	Φ	surface potential	V	0.6	0.65
Rd	R_D	drain ohmic resistance	Ω	0.0	1.0
Rs	R_S	source ohmic resistance	Ω	0.0	1.0
Rg	R_G	gate ohmic resistance	Ω	0.0	
L	L	channel length	m	100µ	
Ld	L_D	lateral diffusion length	m	0.0	10^{-7}
W	W	channel width	m	100µ	
Tox	T_{OX}	oxide thickness	m	0.1µ	$2 \cdot 10^{-8}$
Cgso	C_{GSO}	gate-source overlap capacitance per meter of channel width	F/m	0.0	$4 \cdot 10^{-11}$
Cgdo	C_{GDO}	gate-drain overlap capacitance per meter of channel width	F/m	0.0	$4 \cdot 10^{-11}$
Cgbo	C_{GBO}	gate-bulk overlap capacitance per meter of channel length	F/m	0.0	$2 \cdot 10^{-10}$
Cbd	C_{BD}	zero-bias bulk-drain junction capacitance	F	0.0	$6 \cdot 10^{-17}$
Cbs	C_{BS}	zero-bias bulk-source junction capacitance	F	0.0	$6 \cdot 10^{-17}$
Pb	Φ_B	bulk junction potential	V	0.8	0.87
Mj	M_J	bulk junction bottom grading coefficient		0.5	0.5
Fc	F_C	bulk junction forward-bias depletion capacitance coefficient		0.5	
Cjsw	C_{JSW}	zero-bias bulk junction periphery capacitance per meter of junction perimeter	F/m	0.0	
Mjsw	M_{JSW}	bulk junction periphery grading coefficient		0.33	0.33
Tt	T_T	bulk transit time	s	0.0	
Kf	K_F	flicker noise coefficient		0.0	
Af	A_F	flicker noise exponent		1.0	
Ffe	F_{FE}	flicker noise frequency exponent		1.0	
Nsub	N_{SUB}	substrate (bulk) doping density	$1/cm^3$	0.0	$4 \cdot 10^{15}$
Nss	N_{SS}	surface state density	$1/cm^2$	0.0	10^{10}
Tpg	T_{PG}	gate material type (0 = alumina, -1 = same as bulk, 1 = opposite to bulk)		1	
Uo	μ_0	surface mobility	cm^2/Vs	600.0	400.0
Rsh	R_{SH}	drain and source diffusion sheet resistance	Ω/square	0.0	10.0
Nrd	N_{RD}	number of equivalent drain squares		1	
Nrs	N_{RS}	number of equivalent source squares		1	
Cj	C_J	zero-bias bulk junction bottom capacitance per square meter of junction area	F/m^2	0.0	$2 \cdot 10^{-4}$
Js	J_S	bulk junction saturation current per square meter of junction area	A/m^2	0.0	10^{-8}
Ad	A_D	drain diffusion area	m^2	0.0	
As	A_S	source diffusion area	m^2	0.0	
Pd	P_D	drain junction perimeter	m	0.0	
Ps	P_S	source junction perimeter	m	0.0	
Temp	T	device temperature	°C	26.85	
Tnom	T_{NOM}	parameter measurement temperature	°C	26.85	

11.5.1 Large signal model

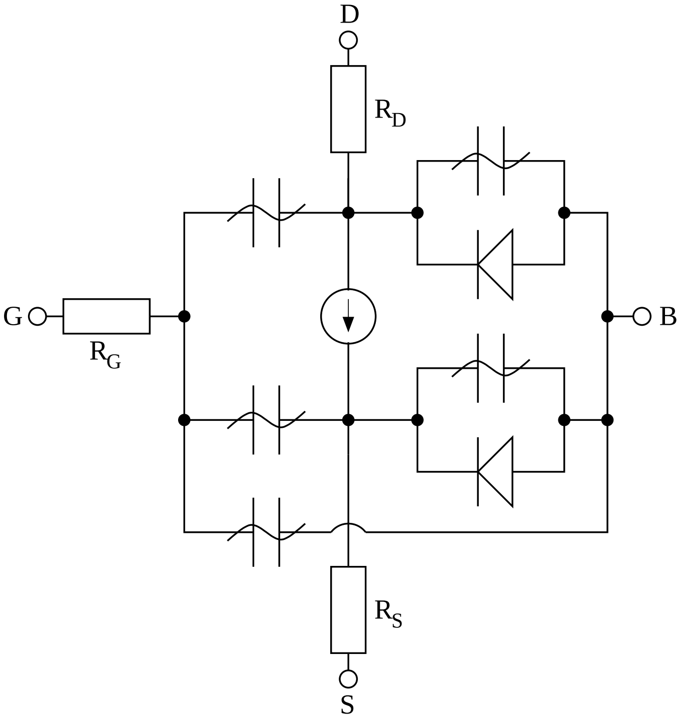


Figure 11.10: n-channel MOSFET large signal model

Beforehand some useful abbreviation are made to simplify the DC current equations.

$$L_{eff} = L - 2 \cdot L_D \tag{11.158}$$

$$\beta = K_P \cdot \frac{W}{L_{eff}} \tag{11.159}$$

The bias-dependent threshold voltage depends on the bulk-source voltage V_{BS} or the bulk-drain voltage V_{BD} depending on the mode of operation.

$$V_{Th} = V_{T0} + \begin{cases} \gamma \cdot \left(\sqrt{\Phi - V_{BS}} - \sqrt{\Phi}\right) & \text{for } V_{DS} \geq 0, \text{ i.e. } V_{BS} \geq V_{BD} \\ \gamma \cdot \left(\sqrt{\Phi - V_{BD}} - \sqrt{\Phi}\right) & \text{for } V_{DS} < 0, \text{ i.e. } V_{BD} > V_{BS} \end{cases} \tag{11.160}$$

The following equations describe the DC current behaviour of a N-channel MOSFET in normal mode, i.e. $V_{DS} > 0$, according to Shichman and Hodges.

- cutoff region: $V_{GS} - V_{Th} < 0$

$$I_d = 0 \tag{11.161}$$

$$g_{ds} = 0 \tag{11.162}$$

$$g_m = 0 \tag{11.163}$$

$$g_{mb} = 0 \tag{11.164}$$

- saturation region: $0 < V_{GS} - V_{Th} < V_{DS}$

$$I_d = \beta/2 \cdot (1 + \lambda V_{DS}) \cdot (V_{GS} - V_{Th})^2 \tag{11.165}$$
$$g_{ds} = \beta/2 \cdot \lambda (V_{GS} - V_{Th})^2 \tag{11.166}$$
$$g_m = \beta \cdot (1 + \lambda V_{DS}) (V_{GS} - V_{Th}) \tag{11.167}$$
$$g_{mb} = g_m \cdot \frac{\gamma}{2\sqrt{\Phi - V_{BS}}} \tag{11.168}$$

- linear region: $V_{DS} < V_{GS} - V_{Th}$

$$I_d = \beta \cdot (1 + \lambda V_{DS}) \cdot (V_{GS} - V_{Th} - V_{DS}/2) \cdot V_{DS} \tag{11.169}$$
$$g_{ds} = \beta \cdot (1 + \lambda V_{DS}) \cdot (V_{GS} - V_{Th} - V_{DS}) + \beta \cdot \lambda V_{DS} \cdot (V_{GS} - V_{Th} - V_{DS}/2) \tag{11.170}$$
$$g_m = \beta \cdot (1 + \lambda V_{DS}) \cdot V_{DS} \tag{11.171}$$
$$g_{mb} = g_m \cdot \frac{\gamma}{2\sqrt{\Phi - V_{BS}}} \tag{11.172}$$

with

$$g_{ds} = \frac{\partial I_d}{\partial V_{DS}} \quad \text{and} \quad g_m = \frac{\partial I_d}{\partial V_{GS}} \quad \text{and} \quad g_{mb} = \frac{\partial I_d}{\partial V_{BS}} \tag{11.173}$$

In the inverse mode of operation, i.e. $V_{DS} < 0$, the same equations can be applied with the following modifications. Replace V_{BS} with V_{BD}, V_{GS} with V_{GD} and V_{DS} with $-V_{DS}$. The drain current I_d gets reversed. Furthermore the transconductances alter their controlling nodes, i.e.

$$g_m = \frac{\partial I_d}{\partial V_{GD}} \quad \text{and} \quad g_{mb} = \frac{\partial I_d}{\partial V_{BD}} \tag{11.174}$$

The current equations of the two parasitic diodes at the bulk node and their derivatives write as follows.

$$I_{BD} = I_{SD} \cdot \left(e^{\frac{V_{BD}}{N \cdot V_T}} - 1 \right) \qquad g_{bd} = \frac{\partial I_{BD}}{\partial V_{BD}} = \frac{I_{SD}}{N \cdot V_T} \cdot e^{\frac{V_{BD}}{N \cdot V_T}} \tag{11.175}$$
$$I_{BS} = I_{SS} \cdot \left(e^{\frac{V_{BS}}{N \cdot V_T}} - 1 \right) \qquad g_{bs} = \frac{\partial I_{BS}}{\partial V_{BS}} = \frac{I_{SS}}{N \cdot V_T} \cdot e^{\frac{V_{BS}}{N \cdot V_T}} \tag{11.176}$$

with

$$I_{SD} = I_S \quad \text{and} \quad I_{SS} = I_S \tag{11.177}$$

Now it is possible to form the MNA matrix and the current vector of the intrinsic MOSFET device.

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ g_m & g_{ds} + g_{bd} & -g_{ds} - g_m - g_{mb} & g_{mb} - g_{bd} \\ -g_m & -g_{ds} & g_{bs} + g_{ds} + g_m + g_{mb} & -g_{bs} - g_{mb} \\ 0 & -g_{bd} & -g_{bs} & g_{bs} + g_{bd} \end{bmatrix} \cdot \begin{bmatrix} V_G \\ V_D \\ V_S \\ V_B \end{bmatrix} = \begin{bmatrix} 0 \\ +I_{BD_{eq}} - I_{DS_{eq}} \\ +I_{BS_{eq}} + I_{DS_{eq}} \\ -I_{BD_{eq}} - I_{BS_{eq}} \end{bmatrix} \tag{11.178}$$

$$I_{BD_{eq}} = I_{BD} - g_{bd} \cdot V_{BD} \tag{11.179}$$
$$I_{BS_{eq}} = I_{BS} - g_{bs} \cdot V_{BS} \tag{11.180}$$
$$I_{DS_{eq}} = I_d - g_m \cdot V_{GS} - g_{mb} \cdot V_{BS} - g_{ds} \cdot V_{DS} \tag{11.181}$$

11.5.2 Physical model

There are electrical parameters as well as physical and geometry parameters in the set of model parameters for the MOSFETs “First Order Model”. Some of the electrical parameters can be derived from the geometry and physical parameters.

The oxide capacitance per square meter of the channel area can be computed as

$$C'_{ox} = \varepsilon_0 \cdot \frac{\varepsilon_{ox}}{T_{ox}} \qquad \text{with} \qquad \varepsilon_{ox} = \varepsilon_{SiO_2} = 3.9 \tag{11.182}$$

Then the overall oxide capacitance can be written as

$$C_{ox} = C'_{ox} \cdot W \cdot L_{eff} \tag{11.183}$$

The transconductance coefficient K_P can be calculated using

$$K_P = \mu_0 \cdot C'_{ox} \tag{11.184}$$

The surface potential Φ is given by (with temperature voltage V_T)

$$\Phi = 2 \cdot V_T \cdot \ln\left(\frac{N_{SUB}}{n_i}\right) \qquad \text{with the intrinsic density } \; n_i = 1.45 \cdot 10^{16} 1/\text{m}^3 \tag{11.185}$$

Equation (11.185) holds for acceptor concentrations N_A (N_{SUB}) essentially greater than the donor concentration N_D. The bulk threshold γ (also sometimes called the body effect coefficient) is

$$\gamma = \frac{\sqrt{2 \cdot e \cdot \varepsilon_{Si} \cdot \varepsilon_0 \cdot N_{SUB}}}{C'_{ox}} \qquad \text{with} \qquad \varepsilon_{Si} = 11.7 \tag{11.186}$$

And finally the zero-bias threshold voltage V_{T0} writes as follows.

$$V_{T0} = V_{FB} + \Phi + \gamma \cdot \sqrt{\Phi} \tag{11.187}$$

Whereas V_{FB} denotes the flat band voltage consisting of the work function difference Φ_{MS} between the gate and substrate material and an additional potential due to the oxide surface charge.

$$V_{FB} = \Phi_{MS} - \frac{e \cdot N_{SS}}{C'_{ox}} \tag{11.188}$$

The temperature dependent bandgap potential E_G of silicon (substrate material Si) writes as follows. With $T = 290$K the bandgap is approximately $1.12eV$.

$$E_G\left(T\right) = 1.16 - \frac{7.02 \cdot 10^{-4} \cdot T^2}{T + 1108} \tag{11.189}$$

The work function difference Φ_{MS} gets computed dependent on the gate conductor material. This can be either alumina ($\Phi_M = 4.1eV$), n-polysilicon ($\Phi_M \approx 4.15eV$) or p-polysilicon ($\Phi_M \approx 5.27eV$). The work function of a semiconductor, which is the energy difference between the vacuum level and the Fermi level (see fig. 11.11), varies with the doping concentration.

$$\Phi_{MS} = \Phi_M - \Phi_S = \Phi_M - \left(4.15 + \frac{1}{2}E_G + \frac{1}{2}\Phi\right) \tag{11.190}$$

$$\Phi_M = \begin{cases} 4.1 & \text{for } T_{PG} = +0, \text{ i.e. alumina} \\ 4.15 & \text{for } T_{PG} = +1, \text{ i.e. opposite to bulk} \\ 4.15 + E_G & \text{for } T_{PG} = -1, \text{ i.e. same as bulk} \end{cases} \tag{11.191}$$

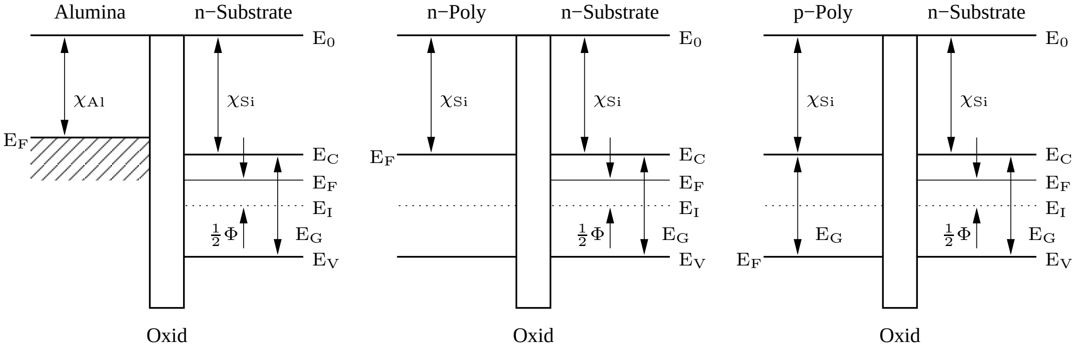


Figure 11.11: energy band diagrams of isolated (flat band) MOS materials

The expression in eq. (11.190) is visualized in fig. 11.11. The abbreviations denote

χ_{Al}	electron affinity of alumina $= 4.1eV$
χ_{Si}	electron affinity of silicon $= 4.15eV$
E_0	vacuum level
E_C	conduction band
E_V	valence band
E_F	Fermi level
E_I	intrinsic Fermi level
E_G	bandgap of silicon $\approx 1.12eV$ at room temperature

Please note that the potential $1/2 \cdot \Phi$ is positive in p-MOS and negative in n-MOS as the following equation reveals.

$$\Phi_F = \frac{E_F - E_I}{e} \tag{11.192}$$

When the gate conductor material is a heavily doped polycrystalline silicon (also called polysilicon) then the model assumes that the Fermi level of this semiconductor is the same as the conduction band (for n-poly) or the valence band (for p-poly). In alumina the Fermi level, valence and conduction band all equal the electron affinity.

If the zero-bias bulk junction bottom capacitance per square meter of junction area C_J is not given it can be computed as follows.

$$C_J = \sqrt{\frac{\varepsilon_{Si} \cdot \varepsilon_0 \cdot e \cdot N_{SUB}}{2 \cdot \Phi_B}} \tag{11.193}$$

That's it for the physical parameters. The geometry parameters account for the electrical parameters per length, area or volume. Thus the MOS model is scalable.

The diffusion resistances at drain and gate are computed as follows. The sheet resistance R_{SH} refers to the thickness of the diffusion area.

$$R_D = N_{RD} \cdot R_{SH} \quad \text{and} \quad R_S = N_{RS} \cdot R_{SH} \tag{11.194}$$

If the bulk junction saturation current per square meter of the junction area J_S and the drain and source areas are given the according saturation currents are calculated with the following equations.

$$I_{SD} = A_D \cdot J_S \quad \text{and} \quad I_{SS} = A_S \cdot J_S \tag{11.195}$$

If the parameters C_{BD} and C_{BS} are not given the zero-bias depletion capacitances for the bottom and sidewall capacitances are computed as follows.

$$C_{BD} = C_J \cdot A_D \tag{11.196}$$

$$C_{BS} = C_J \cdot A_S \tag{11.197}$$

$$C_{BDS} = C_{JSW} \cdot P_D \tag{11.198}$$

$$C_{BSS} = C_{JSW} \cdot P_S \tag{11.199}$$

11.5.3 Small signal model

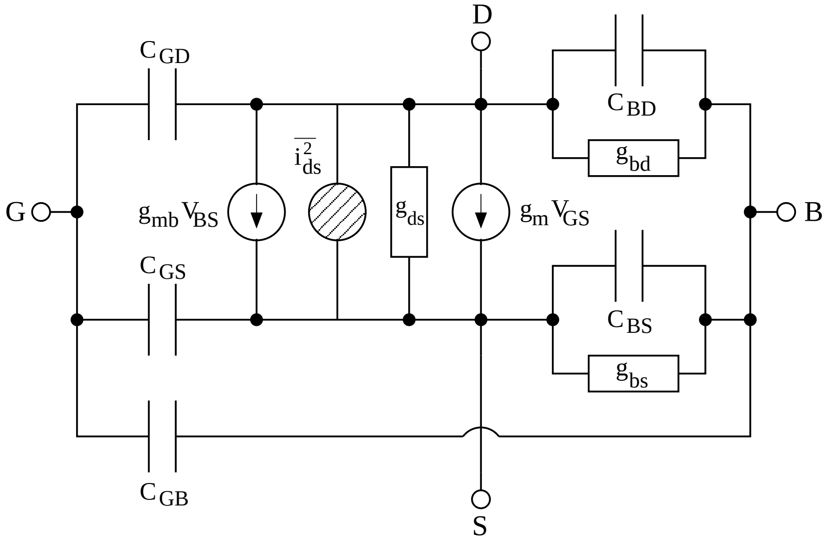


Figure 11.12: small signal model of intrinsic MOSFET with noise source

The bulk-drain and bulk-source capacitances in the MOSFET model split into three parts: the junctions depletion capacitance which consists of an area and a sidewall part and the diffusion capacitance.

$$C_{BD_{dep}} = \begin{cases} C_{BD} \cdot \left(1 - \dfrac{V_{BD}}{\Phi_B}\right)^{-M_J} & \text{for } V_{BD} \le F_C \cdot \Phi_B \\ \dfrac{C_{BD}}{(1-F_C)^{M_J}} \cdot \left(1 + \dfrac{M_J \cdot (V_{BD} - F_C \cdot \Phi_B)}{\Phi_B \cdot (1 - F_C)}\right) & \text{for } V_{BD} > F_C \cdot \Phi_B \end{cases} \tag{11.200}$$

$$C_{BDS_{dep}} = \begin{cases} C_{BDS} \cdot \left(1 - \dfrac{V_{BD}}{\Phi_B}\right)^{-M_{JSW}} & \text{for } V_{BD} \le F_C \cdot \Phi_B \\ \dfrac{C_{BDS}}{(1-F_C)^{M_{JSW}}} \cdot \left(1 + \dfrac{M_{JSW} \cdot (V_{BD} - F_C \cdot \Phi_B)}{\Phi_B \cdot (1 - F_C)}\right) & \text{for } V_{BD} > F_C \cdot \Phi_B \end{cases} \tag{11.201}$$

$$C_{BS_{dep}} = \begin{cases} C_{BS} \cdot \left(1 - \dfrac{V_{BS}}{\Phi_B}\right)^{-M_J} & \text{for } V_{BS} \le F_C \cdot \Phi_B \\ \dfrac{C_{BS}}{(1-F_C)^{M_J}} \cdot \left(1 + \dfrac{M_J \cdot (V_{BS} - F_C \cdot \Phi_B)}{\Phi_B \cdot (1 - F_C)}\right) & \text{for } V_{BS} > F_C \cdot \Phi_B \end{cases} \tag{11.202}$$

$$C_{BSS_{dep}} = \begin{cases} C_{BSS} \cdot \left(1 - \dfrac{V_{BS}}{\Phi_B}\right)^{-M_{JSW}} & \text{for } V_{BS} \le F_C \cdot \Phi_B \\ \dfrac{C_{BSS}}{(1-F_C)^{M_{JSW}}} \cdot \left(1 + \dfrac{M_{JSW} \cdot (V_{BS} - F_C \cdot \Phi_B)}{\Phi_B \cdot (1 - F_C)}\right) & \text{for } V_{BS} > F_C \cdot \Phi_B \end{cases} \tag{11.203}$$

The diffusion capacitances of the bulk-drain and bulk-source junctions are determined by the transit time of the minority charges through the junction.

$$C_{BD_{diff}} = g_{bd} \cdot T_T \tag{11.204}$$

$$C_{BS_{diff}} = g_{bs} \cdot T_T \tag{11.205}$$

Charge storage in the MOSFET consists of capacitances associated with parasitics and the intrinsic device. Parasitic capacitances consist of three constant overlap capacitances. The intrinsic capacitances consist of the nonlinear thin-oxide capacitance, which is distributed among the gate, drain, source and bulk regions. The MOS gate capacitances, as a nonlinear function of the terminal voltages, are modeled by J.E. Meyer's piece-wise linear model [31].

The Meyer model [31] was the first widely used one for the bias-dependent gate-oxide capacitances. But nowadays it's obsolete as it doesn't conserve charge and as its capacitance equations aren't continuous. The Yang-Chatterjee model solves these problems. With $V_{Dsat} = V_{GS} - V_{Th}$ it writes as follows.

- cutoff regions: $V_{GS} - V_{Th} < 0$
  - $V_{GS} - V_{Th} \le -\Phi$

$$C_{GS} = 0 \tag{11.206}$$

$$C_{GD} = 0 \tag{11.207}$$

$$C_{GB} = C_{ox} \tag{11.208}$$

  - $-\Phi < V_{GS} - V_{Th} \le -\Phi/2$

$$C_{GS} = 0 \tag{11.209}$$

$$C_{GD} = 0 \tag{11.210}$$

$$C_{GB} = -C_{ox} \cdot \frac{V_{GS} - V_{Th}}{\Phi} \tag{11.211}$$

- $-\Phi/2 < V_{GS} - V_{Th} \leq 0$

$$C_{GS} = \frac{2}{3} \cdot C_{ox} + \frac{4}{3} \cdot C_{ox} \cdot \frac{V_{GS} - V_{Th}}{\Phi} \tag{11.212}$$

$$C_{GD} = 0 \tag{11.213}$$

$$C_{GB} = -C_{ox} \cdot \frac{V_{GS} - V_{Th}}{\Phi} \tag{11.214}$$

- saturation region: $0 < V_{GS} - V_{Th} < V_{DS}$

$$C_{GS} = \frac{2}{3} \cdot C_{ox} \tag{11.215}$$

$$C_{GD} = 0 \tag{11.216}$$

$$C_{GB} = 0 \tag{11.217}$$

- linear region: $V_{DS} < V_{GS} - V_{Th}$

$$Q_{gs} = C_{ox} \cdot \left(\frac{-V_{ds}^2}{24 \cdot \left(V_{gs} - V_{th} - \frac{V_{ds}}{2}\right)} + \frac{V_{gs} - V_{th}}{2} + \frac{1}{4} \cdot V_{ds} \right) \tag{11.218}$$

$$\frac{\partial Q_{gs}}{\partial V_{gs}} = C_{ox} \cdot \left(\frac{V_{ds}^2}{24 \cdot \left(V_{gs} - V_{th} - \frac{V_{ds}}{2}\right)^2} + \frac{1}{2} \right) \tag{11.219}$$

$$\frac{\partial Q_{gs}}{\partial V_{ds}} = C_{ox} \cdot \left(\frac{-4 \cdot V_{ds} \cdot (V_{gs} - V_{th}) + V_{ds}^2}{48 \cdot \left(V_{gs} - V_{th} - \frac{V_{ds}}{2}\right)^2} + \frac{1}{4} \right) \tag{11.220}$$

$$Q_{gd} = C_{ox} \cdot \left(\frac{V_{ds}^2}{8 \cdot \left(V_{gd} - V_{th} + \frac{V_{ds}}{2}\right)} + \frac{V_{gd} - V_{th} + V_{ds}}{2} - \frac{3}{4} \cdot V_{ds} \right) \tag{11.221}$$

$$\frac{\partial Q_{gd}}{\partial V_{gs}} = C_{ox} \cdot \left(\frac{-V_{ds}^2}{8 \cdot \left(V_{gs} - V_{th} - \frac{V_{ds}}{2}\right)^2} + \frac{1}{2} \right) \tag{11.222}$$

$$\frac{\partial Q_{gd}}{\partial V_{ds}} = C_{ox} \cdot \left(\frac{4 \cdot V_{ds} \cdot (V_{gd} - V_{th}) + V_{ds}^2}{16 \cdot \left(V_{gd} - V_{th} - \frac{V_{ds}}{2}\right)^2} - \frac{1}{4} \right) \tag{11.223}$$

$$C_{GB} = 0 \tag{11.224}$$

In the inverse mode of operation V_{GS} and V_{GD} need to be exchanged, V_{DS} changes its sign, then the above formulas can be applied as well.

The constant overlap capacitances compute as follows.

$$C_{GS_{OVL}} = C_{GSO} \cdot W \tag{11.225}$$

$$C_{GD_{OVL}} = C_{GDO} \cdot W \tag{11.226}$$

$$C_{GB_{OVL}} = C_{GBO} \cdot L_{eff} \tag{11.227}$$

With these definitions it is possible to form the small signal Y-parameter matrix of the intrinsic MOSFET device in an operating point which can be converted into S-parameters.

$$Y = \begin{bmatrix} \begin{array}{c} Y_{GS} + \\ Y_{GD} + Y_{GB} \end{array} & -Y_{GD} & -Y_{GS} & -Y_{GB} \\ g_m - Y_{GD} & Y_{GD} + Y_{BD} + Y_{DS} & -Y_{DS} - g_m - g_{mb} & -Y_{BD} + g_{mb} \\ -g_m - Y_{GS} & -Y_{DS} & \begin{array}{c} Y_{GS} + Y_{DS} + \\ Y_{BS} + g_m + g_{mb} \end{array} & -Y_{BS} - g_{mb} \\ -Y_{GB} & -Y_{BD} & -Y_{BS} & Y_{BD} + Y_{BS} + Y_{GB} \end{bmatrix} \tag{11.228}$$

with

$$Y_{GS} = j\omega \left(C_{GS} + C_{GS_{OVL}}\right) \tag{11.229}$$

$$Y_{GD} = j\omega \left(C_{GD} + C_{GD_{OVL}}\right) \tag{11.230}$$

$$Y_{GB} = j\omega \left(C_{GB} + C_{GB_{OVL}}\right) \tag{11.231}$$

$$Y_{BD} = g_{bd} + j\omega \left(C_{BD_{dep}} + C_{BDS_{dep}} + C_{BD_{diff}}\right) \tag{11.232}$$

$$Y_{BS} = g_{bs} + j\omega \left(C_{BS_{dep}} + C_{BSS_{dep}} + C_{BS_{diff}}\right) \tag{11.233}$$

$$Y_{DS} = g_{ds} \tag{11.234}$$

11.5.4 Noise model

The thermal noise generated by the external resistors R_G, R_S and R_D is characterized by the following spectral density.

$$\frac{\overline{i_{R_G}^2}}{\Delta f} = \frac{4k_B T}{R_G} \quad \text{and} \quad \frac{\overline{i_{R_D}^2}}{\Delta f} = \frac{4k_B T}{R_D} \quad \text{and} \quad \frac{\overline{i_{R_S}^2}}{\Delta f} = \frac{4k_B T}{R_S} \tag{11.235}$$

Channel and flicker noise generated by the DC transconductance g_m and current flow from drain to source is characterized by the spectral density

$$\frac{\overline{i_{ds}^2}}{\Delta f} = \frac{8k_B T g_m}{3} + K_F \frac{I_{DS}^{A_F}}{f^{F_{FE}}} \tag{11.236}$$

The noise current correlation matrix (admittance representation) of the intrinsic MOSFET can be expressed as

$$\underline{C}_Y = \Delta f \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & +\overline{i_{ds}^2} & -\overline{i_{ds}^2} & 0 \\ 0 & -\overline{i_{ds}^2} & +\overline{i_{ds}^2} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{11.237}$$

This matrix representation can be easily converted to the noise-wave representation $\underline{C}_S$ if the small signal S-parameter matrix is known.

11.5.5 Temperature model

Temperature affects some MOS model parameters which are updated according to the new temperature. The reference temperature T_1 in the following equations denotes the nominal temperature T_{NOM} specified by the MOS transistor model. The temperature dependence of K_P and μ_0 is determined by

$$K_P\left(T_2\right) = K_P\left(T_1\right) \cdot \left(\frac{T_1}{T_2}\right)^{1.5} \tag{11.238}$$

$$\mu_0\left(T_2\right) = \mu_0\left(T_1\right) \cdot \left(\frac{T_1}{T_2}\right)^{1.5} \tag{11.239}$$

The effect of temperature on Φ_B and Φ is modeled by

$$\Phi\left(T_2\right) = \frac{T_2}{T_1} \cdot \Phi\left(T_1\right) - \frac{3 \cdot k_B \cdot T_2}{e} \cdot \ln\left(\frac{T_2}{T_1}\right) - \left(\frac{T_2}{T_1} \cdot E_G\left(T_1\right) - E_G\left(T_2\right)\right) \tag{11.240}$$

where the $E_G\left(T\right)$ dependency has already been described in section 11.2.5 on page 132. The temperature dependence of C_{BD}, C_{BS}, C_J and C_{JSW} is described by the following relations

$$C_{BD}\left(T_2\right) = C_{BD}\left(T_1\right) \cdot \left(1 + M_J \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{\Phi_B\left(T_2\right) - \Phi_B\left(T_1\right)}{\Phi_B\left(T_1\right)}\right)\right) \tag{11.241}$$

$$C_{BS}\left(T_2\right) = C_{BS}\left(T_1\right) \cdot \left(1 + M_J \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{\Phi_B\left(T_2\right) - \Phi_B\left(T_1\right)}{\Phi_B\left(T_1\right)}\right)\right) \tag{11.242}$$

$$C_J\left(T_2\right) = C_J\left(T_1\right) \cdot \left(1 + M_J \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{\Phi_B\left(T_2\right) - \Phi_B\left(T_1\right)}{\Phi_B\left(T_1\right)}\right)\right) \tag{11.243}$$

$$C_{JSW}\left(T_2\right) = C_{JSW}\left(T_1\right) \cdot \left(1 + M_{JSW} \cdot \left(400 \cdot 10^{-6} \cdot \left(T_2 - T_1\right) - \frac{\Phi_B\left(T_2\right) - \Phi_B\left(T_1\right)}{\Phi_B\left(T_1\right)}\right)\right) \tag{11.244}$$

The temperature dependence of I_S is given by the relation

$$I_S\left(T_2\right) = I_S\left(T_1\right) \cdot \exp\left[-\frac{e}{k_B \cdot T_2} \cdot \left(\frac{T_2}{T_1} \cdot E_G\left(T_1\right) - E_G\left(T_2\right)\right)\right] \tag{11.245}$$

An analogue dependence holds for J_S.

11.6 Thyristors

11.6.1 Diac

A diac is a bidirectional trigger diode normally used to set the switching of a triac to a specific level. A diac switches to the "on" state when the voltage across its terminals exceeds the break-over voltage U_{bo}. At this point the current I_{bo} is flowing through the device. The diac switches to the "off" state when its current I_D falls below the holding current I_{bo}. Figure 11.13 shows the equivalent circuit and its IV characteristic. As can be seen an internal node is needed. The device equations are:

$$I_D = I_S \cdot \left(\exp\left(\frac{U_D}{U_t}\right) - 1\right) - I_S \cdot \left(\exp\left(\frac{-U_D}{U_t}\right) - 1\right) \tag{11.246}$$

$$\text{with} \qquad U_t = U_{t,on} = n \cdot \frac{k \cdot T}{e} \qquad \text{if} \quad U_i > I_{bo} \cdot R_i \tag{11.247}$$

$$U_t = U_{t,off} = \frac{U_{bo}}{\ln\left(\dfrac{I_{bo}}{I_S}\right)} \qquad \text{if} \quad U_i \le I_{bo} \cdot R_i \tag{11.248}$$

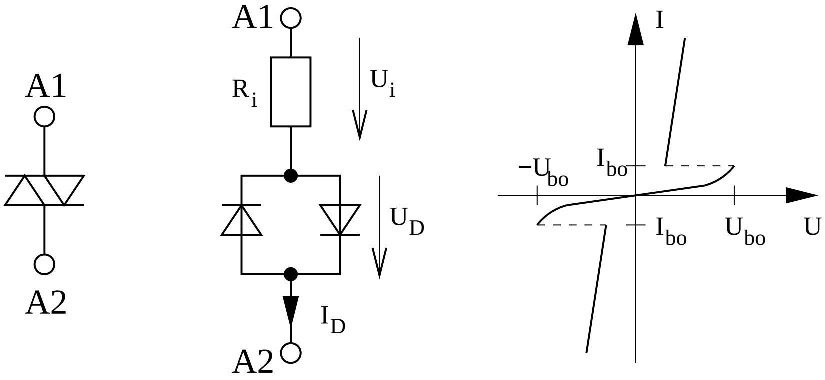


Figure 11.13: Diac symbol (left), equivalent circuit (middle) and IV characteristic (right)

Because the voltage-current characteristic is not monotone, the device encounters many convergence problems. Therefore, the following additional rules should be added.

- The exponent function is replaced by the limited exponential function:

$$\text{limexp}(x) = \begin{cases} \exp(x) & \text{for} \quad x < 80 \\ \exp(80) \cdot (1 + x - 80) & \text{for} \quad x \geq 80 \end{cases} \tag{11.249}$$

- In transienten simulation, the switching does never occur during an iteration, but when the switching rule is valid for the voltage of the previous time step.

- The switching at time t_s does not happen abrupt, but take a time of $\Delta t = 1\mu s$. During this time U_t changes linearly from "on" to "off" state:

$$U_t = (U_{t,off} - U_{t,on}) \cdot \frac{t - t_s}{\Delta t} + U_{t,on} \tag{11.250}$$

  or from "off" to "on" state, respectively:

$$U_t = (U_{t,on} - U_{t,off}) \cdot \frac{t - t_s}{\Delta t} + U_{t,off} \tag{11.251}$$

11.6.2 Triac

A triac is a bidirectional diode that can be switched on via a gate terminal. It is switched off when the current through it falls below the holding current. Figure 11.14 shows its symbol, equivalent circuit and iv characteristics. The model equations are as follows:

$$I_D = I_S \cdot \left(\exp\left(\frac{U_D}{U_t}\right) - 1 \right) - I_S \cdot \left(\exp\left(\frac{-U_D}{U_t}\right) - 1 \right) \tag{11.252}$$

$$\text{with} \qquad U_t = n \cdot \frac{k \cdot T}{e} \tag{11.253}$$

$$U_{D,bo} = U_t \cdot \ln\left(\frac{I_{bo}}{I_S}\right) \tag{11.254}$$

$$R_i = \begin{cases} R_{i,on} & \text{if} \quad |U_D| \geq U_{D,bo} \\ \frac{U_{br}}{I_{bo}} & \text{if} \quad |U_D| < U_{D,bo} \end{cases} \tag{11.255}$$

The resistance $R_{i,on}$ during switched-on state and the breakdown voltage U_{br} are model parameters. The convergence problems of the diac are also valid for the triac. Therefore, the same rules must be applied here:

- The limited exponential function must be used.
- In transienten simulation, the switching does never occur during an iteration, but when the switching rule is valid for the voltage of the previous time step.
- The switching (at t_s) takes a time of $\Delta t = 1 \mu s$. During this time R_i changes linearly from "on" to "off" state:

  $$R_i = (R_{i,off} - R_{i,on}) \cdot \frac{t - t_s}{\Delta t} + R_{i,on} \tag{11.256}$$

  or from "off" to "on" state, respectively:

  $$R_i = (R_{i,on} - R_{i,off}) \cdot \frac{t - t_s}{\Delta t} + R_{i,off} \tag{11.257}$$

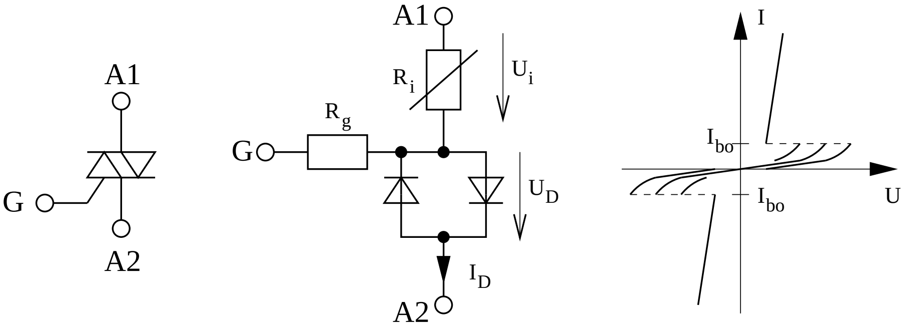


Figure 11.14: Triac symbol (left), equivalent circuit (middle) and IV characteristic (right)

11.6.3 SCR

An SCR (silicon controlled rectifier) is a uni-directional diode that can be switched on via a gate terminal. It is switched off when the current through it falls below the holding current. Figure 11.15 shows its symbol, equivalent circuit and iv characteristics. The model equations are the ones of a uni-directional triac:

$$I_D = I_S \cdot \left(\exp\left(\frac{U_D}{U_t} \right) - 1 \right) \tag{11.258}$$

$$\text{with} \qquad U_t = n \cdot \frac{k \cdot T}{e} \tag{11.259}$$

$$U_{D,bo} = U_t \cdot \ln\left(\frac{I_{bo}}{I_S} + 1 \right) \tag{11.260}$$

$$R_i = \begin{cases} R_{i,on} & \text{if} \quad U_D \geq U_{D,bo} \\ \frac{U_{br}}{I_{bo}} & \text{if} \quad U_D < U_{D,bo} \end{cases} \tag{11.261}$$

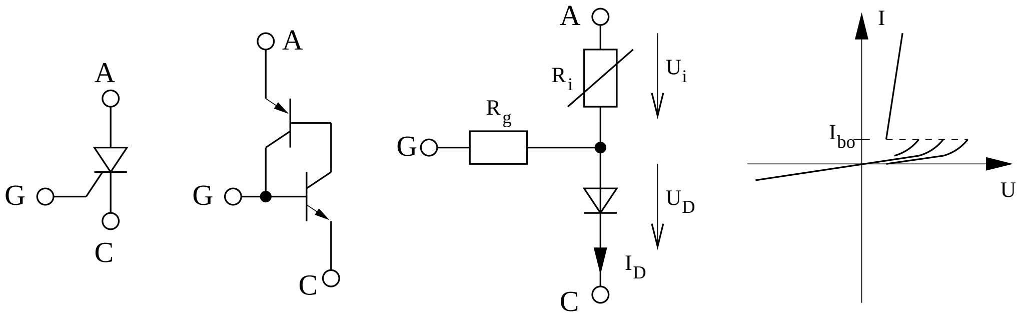


Figure 11.15: SCR; from left to right: symbol, transistor model, equivalent circuit, IV characteristic

11.7 Models for boolean devices

Logical (boolean) functions (OR, AND, XOR etc.) can be modeled using macro models. Here, each input gets the transfer characteristic and its derivative described as follows:

$$u_i = \tanh(10 \cdot (u_{in} - 0.5)) \tag{11.262}$$

$$u_i' = 10 \cdot \left(1 - \tanh^2(10 \cdot (u_{in} - 0.5))\right) \tag{11.263}$$

The resulting voltages u_i for each input are combined to create the wanted function for a device with N inputs:

$$\text{Inverter:} \quad u_{out} = 0.5 \cdot (1 - u_i) \tag{11.264}$$

$$\text{NOR:} \quad u_{out} = \frac{N}{\sum\limits_m \dfrac{2}{1 - u_{i,m}}} \tag{11.265}$$

$$\text{OR:} \quad u_{out} = 1 - u_{out,NOR} \tag{11.266}$$

$$\text{AND:} \quad u_{out} = \frac{N}{\sum\limits_m \dfrac{2}{1 + u_{i,m}}} \tag{11.267}$$

$$\text{NAND:} \quad u_{out} = 1 - u_{out,AND} \tag{11.268}$$

$$\text{XOR:} \quad u_{out} = 0.5 \cdot \left(1 - \prod_m -u_{i,m}\right) \tag{11.269}$$

$$\text{XNOR:} \quad u_{out} = 0.5 \cdot \left(1 + \prod_m u_{i,m}\right) \tag{11.270}$$

The above-mentioned functions model devices with 0V as logical low-level and 1V as logical high-level. Of course, they can be easily transformed into higher voltage levels by multiplying the desired high-level voltage to the output voltage u_{out} and dividing the input voltages u_{in} by the desired high-level voltage. Note: The derivatives also get u_{in} divided by the desired high-level voltage, but they are not multiplied by the desired high-level voltage.

To perform a simulation on these devices, the first derivatives are also needed:

$$\text{Inverter:}\quad \frac{\partial u_{out}}{\partial u_{in}} = -0.5 \cdot u'_i \tag{11.271}$$

$$\text{OR:}\quad \frac{\partial u_{out}}{\partial u_{in,n}} = \frac{2 \cdot N \cdot u'_{i,n}}{\left((1 - u_{i,n}) \cdot \sum\limits_m \dfrac{2}{1 - u_{i,m}} \right)^2} \tag{11.272}$$

$$\text{NOR:}\quad \frac{\partial u_{out}}{\partial u_{in,n}} = - \left.\frac{\partial u_{out}}{\partial u_{in,n}}\right|_{\text{OR}} \tag{11.273}$$

$$\text{AND:}\quad \frac{\partial u_{out}}{\partial u_{in,n}} = \frac{2 \cdot N \cdot u'_{i,n}}{\left((1 + u_{i,n}) \cdot \sum\limits_m \dfrac{2}{1 + u_{i,m}} \right)^2} \tag{11.274}$$

$$\text{NAND:}\quad \frac{\partial u_{out}}{\partial u_{in,n}} = - \left.\frac{\partial u_{out}}{\partial u_{in,n}}\right|_{\text{AND}} \tag{11.275}$$

$$\text{XOR:}\quad \frac{\partial u_{out}}{\partial u_{in,n}} = 0.5 \cdot u'_{i,n} \cdot \prod_{m \neq n} -u_{i,m} \tag{11.276}$$

$$\text{XNOR:}\quad \frac{\partial u_{out}}{\partial u_{in,n}} = 0.5 \cdot u'_{i,n} \cdot \prod_{m \neq n} u_{i,m} \tag{11.277}$$

A problem of these macro models are the numbers of input ports. The output voltage levels worsen with increasing number of ports. The practical limit lies around eight input ports.

With that knowledge it is now easy to create the MNA matrix. The first port is the output port of the device. So, for a 2-input port device, it is:

$$\begin{bmatrix} . & . & . & 1 \\ . & . & . & 0 \\ . & . & . & 0 \\ -1 & \partial u_{out}/\partial u_{in,1} & \partial u_{out}/\partial u_{in,2} & 0 \end{bmatrix} \cdot \begin{bmatrix} V_{out} \\ V_{in,1} \\ V_{in,2} \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_0 \\ I_1 \\ I_2 \\ 0 \end{bmatrix} \tag{11.278}$$

The above MNA matrix entries are also used during the non-linear DC and transient analysis with the 0 in the right hand side vector replaced by an equivalent voltage

$$V_{eq} = \frac{\partial u_{out}}{\partial u_{in,1}} \cdot V_{in,1} + \frac{\partial u_{out}}{\partial u_{in,2}} \cdot V_{in,2} - u_{out} \tag{11.279}$$

with u_{out} computed using equations (11.264) to (11.270).

With the given small-signal matrix representation, building the S-parameters is easy.

$$(\underline{S}) = \begin{bmatrix} -1 & 4 \cdot \partial u_{out}/\partial u_{in,1} & 4 \cdot \partial u_{out}/\partial u_{in,2} \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{11.280}$$

These matrices can easily extended to any number of input ports.

11.8 Equation defined models

Often it will happen that a user needs to implement his own model. Therefore, it is useful to supply devices that are defined by arbitrary equations.

11.8.1 Models with Explicit Equations

For example the user must enter an equation $i(V)$ describing how the port current I depends on the port voltage $V = V_1 - V_2$ and an equation $q(V)$ describing how much charge Q is held due to the voltage V. These are time domain equations. The most simple way then is a device with two nodes. Defining

$$I = i(V) \qquad \text{and} \qquad g = \frac{\partial I}{\partial V} = \lim_{h\to 0} \frac{I(V+h) - I(V)}{h} \tag{11.281}$$

as well as

$$Q = q(V) \qquad \text{and} \qquad c = \frac{\partial Q}{\partial V} = \lim_{h\to 0} \frac{Q(V+h) - Q(V)}{h} \tag{11.282}$$

the MNA matrix for a (non-linear) DC analysis writes:

$$\begin{aligned} \begin{bmatrix} +g^{(m)} & -g^{(m)} \\ -g^{(m)} & +g^{(m)} \end{bmatrix} \cdot \begin{bmatrix} V_1^{(m+1)} \\ V_2^{(m+1)} \end{bmatrix} &= \begin{bmatrix} -I^{(m)} + g^{(m)} \cdot V^{(m)} \\ +I^{(m)} - g^{(m)} \cdot V^{(m)} \end{bmatrix} \\ &= \begin{bmatrix} -I^{(m)} + g^{(m)} \cdot (V_1^{(m)} - V_2^{(m)}) \\ +I^{(m)} - g^{(m)} \cdot (V_1^{(m)} - V_2^{(m)}) \end{bmatrix} \end{aligned} \tag{11.283}$$

For a transient simulation, equation 6.91 on page 65 has to be used with Q and c.

For an AC analysis the MNA matrix writes:

$$(\underline{Y}) = (g + j\omega \cdot c) \cdot \begin{bmatrix} +1 & -1 \\ -1 & +1 \end{bmatrix} \tag{11.284}$$

And the S-parameter matrix writes:

$$S_{11} = S_{22} = \frac{1}{2 \cdot Z_0 \cdot Y + 1} \tag{11.285}$$

$$S_{12} = S_{21} = 1 - S_{11} \tag{11.286}$$

$$Y = g + j\omega \cdot c \tag{11.287}$$

The simulator needs to create the derivatives g and c by its own. This can be done numerically or symbolically. One might ask why the non-linear capacitance is modeled as charge, not as capacitance. Indeed this may be changed, but with a computer algorithm, creating the derivative is easier than to integrate.

The component described above can be expanded to one with two ports (two pairs of terminals: terminal 1 and 2 and terminal 3 and 4). That is, the currents and charges of both ports depend on both port voltages $V_{12} = V_1 - V_2$ and $V_{34} = V_3 - V_4$. Thus, the defining equations are:

$$I_1 = i_1(V_{12}, V_{34}) \qquad \text{and} \qquad g_{11} = \frac{\partial I_1}{\partial V_{12}} \qquad \text{and} \qquad g_{12} = \frac{\partial I_1}{\partial V_{34}} \tag{11.288}$$

$$I_2 = i_2(V_{12}, V_{34}) \qquad \text{and} \qquad g_{21} = \frac{\partial I_2}{\partial V_{12}} \qquad \text{and} \qquad g_{22} = \frac{\partial I_2}{\partial V_{34}} \tag{11.289}$$

as well as

$$Q_1 = q_1(V_{12}, V_{34}) \qquad \text{and} \qquad c_{11} = \frac{\partial Q_1}{\partial V_{12}} \qquad \text{and} \qquad c_{12} = \frac{\partial Q_1}{\partial V_{34}} \tag{11.290}$$

$$Q_2 = q_2(V_{12}, V_{34}) \qquad \text{and} \qquad c_{21} = \frac{\partial Q_2}{\partial V_{12}} \qquad \text{and} \qquad c_{22} = \frac{\partial Q_2}{\partial V_{34}} \tag{11.291}$$

The MNA matrix for the DC analysis writes:

$$\begin{bmatrix} g_{11}^{(m)} & -g_{11}^{(m)} & g_{12}^{(m)} & -g_{12}^{(m)} \\ -g_{11}^{(m)} & g_{11}^{(m)} & -g_{12}^{(m)} & g_{12}^{(m)} \\ g_{21}^{(m)} & -g_{21}^{(m)} & g_{22}^{(m)} & -g_{22}^{(m)} \\ -g_{21}^{(m)} & g_{21}^{(m)} & -g_{22}^{(m)} & g_{22}^{(m)} \end{bmatrix} \cdot \begin{bmatrix} V_1^{(m+1)} \\ V_2^{(m+1)} \\ V_3^{(m+1)} \\ V_4^{(m+1)} \end{bmatrix} = \begin{bmatrix} -I_1^{(m)} + g_{11}^{(m)} \cdot V_{12}^{(m)} + g_{12}^{(m)} \cdot V_{34}^{(m)} \\ I_1^{(m)} - g_{11}^{(m)} \cdot V_{12}^{(m)} - g_{12}^{(m)} \cdot V_{34}^{(m)} \\ -I_2^{(m)} + g_{21}^{(m)} \cdot V_{12}^{(m)} + g_{22}^{(m)} \cdot V_{34}^{(m)} \\ I_2^{(m)} - g_{21}^{(m)} \cdot V_{12}^{(m)} - g_{22}^{(m)} \cdot V_{34}^{(m)} \end{bmatrix} \tag{11.292}$$

For a transient simulation, equation 6.95 on page 65 has to be used with Q_1, c_{11} and c_{12}, as well as with Q_2, c_{21} and c_{22}.

For an AC analysis the MNA matrix writes:

$$(\underline{Y}) = \begin{bmatrix} g_{11} + j\omega \cdot c_{11} & -g_{11} - j\omega \cdot c_{11} & g_{12} + j\omega \cdot c_{12} & -g_{12} - j\omega \cdot c_{12} \\ -g_{11} - j\omega \cdot c_{11} & g_{11} + j\omega \cdot c_{11} & -g_{12} - j\omega \cdot c_{12} & g_{12} + j\omega \cdot c_{12} \\ g_{21} + j\omega \cdot c_{21} & -g_{21} - j\omega \cdot c_{21} & g_{22} + j\omega \cdot c_{22} & -g_{22} - j\omega \cdot c_{22} \\ -g_{21} - j\omega \cdot c_{21} & g_{21} + j\omega \cdot c_{21} & -g_{22} - j\omega \cdot c_{22} & g_{22} + j\omega \cdot c_{22} \end{bmatrix} \tag{11.293}$$

As can bee seen, this scheme can be expanded to any number of ports. The matrices soon become quite complex, but fortunately, modern computers are able to cope with this complexity. S-parameters must be obtained numerical by setting equation 11.293 into equation 19.5.

11.8.2 Models with Implicit Equations

The above-mentioned explicit models are not useable for all components. If the Y-parameters do not exist or if the equations cannot be analytical transformed into the explicit form, then an implicit representation must be taken. That is, for a one-port (two-terminal) component the following formulas are defined by the user:

$$0 = f(V, I) \qquad \text{and} \qquad g_V = \frac{\partial f(V, I)}{\partial V} = \lim_{h \to 0} \frac{f(V + h, I)) - f(V, I)}{h} \tag{11.294}$$

$$\text{and} \qquad g_I = \frac{\partial f(V, I)}{\partial I} = \lim_{h \to 0} \frac{f(V, I + h)) - f(V, I)}{h} \tag{11.295}$$

The MNA matrix for the AC analysis writes as follows:

$$\begin{bmatrix} . & . & 1 \\ . & . & -1 \\ g_V & -g_V & g_I \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{out} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \tag{11.296}$$

As usual, for the DC analysis the last zero on the right hand side has to be replaced by the iteration formula:

$$g_V \cdot (V_1 - V_2) + g_I \cdot I_{out} - f(V_1 - V_2, I_{out}) \tag{11.297}$$

The S-parameters are:

$$S_{11} = S_{22} = \frac{g_I}{g_I - 2 \cdot Z_0 \cdot g_V} \tag{11.298}$$

$$S_{12} = S_{21} = 1 - S_{11} \tag{11.299}$$

Consequently, for a two-port device two equation are necessary: One for first port and one for second port:

$$0 = f_1(V_{12}, V_{34}, I_1, I_2) \tag{11.300}$$

$$0 = f_2(V_{12}, V_{34}, I_1, I_2) \tag{11.301}$$

Building the MNA matrix is again straight forward:

$$\begin{bmatrix} . & . & . & . & 1 & 0 \\ . & . & . & . & -1 & 0 \\ . & . & . & . & 0 & 1 \\ . & . & . & . & 0 & -1 \\ g_{f1,V12} & -g_{f1,V12} & g_{f1,V34} & -g_{f1,V34} & g_{f1,I1} & g_{f1,I2} \\ g_{f2,V12} & -g_{f2,V12} & g_{f2,V34} & -g_{f2,V34} & g_{f2,I1} & g_{f2,I2} \end{bmatrix} \cdot \begin{bmatrix} V_1^{(m+1)} \\ V_2^{(m+1)} \\ V_3^{(m+1)} \\ V_4^{(m+1)} \\ I_{out1}^{(m+1)} \\ I_{out2}^{(m+1)} \end{bmatrix} \tag{11.302}$$
$$= \begin{bmatrix} g_{f1,V12} \cdot V_{12} + g_{f1,V34} \cdot V_{34} + g_{f1,I1} \cdot I_{out1} + g_{f1,I2} \cdot I_{out2} - f_1(V_{12}, V_{34}, I_{out1}, I_{out2}) \\ g_{f2,V12} \cdot V_{12} + g_{f2,V34} \cdot V_{34} + g_{f2,I1} \cdot I_{out1} + g_{f2,I2} \cdot I_{out2} - f_2(V_{12}, V_{34}, I_{out1}, I_{out2}) \end{bmatrix}$$

Once more, this concept can easily expanded to any number of ports. It is also possible mix implicit and explicit definitions, i.e. some ports of the device may be defined by explicit equations whereas the others are defined by implicit equations.

The calculation of the S-parameters is not that trival. The Y-parameters as well as the Z-parameters might be infinite. A small trick can avoid this problem, as will be shown in the following 2-port example. First, the small-signal Y-parameters should be derived by using the law about implicit functions:

$$(\underline{J}) = \begin{pmatrix} y_{11} & y_{12} \\ y_{21} & y_{22} \end{pmatrix} = -\underbrace{\begin{pmatrix} \dfrac{\partial f_1}{\partial I_1} & \dfrac{\partial f_1}{\partial I_2} \\ \dfrac{\partial f_2}{\partial I_1} & \dfrac{\partial f_2}{\partial I_2} \end{pmatrix}^{-1}}_{J_i} \cdot \underbrace{\begin{pmatrix} \dfrac{\partial f_1}{\partial V_1} & \dfrac{\partial f_1}{\partial V_2} \\ \dfrac{\partial f_2}{\partial V_1} & \dfrac{\partial f_2}{\partial V_2} \end{pmatrix}}_{J_v} \tag{11.303}$$

The equation reveals immediately the difficulty: The inverse of the current Jacobi matrix J_i may not exist. But this problem can be outsourced to one single scalar number by using Cramer's rule for matrix inversion:

$$J_i^{-1} = \frac{1}{\det J_i} \cdot A_{Ji} \tag{11.304}$$

The matrix A_{Ji} is built of the sub-determinantes of J_i in the way that $a_{(n,m)}$ is the determinante of J_i without row m and without column n but multiplied with $(-1)^{n+m}$. It therefore always exists, whereas dividing by the determinante of J_i may become infinity. Now parameters can be defined as follows:

$$\det J_i \cdot (\underline{J}) = \begin{pmatrix} y'_{11} & y'_{12} \\ y'_{21} & y'_{22} \end{pmatrix} = -A_{Ji} \cdot J_v \tag{11.305}$$

Before converting to S-parameters the matrix must be expanded to a 4-port matrix, because the 2-ports are not referenced to ground:

$$\det J_i \cdot (\underline{J}') = \begin{pmatrix} y'_{11} & -y'_{11} & y'_{12} & -y'_{12} \\ -y'_{11} & y'_{11} & -y'_{12} & y'_{12} \\ y'_{21} & -y'_{21} & y'_{22} & -y'_{22} \\ -y'_{21} & y'_{21} & -y'_{22} & y'_{22} \end{pmatrix} = -A'_{Ji} \cdot J'_v \tag{11.306}$$

Finally, equation 19.5 converts the parameters to S-parameters:

$$(\underline{S}) = ((E) - Z_0 \cdot (\underline{Y})) \cdot ((E) + Z_0 \cdot (\underline{Y}))^{-1} \tag{11.307}$$
$$= \left((E) + Z_0 \cdot \frac{1}{\det J_i} \cdot A'_{Ji} \cdot J'_v\right) \cdot \left((E) - Z_0 \cdot \frac{1}{\det J_i} \cdot A'_{Ji} \cdot J'_v\right)^{-1} \tag{11.308}$$
$$= (\det J_i \cdot (E) + Z_0 \cdot A'_{Ji} \cdot J'_v) \cdot (\det J_i \cdot (E) - Z_0 \cdot A'_{Ji} \cdot J'_v)^{-1} \tag{11.309}$$

The calculations proofs that the critical factor $1/\det J_i$ disappears and a solution exists if and only if the S-parameters of this device exist.

11.9 Resonance tunnel diode

A resonance tunnel diode (RTD) is a two-terminal semiconductor device. Its schematic symbol and its IV-characteristic are shown in fig. 11.16. An RTD consists of a very thin double-barrier structure that only electrons with a specific energy can pass. This leads to the typical peak current I_P at the peak voltage V_P. Above the valley voltage V_V the thermal-ionic current I_{TH} dominates and thus, the current increases rapidly. The negative differential resistance (NDR) between V_P and V_V makes it possible to build oscillators and logic gates that work at extremely high frequencies.

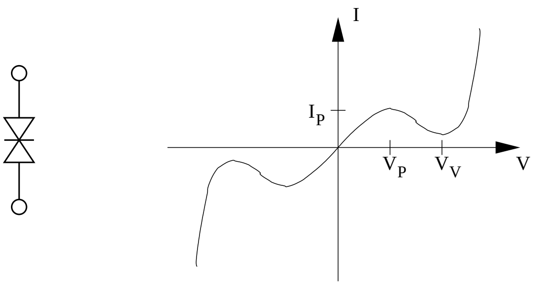


Figure 11.16: schematic symbol (left) and IV-characteristic (right) of a resonance tunnel diode

The IV-characteristic can be modeled by the following equations [32]. The model parameters are listed in table at the end of this section.

$$I_D(V) = I_T(V) - I_T(-V) + I_{TH} \tag{11.310}$$

$$\text{with} \tag{11.311}$$

$$X = W_r - \frac{q \cdot V}{\delta_V} \tag{11.312}$$

$$I_T(V) = J_P \cdot A \cdot T_{SC} \cdot \frac{2 \cdot \delta_T \cdot k_B \cdot T}{\pi \cdot \eta} \cdot \ln\left(1 + \exp\left(\frac{\eta - X}{\delta_T \cdot k_B \cdot T}\right)\right) \cdot \left(\frac{\pi}{2} + \arctan\frac{X}{\Delta W}\right) \tag{11.313}$$

$$I_{TH} = J_V \cdot A \cdot \frac{\sinh\dfrac{q \cdot V}{n_V \cdot k_B \cdot T}}{\sinh\dfrac{q \cdot V_V}{n_V \cdot k_B \cdot T}} \tag{11.314}$$

Where $q = 1.6 \cdot 10^{-19}$As is charge of electron, $k_B = 1.38 \cdot 10^{-23}$J/K is Boltzmann's constant and $\pi = 3.14\ldots$ is a mathematical constant. The non-linear (parallel) capacitance consists of the depletion capacitance C_{dep} and the quantum-well capacitance C_{qw}. They can be modeled as follows [32]:

$$C(V) = C_{dep} + C_{qw} \tag{11.315}$$

$$\text{with} \tag{11.316}$$

$$C_{dep} = \frac{A \cdot C_0}{\left(1 + \dfrac{V}{\phi_C}\right)^m} \tag{11.317}$$

$$C_{qw} = -\tau_{qw} \cdot \frac{\partial I_D(V)}{\partial V} \tag{11.318}$$

Symbol	Description	Default	Unit
J_P	peak current density	$4 \cdot 10^8$	A$/m^2$
A	area	10^{-11}	m^2
W_r	resonance energy	$2.7 \cdot 10^{-20}$	Ws
η	level of Fermi energy W_F to W_L	$1.1 \cdot 10^{-20}$	Ws
ΔW	width of resonance	$4.5 \cdot 10^{-21}$	Ws
T_{SC}	maximum of transmission	0.95	1
δ_T	fitting factor for electron density	0.9	1
δ_V	fitting factor for voltage drop	2.0	1
J_V	valley current density	$6.2 \cdot 10^7$	A$/m^2$
V_V	valley voltage	0.8	V
n_V	fitting factor for diode current	16.5	V
T	device temperature	300	K
C_0	depletion capacitance at $V = 0$V	$8 \cdot 10^{-3}$	F/m^2
ϕ_C	voltage scaling factor	0.5	1
m	voltage exponent	0.5	1
τ_{qw}	life-time of electrons in quantum wel	$6 \cdot 10^{-13}$	sec

11.10 Photodiode

A photodiode (usually a PIN diode) converts an optical input power into an electrical current. Figure 11.17 shows the eqivalent circuit. The optical power is modeled as a voltage at the optical node. Note that a photodiode is biased in backward direction, i.e. with a positive potential at the cathode. The responsivity R_{oe} determines the conversion from optical power P_{opt} to electrical photocurrent I_{photo}:

$$I_{photo} = R_{oe} \cdot P_{opt} \tag{11.319}$$

As usual, the transit time τ of the intrinsic diode is approximated with the diffusions capacitance, whereas the photocurrent exhibits a rectangle impulse response of width τ. The frequency response is

$$H(j\omega) = \frac{\exp(j\omega \cdot \tau) - 1}{j\omega \cdot \tau} \tag{11.320}$$

In addition to the usual noise sources (thermal noise of resistor R_S and shot noise of diode current) the photocurrent creates shot noise, too:

$$S_i = 2 \cdot e \cdot I_{photo} \tag{11.321}$$

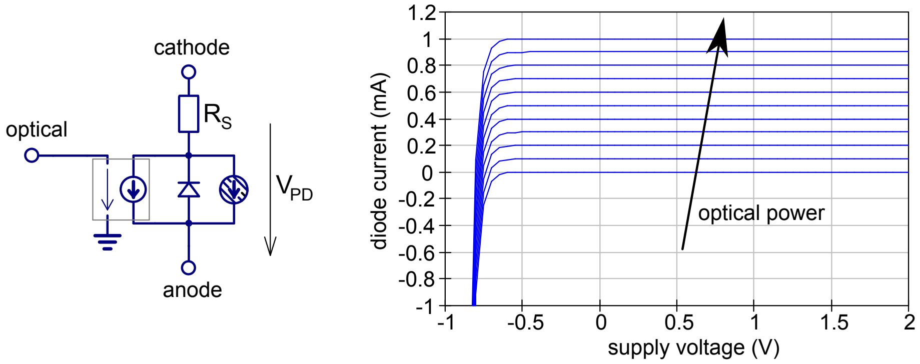


Figure 11.17: electrical equivalent circuit (left) and IV characteristic (right) of a photodiode

11.11 TOM3 Model

TriQuint's Own Model III (TOM3) was developed for simulating GaAs MESFETs. It was designed to fit the drain-source current I_{DS} as well as its first three derivatives. This is important in order to model non-linearities. The equations writes as follows [33].

$$\begin{aligned}
I_{DS} &= I_0 \cdot (1 + \lambda \cdot V_{DS}) && (11.322)\\
I_0 &= \beta \cdot V_g^Q \cdot \frac{\alpha \cdot V_{DS}}{\left(1 + (\alpha \cdot V_{DS})^\kappa\right)^{1/\kappa}} && (11.323)\\
V_g &= Q \cdot V_{st} \cdot \ln\left(\exp(u) + 1\right) && (11.324)\\
u &= \frac{V_{GS} - V_{T0} + \gamma \cdot V_{DS}}{Q \cdot V_{st}} && (11.325)\\
V_{st} &= V_{st0} \cdot (1 + M_{st0} \cdot V_{DS}) && (11.326)
\end{aligned}$$

with V_{GS} being the intrinsic gate-source voltage and V_{DS} being the intrinsic drain-source voltage. The term "ln(exp(u) + 1)" creates a soft channel pinch-off and "$(1 + \lambda \cdot V_{DS})$" creates a finite output resistance.

The transconductance G_M and the output conductance G_{DS} are as follows.

$$\begin{aligned}
G_M &= \frac{\partial I_{DS}}{\partial V_{GS}} = \frac{I_{DS} \cdot Q}{V_g \cdot (1 + \exp(-u))} && (11.327)\\
G_{DS} &= \frac{\partial I_{DS}}{\partial V_{DS}} && (11.328)\\
&= \lambda \cdot I_0 + G_M \cdot (\gamma - u \cdot Q \cdot V_{st0} \cdot M_{st0}) && (11.329)\\
&+ \left(\frac{Q \cdot I_0 \cdot M_{st0}}{1 + M_{st0} \cdot V_{DS}} + \frac{\alpha \cdot \beta \cdot V_g^Q}{\left(1 + (\alpha \cdot V_{DS})^\kappa\right)^{1+1/\kappa}}\right) \cdot (1 + \lambda \cdot V_{DS}) && (11.330)
\end{aligned}$$

The gate-source and the gate-drain diodes exhibit the usual Shockley equation. In parallel to each of them there are additional leakage currents:

$$\begin{aligned}
I_L &= I_{LK} \cdot \left(1 - \exp\frac{-V}{\phi_{LK}}\right) && (11.331)\\
G_L &= \frac{\partial I_L}{\partial V} = \frac{I_L}{\phi_{LK}} \cdot \exp\frac{-V}{\phi_{LK}} && (11.332)
\end{aligned}$$

The non-linear gate-source and gate-drain charges and capacitances are modeled with the following equations.

$$Q_{GG} = 2 \cdot Q_{GS} = 2 \cdot Q_{GD} = Q_{gl} \cdot F_t + Q_{gh} \cdot (1 - F_t) + Q_{GG0} \cdot (V_{GS} + V_{GD}) \tag{11.333}$$

$$F_t = \exp(-Q_{GGB} \cdot I_{DS} \cdot V_{DS}) \tag{11.334}$$

$$\frac{\partial F_t}{\partial V_{GS}} = -Q_{GGB} \cdot (I_{DS} + (G_M + G_{DS}) \cdot V_{DS}) \cdot F_t \tag{11.335}$$

$$\frac{\partial F_t}{\partial V_{GD}} = Q_{GGB} \cdot (I_{DS} + G_{DS} \cdot V_{DS}) \cdot F_t \tag{11.336}$$

$$F_l = Q_{GQL} \cdot \exp\left(Q_{GAG} \cdot (V_{GS} + V_{GD})\right) \tag{11.337}$$

$$Q_{gl} = F_l \cdot \cosh(Q_{GAD} \cdot V_{DS}) + Q_{GCL} \cdot (V_{GS} + V_{GD}) \tag{11.338}$$

$$Q_{gh} = Q_{GQH} \cdot \ln\left(1 + \frac{I_{DS}}{Q_{GI0}}\right) + Q_{GSH} \cdot V_{GS} + Q_{GDH} \cdot V_{GD} \tag{11.339}$$

$$C_{gsh} = (G_M + G_{DS}) \cdot \frac{Q_{GQH}}{I_{DS} + Q_{GI0}} + Q_{GSH} \tag{11.340}$$

$$C_{gdh} = -G_{DS} \cdot \frac{Q_{GQH}}{I_{DS} + Q_{GI0}} + Q_{GDH} \tag{11.341}$$

$$C_{gsl} = F_l \cdot (Q_{GAG} \cdot \cosh(Q_{GAD} \cdot V_{DS}) + Q_{GAD} \cdot \sinh(Q_{GAD} \cdot V_{DS})) + Q_{GCL} \tag{11.342}$$

$$C_{gdl} = F_l \cdot (Q_{GAG} \cdot \cosh(Q_{GAD} \cdot V_{DS}) - Q_{GAD} \cdot \sinh(Q_{GAD} \cdot V_{DS})) + Q_{GCL} \tag{11.343}$$

$$C_{GS} = \left(C_{gsl} \cdot F_t + C_{gsh} \cdot (1 - F_t) + (Q_{gl} - Q_{gh}) \cdot \frac{\partial F_t}{\partial V_{GS}} + Q_{GG0}\right) \tag{11.344}$$

$$C_{GD} = \left(C_{gdl} \cdot F_t + C_{gdh} \cdot (1 - F_t) + (Q_{gl} - Q_{gh}) \cdot \frac{\partial F_t}{\partial V_{GD}} + Q_{GG0}\right) \tag{11.345}$$

If the drain-source voltage is negative, V_{GS} and V_{GD} have to be exchanged.

Most parameters scale proportionally with the transistor size, i.e. they are multiplied with the channel width W and with the number of gate finger N_g. This holds for the following parameters:

$$\beta, \quad I_S, \quad C_{DS}, \quad Q_{gql}, \quad Q_{gqh}, \quad Q_{gi0}, \quad Q_{gcl}, \quad Q_{gsh}, \quad Q_{gdh}, \quad Q_{gg0}, \quad I_{LK} \tag{11.346}$$

The following parameters are scaled inversly, i.e. they are divided by the channel width W and by the number of gate finger N_g.

$$Q_{ggb}, \quad R_d, \quad R_s, \quad R_g \tag{11.347}$$

The scaling of the gate metal resistor is somewhat different:

$$R_{gmet,scaled} = R_{gmet} \cdot \frac{W}{N_g} \tag{11.348}$$

There are three types of temperature dependencies. The linear scaling is defined as

$$P_{scaled} = P_{nom} + Scale \cdot (T - T_{nom}) \tag{11.349}$$

and applies to

$$V_{T0}, \quad \gamma, \quad V_{st0}, \quad M_{st0} \tag{11.350}$$

The linear, relative scaling is defined as

$$P_{scaled} = P_{nom} \cdot (1 + Scale \cdot (T - T_{nom})) \tag{11.351}$$

and applies to

$$R_d, \quad R_s \tag{11.352}$$

The exponential scaling is defined as

$$P_{scaled} = P_{nom} \cdot 1.01^{Scale\,\cdot\,(T - T_{nom})} \tag{11.353}$$

and applies to

$$\alpha, \quad \beta \tag{11.354}$$

The noise model is typical for a JFET. The parasitic resistances R_d, R_s, R_g and R_{gmet} creates thermal noise. The channel noise is modeled with a user-defined parameter P:

$$\frac{\overline{i_{ds}^2}}{\Delta f} = 4 \cdot k_B \cdot T \cdot G_M \cdot P + K_F \frac{I_{DS}^{A_F}}{f^{F_{FE}}} \tag{11.355}$$

11.12 Statz Model

A popular GaAs MESFET model is the Statz Raytheon model [34]. The equations for the drain-source current writes as follows.

for $0 < V_{ds} < \frac{\alpha}{3}$

$$\begin{aligned}
I_{ds} &= \frac{\beta \cdot (V_{gs} - V_{t0})^2}{1 + b \cdot (V_{gs} - V_{t0})} \cdot \left(1 - \left(1 - \frac{\alpha \cdot V_{ds}}{3}\right)^3\right) \cdot (1 + \lambda \cdot V_{ds}) && (11.356)\\
\frac{\partial I_{ds}}{\partial V_{ds}} &= \frac{\beta \cdot (V_{gs} - V_{t0})^2}{1 + b \cdot (V_{gs} - V_{t0})} && (11.357)\\
&\quad \cdot \left(\alpha \cdot \left(1 - \frac{\alpha \cdot V_{ds}}{3}\right)^2 \cdot (1 + \lambda \cdot V_{ds}) + \lambda \cdot \left(1 - \left(1 - \frac{\alpha \cdot V_{ds}}{3}\right)^3\right)\right) && (11.358)\\
\frac{\partial I_{ds}}{\partial V_{gs}} &= \beta \cdot (V_{gs} - V_{t0}) \cdot \frac{2 + b \cdot (V_{gs} - V_{t0})}{(1 + b \cdot (V_{gs} - V_{t0}))^2} \cdot (1 + \lambda \cdot V_{ds}) && (11.359)\\
&\quad \cdot \left(1 - \left(1 - \frac{\alpha \cdot V_{ds}}{3}\right)^3\right) && (11.360)
\end{aligned}$$

and for $V_{ds} \geq \frac{\alpha}{3}$

$$\begin{aligned}
I_{ds} &= \frac{\beta \cdot (V_{gs} - V_{t0})^2}{1 + b \cdot (V_{gs} - V_{t0})} \cdot (1 + \lambda \cdot V_{ds}) && (11.361)\\
\frac{\partial I_{ds}}{\partial V_{ds}} &= \frac{\beta \cdot (V_{gs} - V_{t0})^2}{1 + b \cdot (V_{gs} - V_{t0})} \cdot \lambda && (11.362)\\
\frac{\partial I_{ds}}{\partial V_{gs}} &= \beta \cdot (V_{gs} - V_{t0}) \cdot \frac{2 + b \cdot (V_{gs} - V_{t0})}{(1 + b \cdot (V_{gs} - V_{t0}))^2} \cdot (1 + \lambda \cdot V_{ds}) && (11.363)
\end{aligned}$$

with V_{gs} being the intrinsic gate-source voltage and V_{ds} being the intrinsic drain-source voltage.

The charge and capacitances are:

for $V_{new} > V_{max}$

$$Q_{gs} = C_{gs0} \cdot \left(2 \cdot V_{bi} \cdot \left(1 - \sqrt{1 - \frac{V_{max}}{V_{bi}}} \right) + \frac{V_{new} - V_{max}}{\sqrt{1 - \frac{V_{max}}{V_{bi}}}} \right) \tag{11.364}$$

$$\frac{\partial Q_{gs}}{\partial V_{gs}} = \frac{C_{gs0}}{\sqrt{1 - \frac{V_{max}}{V_{bi}}}} \cdot \frac{\partial V_{new}}{\partial V_{gs}} \tag{11.365}$$

$$\frac{\partial Q_{gs}}{\partial V_{gd}} = \frac{C_{gs0}}{\sqrt{1 - \frac{V_{max}}{V_{bi}}}} \cdot \frac{\partial V_{new}}{\partial V_{gd}} \tag{11.366}$$

and for $V_{new} \leq V_{max}$

$$Q_{gs} = C_{gs0} \cdot 2 \cdot V_{bi} \cdot \left(1 - \sqrt{1 - \frac{V_{new}}{V_{bi}}} \right) \tag{11.367}$$

$$\frac{\partial Q_{gs}}{\partial V_{gs}} = \frac{C_{gs0}}{\sqrt{1 - \frac{V_{new}}{V_{bi}}}} \cdot \frac{\partial V_{new}}{\partial V_{gs}} \tag{11.368}$$

$$\frac{\partial Q_{gs}}{\partial V_{gd}} = \frac{C_{gs0}}{\sqrt{1 - \frac{V_{new}}{V_{bi}}}} \cdot \frac{\partial V_{new}}{\partial V_{gd}} \tag{11.369}$$

with

$$V_{max} = F_c \cdot V_{bi} \tag{11.370}$$

$$V_{new} = \frac{1}{2} \cdot \left(V_{eff1} + V_{t0} + \sqrt{(V_{eff1} - V_{t0})^2 + \Delta_2^2} \right) \tag{11.371}$$

$$V_{eff1} = \frac{1}{2} \cdot \left(V_{gs} + V_{gd} + \sqrt{(V_{gs} - V_{gd})^2 + \Delta_1^2} \right) \tag{11.372}$$

$$V_{eff2} = \frac{1}{2} \cdot \left(V_{gs} + V_{gd} - \sqrt{(V_{gs} - V_{gd})^2 + \Delta_1^2} \right) \tag{11.373}$$

$$\frac{\partial V_{new}}{\partial V_{gs}} = \left(\frac{1}{2} + \frac{V_{eff1} - V_{t0}}{2 \cdot \sqrt{(V_{eff1} - V_{t0})^2 + \Delta_2^2}} \right) \cdot \frac{\partial V_{eff1}}{\partial V_{gs}} \tag{11.374}$$

$$= \left(\frac{1}{2} + \frac{V_{eff1} - V_{t0}}{2 \cdot \sqrt{(V_{eff1} - V_{t0})^2 + \Delta_2^2}} \right) \cdot \left(\frac{1}{2} + \frac{V_{gs} - V_{gd}}{2 \cdot \sqrt{(V_{gs} - V_{gd})^2 + \Delta_1^2}} \right) \tag{11.375}$$

$$\frac{\partial V_{new}}{\partial V_{gd}} = \left(\frac{1}{2} + \frac{V_{eff1} - V_{t0}}{2 \cdot \sqrt{(V_{eff1} - V_{t0})^2 + \Delta_2^2}} \right) \cdot \left(\frac{1}{2} - \frac{V_{gs} - V_{gd}}{2 \cdot \sqrt{(V_{gs} - V_{gd})^2 + \Delta_1^2}} \right) \tag{11.376}$$

Furthermore it is:

$$Q_{gd} = \frac{C_{gd0}}{2} \cdot \left(V_{gs} + V_{gd} - \sqrt{(V_{gs} - V_{gd})^2 + \Delta_1^2} \right) \tag{11.377}$$

$$\frac{\partial Q_{gd}}{\partial V_{gs}} = C_{gd0} \cdot \left(\frac{1}{2} - \frac{V_{gs} - V_{gd}}{2 \cdot \sqrt{(V_{gs} - V_{gd})^2 + \Delta_1^2}} \right) \tag{11.378}$$

$$\frac{\partial Q_{gd}}{\partial V_{gd}} = C_{gd0} \cdot \left(\frac{1}{2} + \frac{V_{gs} - V_{gd}}{2 \cdot \sqrt{(V_{gs} - V_{gd})^2 + \Delta_1^2}} \right) \tag{11.379}$$

11.13 HICUM

HICUM (High-Current Model) is a modern and very popular circuit model for bipolar junction transistors. It is designed for standard BJTs as well as for HBTs (Heterojunction Bipolar Transistors) and especially takes into account the effects at high frequencies and high currents. The

model was developed by the HICUM Group at CEDIC at the University of Technology Dresden and the University of California at San Diego. The model owner is Michael Schröter. All information about HICUM can be found on the internet:
`https://www.iee.et.tu-dresden.de/iee/eb/hic_new/hic_intro.html`

Chapter 12

Microstrip components

12.1 Single microstrip line

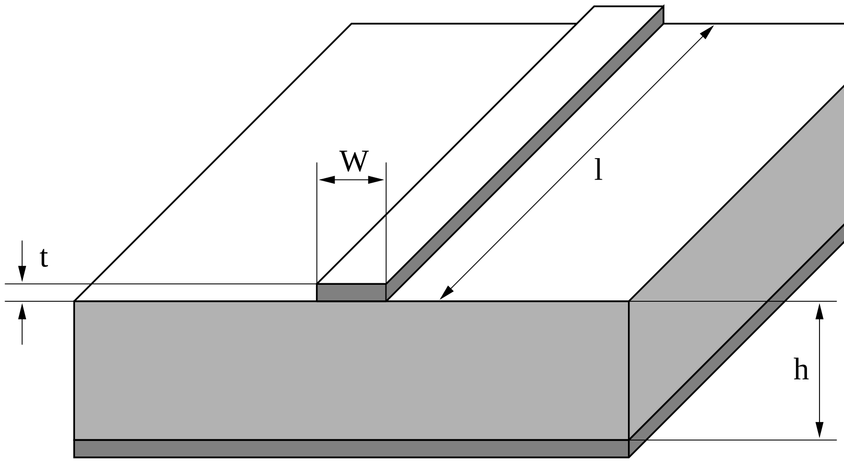


Figure 12.1: single microstrip line

The electrical parameters of microstrip lines which are required for circuit design are impedance, attenuation, wavelength and propagation constant. These parameters are interrelated for all microstrips assuming that the propagation mode is a transverse electromagnetic mode, or it can be approximated by a transverse electromagnetic mode. The Y and S parameters can be found in section 10.23.

12.1.1 Quasi-static characteristic impedance

Wheeler

Harold A. Wheeler [35] formulated his synthesis and analysis equations based upon a conformal mapping's approximation of the dielectric boundary with parallel conductor strips separated by a dielectric sheet.

For wide strips ($W/h > 3.3$) he obtains the approximation

$$Z_L\left(W, h, \varepsilon_r\right) = \frac{Z_{F0}}{2\sqrt{\varepsilon_r}} \cdot \frac{1}{\dfrac{W}{2h} + \dfrac{1}{\pi}\ln 4 + \dfrac{\varepsilon_r + 1}{2\pi\varepsilon_r}\ln\left(\dfrac{\pi e}{2}\left(\dfrac{W}{2h} + 0.94\right)\right) + \dfrac{\varepsilon_r - 1}{2\pi\varepsilon_r^2} \cdot \ln\dfrac{e\pi^2}{16}} \tag{12.1}$$

For narrow strips ($W/h \leq 3.3$) he obtains the approximation

$$Z_L\left(W, h, \varepsilon_r\right) = \frac{Z_{F0}}{\pi\sqrt{2\left(\varepsilon_r + 1\right)}} \cdot \left(\ln\left(\frac{4h}{W} + \sqrt{\left(\frac{4h}{W}\right)^2 + 2}\right) - \frac{1}{2} \cdot \frac{\varepsilon_r - 1}{\varepsilon_r + 1}\left(\ln\frac{\pi}{2} + \frac{1}{\varepsilon_r}\ln\frac{4}{\pi}\right)\right) \tag{12.2}$$

The formulae are applicable to alumina-type substrates ($8 \leq \varepsilon_r \leq 12$) and have an estimated relative error less than 1 per cent.

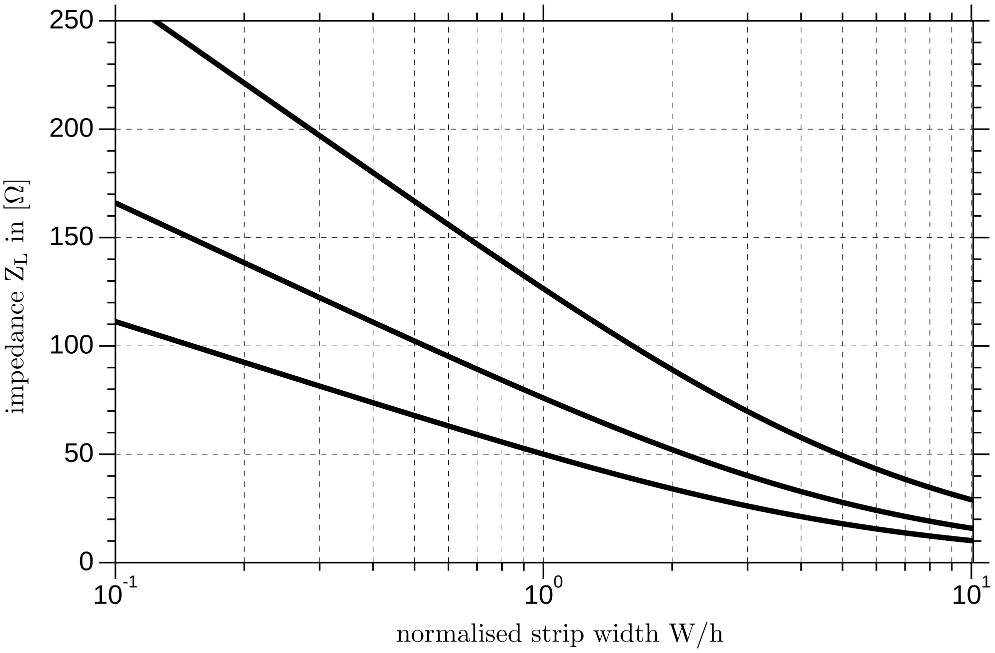


Figure 12.2: characteristic impedance as approximated by Hammerstad for $\varepsilon_r = 1.0$ (air), 3.78 (quartz) and 9.5 (alumina)

Schneider

The following formulas obtained by rational function approximation give accuracy of ±0.25% for $0 \leq W/h \leq 10$ which is the range of importance for most engineering applications. M.V. Schneider [36] found these approximations for the complete elliptic integrals of the first kind as accurate as ±1% for $W/h > 10$.

$$Z_L = \frac{Z_{F0}}{\sqrt{\varepsilon_{r_{eff}}}} \cdot \begin{cases} \dfrac{1}{2\pi} \cdot \ln\left(\dfrac{8 \cdot h}{W} + \dfrac{W}{4 \cdot h}\right) & \text{for } \dfrac{W}{h} \leq 1 \\ \dfrac{1}{\dfrac{W}{h} + 2.42 - 0.44 \cdot \dfrac{h}{W} + \left(1 - \dfrac{h}{W}\right)^6} & \text{for } \dfrac{W}{h} > 1 \end{cases} \tag{12.3}$$

Hammerstad and Jensen

The equations for the single microstrip line presented by E. Hammerstad and Ø. Jensen [37] are based upon an equation for the impedance of microstrip in an homogeneous medium and an equation for the microstrip effective dielectric constant. The obtained accuracy gives errors at least less than those caused by physical tolerances and is better than 0.01% for $W/h \leq 1$ and 0.03% for $W/h \leq 1000$.

$$Z_{L1}\left(W,h\right) = \frac{Z_{F0}}{2\pi} \cdot \ln\left(f_u \frac{h}{W} + \sqrt{1 + \left(\frac{2h}{W}\right)^2}\right) \tag{12.4}$$

$$Z_L\left(W,h,\varepsilon_r\right) = \frac{Z_{L1}\left(W,h\right)}{\sqrt{\varepsilon_r}} = \frac{Z_{F0}}{2\pi \cdot \sqrt{\varepsilon_r}} \cdot \ln\left(f_u \frac{h}{W} + \sqrt{1 + \left(\frac{2h}{W}\right)^2}\right) \tag{12.5}$$

with

$$f_u = 6 + (2\pi - 6) \cdot \exp\left(-\left(30.666 \cdot \frac{h}{W}\right)^{0.7528}\right) \tag{12.6}$$

The comparison of the expression given for the quasi-static impedance as shown in fig. 12.3 has been done with respect to E. Hammerstad and Ø. Jensen. It reveals the advantage of closed-form expressions. The impedance step for Wheelers formulae at $W/h = 3.3$ is approximately 0.1Ω.

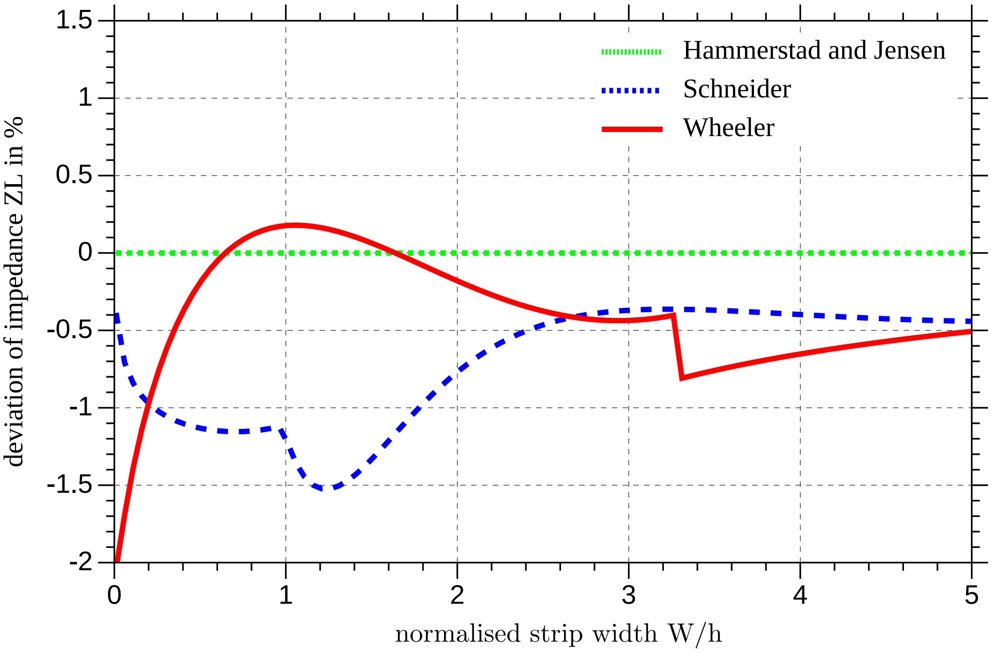


Figure 12.3: characteristic impedance in comparison for $\varepsilon_r = 9.8$

12.1.2 Quasi-static effective dielectric constant

Wheeler

Harold A. Wheeler [38] gives the following approximation for narrow strips ($W/h < 3$) based upon the characteristic impedance Z_L. The estimated relative error is less than 1%.

$$\varepsilon_{r_{eff}} = \frac{\varepsilon_r + 1}{2} + \frac{Z_{F0}}{2\pi Z_L} \cdot \frac{\varepsilon_r - 1}{2} \cdot \left(\ln\frac{\pi}{2} + \frac{1}{\varepsilon_r}\ln\frac{4}{\pi}\right) \tag{12.7}$$

For narrow strips ($W/h \leq 1.3$):

$$\varepsilon_{r_{eff}} = \frac{1+\varepsilon_r}{2} \cdot \left(\frac{A}{A-B}\right)^2 \tag{12.8}$$

with

$$A = \ln\left(8\frac{h}{W}\right) + \frac{1}{32} \cdot \left(\frac{W}{h}\right)^2 \tag{12.9}$$

$$B = \frac{1}{2} \cdot \frac{\varepsilon_r - 1}{\varepsilon_r + 1} \cdot \left(\ln\frac{\pi}{2} + \frac{1}{\varepsilon_r}\ln\frac{4}{\pi}\right) \tag{12.10}$$

For wide strips ($W/h > 1.3$):

$$\varepsilon_{r_{eff}} = \varepsilon_r \cdot \left(\frac{E-D}{E}\right)^2 \tag{12.11}$$

with

$$D = \frac{\varepsilon_r - 1}{2\pi\varepsilon_r} \cdot \left(\ln\left(\frac{\pi e}{2}\left(\frac{W}{2h} + 0.94\right)\right) - \frac{1}{\varepsilon_r}\ln\frac{e\pi^2}{16}\right) \tag{12.12}$$

$$E = \frac{1}{2} \cdot \frac{W}{h} + \frac{1}{\pi} \cdot \ln\left(\pi e\frac{W}{h} + 16.0547\right) \tag{12.13}$$

Schneider

The approximate function found by M.V. Schneider [36] is meant to have an accuracy of ±2% for $\varepsilon_{r_{eff}}$ and an accuracy of ±1% for $\sqrt{\varepsilon_{r_{eff}}}$.

$$\varepsilon_{r_{eff}} = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2} \cdot \frac{1}{\sqrt{1 + 10\dfrac{h}{W}}} \tag{12.14}$$

Hammerstad and Jensen

The accuracy of the E. Hammerstad and Ø. Jensen [37] model is better than 0.2% at least for $\varepsilon_r < 128$ and $0.01 \leq W/h \leq 100$.

$$\varepsilon_{r_{eff}}(W, h, \varepsilon_r) = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2} \cdot \left(1 + 10\frac{h}{W}\right)^{-ab} \tag{12.15}$$

with

$$a(u) = 1 + \frac{1}{49} \cdot \ln\left(\frac{u^4 + (u/52)^2}{u^4 + 0.432}\right) + \frac{1}{18.7} \cdot \ln\left(1 + \left(\frac{u}{18.1}\right)^3\right) \tag{12.16}$$

$$b(\varepsilon_r) = 0.564 \cdot \left(\frac{\varepsilon_r - 0.9}{\varepsilon_r + 3}\right)^{0.053} \tag{12.17}$$

$$u = \frac{W}{h} \tag{12.18}$$

12.1.3 Strip thickness correction

The formulas given for the quasi-static characteristic impedance and effective dielectric constant in the previous sections are based upon an infinite thin microstrip line thickness $t = 0$. A finite thickness t can be compensated by a reduction of width. That means a strip with the width W and the finite thickness t appears to be a wider strip.

Wheeler

Harold A. Wheeler [38] proposes the following equation to account for the strip thickness effect based on free space without dielectric.

$$\Delta W_1 = \frac{t}{\pi} \ln \frac{4e}{\sqrt{\left(\frac{t}{h}\right)^2 + \left(\frac{1/\pi}{W/t + 1.10}\right)}} \tag{12.19}$$

For the mixed media case with dielectric he obtains the approximation

$$\Delta W_r = \frac{1}{2} \Delta W_1 \left(1 + \frac{1}{\varepsilon_r}\right) \tag{12.20}$$

Schneider

M.V. Schneider [36] derived the following approximate expressions.

$$\Delta W = \begin{cases} \frac{t}{\pi} \cdot \left(1 + \ln \frac{4 \cdot \pi \cdot W}{t}\right) & \text{for } \frac{W}{h} \leq \frac{1}{2\pi} \\ \frac{t}{\pi} \cdot \left(1 + \ln \frac{2 \cdot h}{t}\right) & \text{for } \frac{W}{h} > \frac{1}{2\pi} \end{cases} \tag{12.21}$$

Additional restrictions for applying these expressions are $t \ll h$, $t < W/2$ and $t/\Delta W < 0.75$. Notice also that the ratio $\Delta W/t$ is divergent for $t \to 0$.

Hammerstad and Jensen

E. Hammerstad and Ø. Jensen are using the method described by Wheeler [38] to account for a non-zero strip thickness. However, some modifications in his equations have been made, which give better accuracy for narrow strips and for substrates with low dielectric constant. For the homogeneous media the correction is

$$\Delta W_1 = \frac{t}{\pi} \cdot \ln \left(1 + \frac{4e}{\frac{t}{h} \cdot \coth^2 \sqrt{6.517W}}\right) \tag{12.22}$$

and for the mixed media the correction is

$$\Delta W_r = 0.5 \cdot \Delta W_1 \cdot \left(1 + \operatorname{sech}\sqrt{\varepsilon_r - 1}\right) \tag{12.23}$$

By defining corrected strip widths, $W_1 = W + \Delta W_1$ and $W_r = W + \Delta W_r$, the effect of strip thickness may be included in the equations (12.4) and (12.15).

$$Z_L\left(W, h, t, \varepsilon_r\right) = \frac{Z_{L1}\left(W_r, h\right)}{\sqrt{\varepsilon_{r_{eff}}\left(W_r, h, \varepsilon_r\right)}} \tag{12.24}$$

$$\varepsilon_{r_{eff}}\left(W, h, t, \varepsilon_r\right) = \varepsilon_{r_{eff}}\left(W_r, h, \varepsilon_r\right) \cdot \left(\frac{Z_{L1}\left(W_1, h\right)}{Z_{L1}\left(W_r, h\right)}\right)^2 \tag{12.25}$$

12.1.4 Dispersion

Dispersion can be a strong effect in microstrip transmission lines due to their inhomogeneity. Typically, as frequency is increased, $\varepsilon_{r_{eff}}$ increases in a non-linear manner, approaching an asymptotic value. Dispersion affects characteristic impedance in a similar way.

Kirschning and Jansen

The dispersion formulae given by Kirschning and Jansen [39] is meant to have an accuracy better than 0.6% in the range $0.1 \leq W/h \leq 100$, $1 \leq \varepsilon_r \leq 20$ and $0 \leq h/\lambda_0 \leq 0.13$, i.e. up to about 60GHz for 25mm substrates.

$$\varepsilon_r(f) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{r_{eff}}}{1 + P(f)} \tag{12.26}$$

with

$$P(f) = P_1 P_2 \cdot \left((0.1844 + P_3 P_4) \cdot f_n \right)^{1.5763} \tag{12.27}$$

$$P_1 = 0.27488 + \left(0.6315 + \frac{0.525}{(1 + 0.0157 \cdot f_n)^{20}} \right) \cdot \frac{W}{h} - 0.065683 \cdot \exp\left(-8.7513 \frac{W}{h} \right) \tag{12.28}$$

$$P_2 = 0.33622 \cdot \left(1 - \exp\left(-0.03442 \cdot \varepsilon_r \right) \right) \tag{12.29}$$

$$P_3 = 0.0363 \cdot \exp\left(-4.6 \frac{W}{h} \right) \cdot \left(1 - \exp\left(- \left(\frac{f_n}{38.7} \right)^{4.97} \right) \right) \tag{12.30}$$

$$P_4 = 1 + 2.751 \cdot \left(1 - \exp\left(- \left(\frac{\varepsilon_r}{15.916} \right)^{8} \right) \right) \tag{12.31}$$

$$f_n = f \cdot h = \text{normalised frequency in } [\text{GHz} \cdot \text{mm}] \tag{12.32}$$

Dispersion of the characteristic impedance according to [40] can be applied for the range $0 \leq h/\lambda_0 \leq 0.1$, $0.1 \leq W/h \leq 10$ and for substrates with $1 \leq \varepsilon_r \leq 18$ and is is given by the following set of equations.

$$R_1 = 0.03891 \cdot \varepsilon_r^{1.4} \tag{12.33}$$

$$R_2 = 0.267 \cdot u^{7.0} \tag{12.34}$$

$$R_3 = 4.766 \cdot \exp\left(-3.228 \cdot u^{0.641} \right) \tag{12.35}$$

$$R_4 = 0.016 + (0.0514 \cdot \varepsilon_r)^{4.524} \tag{12.36}$$

$$R_5 = (f_n / 28.843)^{12.0} \tag{12.37}$$

$$R_6 = 22.20 \cdot u^{1.92} \tag{12.38}$$

and

$$R_7 = 1.206 - 0.3144 \cdot \exp\left(-R_1 \right) \cdot \left(1 - \exp\left(-R_2 \right) \right) \tag{12.39}$$

$$R_8 = 1 + 1.275 \cdot \left(1 - \exp\left(-0.004625 \cdot R_3 \cdot \varepsilon_r^{1.674} \cdot (f_n / 18.365)^{2.745} \right) \right) \tag{12.40}$$

$$R_9 = 5.086 \cdot \frac{R_4 \cdot R_5}{0.3838 + 0.386 \cdot R_4} \cdot \frac{\exp\left(-R_6 \right)}{1 + 1.2992 \cdot R_5} \cdot \frac{(\varepsilon_r - 1)^6}{1 + 10 \cdot (\varepsilon_r - 1)^6} \tag{12.41}$$

and

$$R_{10} = 0.00044 \cdot \varepsilon_r^{2.136} + 0.0184 \tag{12.42}$$

$$R_{11} = \frac{(f_n/19.47)^6}{1 + 0.0962 \cdot (f_n/19.47)^6} \tag{12.43}$$

$$R_{12} = \frac{1}{1 + 0.00245 \cdot u^2} \tag{12.44}$$

$$R_{13} = 0.9408 \cdot \varepsilon_r(f)^{R_8} - 0.9603 \tag{12.45}$$

$$R_{14} = (0.9408 - R_9) \cdot \varepsilon_{r_{eff}}^{R_8} - 0.9603 \tag{12.46}$$

$$R_{15} = 0.707 \cdot R_{10} \cdot (f_n/12.3)^{1.097} \tag{12.47}$$

$$R_{16} = 1 + 0.0503 \cdot \varepsilon_r^2 \cdot R_{11} \cdot \left(1 - \exp\left(-\left(u/15\right)^6\right)\right) \tag{12.48}$$

$$R_{17} = R_7 \cdot \left(1 - 1.1241 \cdot \frac{R_{12}}{R_{16}} \cdot \exp\left(-0.026 \cdot f_n^{1.15656} - R_{15}\right)\right) \tag{12.49}$$

Finally the frequency-dependent characteristic impedance can be written as

$$Z_L(f_n) = Z_L(0) \cdot \left(\frac{R_{13}}{R_{14}}\right)^{R_{17}} \tag{12.50}$$

The abbreviations used in these expressions are f_n for the normalized frequency as denoted in eq. (12.32) and $u = W/h$ for the microstrip width normalised with respect to the substrate height. The terms $Z_L(0)$ and $\varepsilon_{r_{eff}}$ denote the static values of microstrip characteristic impedance and effective dielectric constant. The value $\varepsilon_r(f)$ is the frequency dependent effective dielectric constant computed according to [39].

R.H. Jansen and M. Kirschning remark in [40] for the implementation of the expressions on a computer, R_1, R_2 and R_6 should be restricted to numerical values less or equal 20 in order to prevent overflow.

Yamashita

The values obtained by the approximate dispersion formula as given by E. Yamashita [41] deviate within 1% in a wide frequency range compared to the integral equation method used to derive the functional approximation. The formula assumes the knowledge of the quasi-static effective dielectric constant. The applicable ranges of the formula are $2 < \varepsilon_r < 16$, $0.06 < W/h < 16$ and $0.1\text{GHz} < f < 100\text{GHz}$. Though the lowest usable frequency is limited by 0.1GHz, the propagation constant for frequencies less than 0.1GHz has been given as the quasi-static one.

$$\varepsilon_r(f) = \varepsilon_{r_{eff}} \cdot \left(\frac{1 + \dfrac{1}{4} \cdot k \cdot F^{1.5}}{1 + \dfrac{1}{4} \cdot F^{1.5}}\right)^2 \tag{12.51}$$

with

$$k = \sqrt{\frac{\varepsilon_r}{\varepsilon_{r_{eff}}}} \tag{12.52}$$

$$F = \frac{4 \cdot h \cdot f \cdot \sqrt{\varepsilon_r - 1}}{c_0} \cdot \left(0.5 + \left(1 + 2 \cdot \log\left(1 + \frac{W}{h}\right)\right)^2\right) \tag{12.53}$$

Kobayashi

The dispersion formula presented by M. Kobayashi [42], derived by comparison to a numerical model, has a high degree of accuracy, better than 0.6% in the range $0.1 \leq W/h \leq 10$, $1 < \varepsilon_r \leq 128$ and any h/λ_0 (no frequency limits).

$$\varepsilon_r(f) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{r_{eff}}}{1 + \left(\dfrac{f}{f_{50}}\right)^m} \tag{12.54}$$

with

$$f_{50} = \frac{c_0}{2\pi \cdot h \cdot \left(0.75 + \left(0.75 - \dfrac{0.332}{\varepsilon_r^{1.73}}\right)\dfrac{W}{h}\right)} \cdot \frac{\arctan\left(\varepsilon_r \cdot \sqrt{\dfrac{\varepsilon_{r_{eff}} - 1}{\varepsilon_r - \varepsilon_{r_{eff}}}}\right)}{\sqrt{\varepsilon_r - \varepsilon_{r_{eff}}}} \tag{12.55}$$

$$m = m_0 \cdot m_c \ \ (\leq 2.32) \tag{12.56}$$

$$m_0 = 1 + \frac{1}{1 + \sqrt{\dfrac{W}{h}}} + 0.32 \cdot \left(\frac{1}{1 + \sqrt{\dfrac{W}{h}}}\right)^3 \tag{12.57}$$

$$m_c = \begin{cases} 1 + \dfrac{1.4}{1 + \dfrac{W}{h}} \cdot \left(0.15 - 0.235 \cdot \exp\left(-0.45\dfrac{f}{f_{50}}\right)\right) & \text{for } W/h \leq 0.7 \\ 1 & \text{for } W/h \geq 0.7 \end{cases} \tag{12.58}$$

Getsinger

Based upon measurements of dispersion curves for microstrip lines on alumina substrates 0.025 and 0.050 inch thick W. J. Getsinger [43] developed a very simple , closed-form expression that allow slide-rule prediction of microstrip dispersion.

$$\varepsilon_r(f) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{r_{eff}}}{1 + G \cdot \left(\dfrac{f}{f_p}\right)^2} \tag{12.59}$$

with

$$f_p = \frac{Z_L}{2\mu_0 h} \tag{12.60}$$

$$G = 0.6 + 0.009 \cdot Z_L \tag{12.61}$$

Also based upon measurements of microstrip lines 0.1, 0.25 and 0.5 inch in width on a 0.250 inch thick alumina substrate Getsinger [44] developed two different dispersion models for the characteristic impedance.

- wave impedance model published in [44]

$$Z_L(f) = Z_L \cdot \sqrt{\frac{\varepsilon_{r_{eff}}}{\varepsilon_r(f)}} \tag{12.62}$$

- group-delay model published in [45]

$$Z_L(f) = Z_L \cdot \sqrt{\frac{\varepsilon_r(f)}{\varepsilon_{r_{eff}}}} \cdot \frac{1}{1 + D(f)} \tag{12.63}$$

with

$$D(f) = \frac{(\varepsilon_r - \varepsilon_r(f)) \cdot \left(\varepsilon_r(f) - \varepsilon_{r_{eff}}\right)}{\varepsilon_r(f) \cdot \left(\varepsilon_r - \varepsilon_{r_{eff}}\right)} \tag{12.64}$$

Hammerstad and Jensen

The dispersion formulae of E. Hammerstad and Ø. Jensen [37] give good results for all types of substrates (not as limited as Getsinger's formulae). The impedance dispersion model is based upon a parallel-plate model using the theory of dielectrics.

$$\varepsilon_r(f) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{r_{eff}}}{1 + G \cdot \left(\dfrac{f}{f_p}\right)^2} \tag{12.65}$$

with

$$f_p = \frac{Z_L}{2\mu_0 h} \tag{12.66}$$

$$G = \frac{\pi^2}{12} \cdot \frac{\varepsilon_r - 1}{\varepsilon_{r_{eff}}} \cdot \sqrt{\frac{2\pi \cdot Z_L}{Z_{F0}}} \tag{12.67}$$

$$Z_L(f) = Z_L \cdot \sqrt{\frac{\varepsilon_{r_{eff}}}{\varepsilon_r(f)}} \cdot \frac{\varepsilon_r(f) - 1}{\varepsilon_{r_{eff}} - 1} \tag{12.68}$$

Edwards and Owens

The authors T. C. Edwards and R. P. Owens [46] developed a dispersion formula based upon measurements of microstrip lines on sapphire in the range $10\Omega \le Z_L \le 100\Omega$ and up to 18GHz. The procedure was repeated for several microstrip width-to-substrate-height ratios (W/h) between 0.1 and 10.

$$\varepsilon_r(f) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{r_{eff}}}{1 + P} \tag{12.69}$$

with

$$P = \left(\frac{h}{Z_L}\right)^{1.33} \cdot \left(0.43 f^2 - 0.009 f^3\right) \tag{12.70}$$

where h is in millimeters and f is in gigahertz. Their new dispersion equation involving the polynomial, which was developed to predict the fine detail of the experimental $\varepsilon_r(f)$ versus frequency curves, includes two empicical parameters. However, it seems the formula is not too sensitive to changes in substrate parameters.

Pramanick and Bhartia

P. Bhartia and P. Pramanick [47] developed dispersion equations without any empirical quantity. Their work expresses dispersion of the dielectric constant and characteristic impedance in terms of a single inflection frequency.

For the frequency-dependent relative dielectric constant they propose

$$\varepsilon_r(f) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{r_{eff}}}{1 + \left(\dfrac{f}{f_T}\right)^2} \tag{12.71}$$

where

$$f_T = \sqrt{\frac{\varepsilon_r}{\varepsilon_{r_{eff}}}} \cdot \frac{Z_L}{2\mu_0 h} \tag{12.72}$$

Dispersion of the characteristic impedance is accounted by

$$Z_L(f) = \frac{Z_{F0} \cdot h}{W_e(f) \cdot \sqrt{\varepsilon_r(f)}} \tag{12.73}$$

whence

$$W_e(f) = W + \frac{W_{eff} - W}{1 + \left(\dfrac{f}{f_T}\right)^2} \quad \text{and} \quad W_{eff} = \frac{Z_{F0} \cdot h}{Z_L \cdot \sqrt{\varepsilon_{r_{eff}}}} \tag{12.74}$$

Schneider

Martin V. Schneider [48] proposed the following equation for the dispersion of the effective dielectric constant of a single microstrip line. The estimated error is less than 3%.

$$\varepsilon_r(f) = \varepsilon_{r_{eff}} \cdot \left(\frac{1 + f_n^2}{1 + k \cdot f_n^2}\right)^2 \tag{12.75}$$

with

$$f_n = \frac{4h \cdot f \cdot \sqrt{\varepsilon_r - 1}}{c_0} \quad \text{and} \quad k = \sqrt{\frac{\varepsilon_{r_{eff}}}{\varepsilon_r}} \tag{12.76}$$

For the dispersion of the characteristic impedance he uses the same wave guide impedance model as Getsinger in his first approach to the problem.

$$Z_L(f) = Z_L \cdot \sqrt{\frac{\varepsilon_{r_{eff}}}{\varepsilon_r(f)}} \tag{12.77}$$

12.1.5 Transmission losses

The attenuation of a microstrip line consists of conductor (ohmic) losses, dielectric (substrate) losses, losses due to radiation and propagation of surface waves and higher order modes.

$$\alpha = \alpha_c + \alpha_d + \alpha_r + \alpha_s \tag{12.78}$$

The conversion into dB/m unit is a multiplication:

$$\alpha[\text{dB/m}] = \alpha \cdot \frac{20}{\ln(10)} \tag{12.79}$$

Dielectric losses

Dielectric loss is due to the effects of finite loss tangent $\tan \delta_d$. Basically the losses rise proportional over the operating frequency. For common microwave substrate materials like Al_2O_3 ceramics with a loss tangent δ_d less than 10^{-3} the dielectric losses can be neglected compared to the conductor losses.

For the inhomogeneous line, an effective dielectric filling fraction give that proportion of the transmission line's cross section not filled by air. For microstrip lines, the result is

$$\alpha_d = \frac{\varepsilon_r}{\sqrt{\varepsilon_{r,eff}(0)}} \cdot \frac{\varepsilon_{r,eff}(0) - 1}{\varepsilon_r - 1} \cdot \frac{\pi}{\lambda_0} \cdot \tan \delta_d \tag{12.80}$$

whereas

$\varepsilon_{r,eff}(0)$	effective relative permittivity without dispersion
δ_d	dielectric loss tangent

Conductor losses

E. Hammerstad and Ø. Jensen [37] proposed the following equation for the conductor losses. The surface roughness of the substrate is necessary to account for an asymptotic increase seen in the apparent surface resistance with decreasing skin depth. This effect is considered by the correction factor K_r. The current distribution factor K_i is a very good approximation provided that the strip thickness exceeds three skin depths ($t > 3\delta$).

$$\alpha_c = \frac{R_{sh}}{Z_L(0) \cdot W} \cdot K_r \cdot K_i \tag{12.81}$$

with

$$R_{sh} = \frac{\rho}{\delta} = \sqrt{\rho \cdot \frac{\omega \cdot \mu}{2}} = \sqrt{\rho \cdot \pi \cdot f \cdot \mu} \tag{12.82}$$

$$K_i = \exp\left(-1.2 \left(\frac{Z_L(0)}{Z_{F0}}\right)^{0.7}\right) \tag{12.83}$$

$$K_r = 1 + \frac{2}{\pi} \arctan\left(1.4 \left(\frac{\Delta}{\delta}\right)^2\right) \tag{12.84}$$

whereas

$Z_L(0)$	characteristic impedance of microstrip without dispersion
R_{sh}	sheet resistance of conductor material (skin resistance)
ρ	specific resistance of conductor
δ	skin depth
K_i	current distribution factor
K_r	correction term due to surface roughness
Δ	effective (rms) surface roughness of substrate
Z_{F0}	wave impedance in vacuum

Radiation losses

The radiation losses of a free (unshielded) and matched (reflectionless terminated) microstrip transmission line may be calculated as follows [49]:

$$\alpha_r = 60 \cdot \left(\frac{2\pi f \cdot h \cdot \sqrt{\epsilon_{r,eff}}}{c_0}\right)^2 \cdot \left(1 - \frac{\epsilon_{r,eff} - 1}{2 \cdot \sqrt{\epsilon_{r,eff}}} \cdot \ln\left(\frac{\sqrt{\epsilon_{r,eff}} + 1}{\sqrt{\epsilon_{r,eff}} - 1}\right)\right) \tag{12.85}$$

whereas

c_0	velocity of light in vacuum
f	frequency
h	height of substrate
$\epsilon_{r,eff}$	effective dielectric constant

12.1.6 Higher modes

In order for a transmission line to work properly, it's important that one mode only propagates. For the microstrip line this is the quasi-TEM mode. Higher-order modes appear above a certain frequency, i.e. the useability is limited to the frequency range below the cut-off of the first higher mode.

The cut-off frequency of the first TE mode (TE1) is:

$$f_c = \frac{c_0}{\sqrt{\epsilon_r} \cdot (2 \cdot W + 0.8 \cdot h)} \tag{12.86}$$

The cut-off frequency of surface waves (TM1 mode) is:

$$f_s = \frac{c_0}{\pi \cdot h} \cdot \sqrt{\frac{\arctan \epsilon_r}{2 \cdot (\epsilon_r - 1)}} \tag{12.87}$$

12.2 Parallel coupled microstrip lines

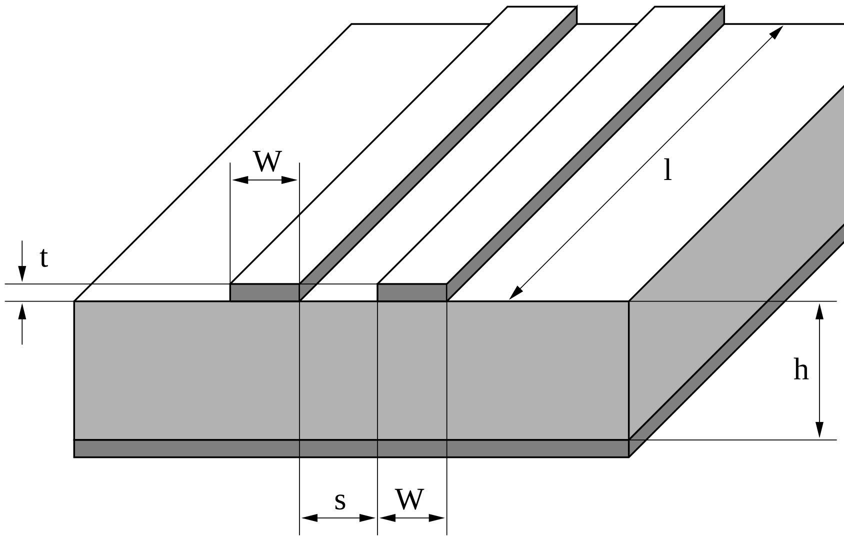


Figure 12.4: parallel coupled microstrip lines

12.2.1 Characteristic impedance and effective dielectric constant

Parallel coupled microstrip lines are defined by the characteristic impedance and the effective permittivity of the even and the odd mode. The y- and S-parameters are depicted in section 10.25.

Kirschning and Jansen

These quantities can very precisely be modeled by the following equations [50], [51].

Beforehand some normalised quantities (with microstrip line width W, spacing s between the lines and substrate height h) are introduced:

$$u = \frac{W}{h} \quad , \quad g = \frac{s}{h} \quad , \quad f_n = \frac{f}{\text{GHz}} \cdot \frac{h}{\text{mm}} = \frac{f}{\text{MHz}} \cdot h \tag{12.88}$$

The applicability of the described model is

$$0.1 \le u \le 10 \qquad , \qquad 0.1 \le g \le 10 \qquad , \qquad 1 \le \varepsilon_r \le 18 \tag{12.89}$$

The accuracies of the formulas holds for these ranges.

Static effective permittivity of even mode:

$$\varepsilon_{eff,e}(0) = 0.5 \cdot (\varepsilon_r + 1) + 0.5 \cdot (\varepsilon_r - 1) \cdot \left(1 + \frac{10}{v}\right)^{-a_e(v) \cdot b_e(\varepsilon_r)} \tag{12.90}$$

with

$$v = u \cdot \frac{20 + g^2}{10 + g^2} + g \cdot \exp\left(-g\right) \tag{12.91}$$

$$a_e\left(v\right) = 1 + \frac{1}{49} \cdot \ln\left(\frac{v^4 + \left(v/52\right)^2}{v^4 + 0.432}\right) + \frac{1}{18.7} \cdot \ln\left(1 + \left(\frac{v}{18.1}\right)^3\right) \tag{12.92}$$

$$b_e\left(\varepsilon_r\right) = 0.564 \cdot \left(\frac{\varepsilon_r - 0.9}{\varepsilon_r + 3}\right)^{0.053} \tag{12.93}$$

Static effective permittivity of odd mode:

$$\varepsilon_{eff,o}(0) = \left(0.5 \cdot \left(\varepsilon_r + 1\right) + a_o\left(u, \varepsilon_r\right) - \varepsilon_{eff}(0)\right) \cdot \exp\left(-c_o \cdot g^{d_o}\right) + \varepsilon_{eff}(0) \tag{12.94}$$

with

$$a_o\left(u, \varepsilon_r\right) = 0.7287 \cdot \left(\varepsilon_{eff}(0) - 0.5 \cdot \left(\varepsilon_r + 1\right)\right) \cdot \left(1 - \exp\left(-0.179 \cdot u\right)\right) \tag{12.95}$$

$$b_o\left(\varepsilon_r\right) = 0.747 \cdot \frac{\varepsilon_r}{0.15 + \varepsilon_r} \tag{12.96}$$

$$c_o = b_o(\varepsilon_r) - \left(b_o\left(\varepsilon_r\right) - 0.207\right) \cdot \exp\left(-0.414 \cdot u\right) \tag{12.97}$$

$$d_o = 0.593 + 0.694 \cdot \exp\left(-0.562 \cdot u\right) \tag{12.98}$$

whence $\varepsilon_{eff}(0)$ refers to the zero-thickness single microstrip line of width W according to [37] (see also eq. (12.15)).

The dispersion formulae for the odd and even mode write as follows.

$$\varepsilon_{eff,e,o}\left(f_n\right) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{eff,e,o}(0)}{1 + F_{e,o}\left(f_n\right)} \tag{12.99}$$

The frequency dependence for the even mode is

$$F_e\left(f_n\right) = P_1 \cdot P_2 \cdot \left(\left(P_3 \cdot P_4 + 0.1844 \cdot P_7\right) \cdot f_n\right)^{1.5763} \tag{12.100}$$

with

$$P_1 = 0.27488 + \left(0.6315 + \frac{0.525}{\left(1 + 0.0157 \cdot f_n\right)^{20}}\right) \cdot u - 0.065683 \cdot \exp\left(-8.7513 \cdot u\right) \tag{12.101}$$

$$P_2 = 0.33622 \cdot \left(1 - \exp\left(-0.03442 \cdot \varepsilon_r\right)\right) \tag{12.102}$$

$$P_3 = 0.0363 \cdot \exp\left(-4.6 \cdot u\right) \cdot \left(1 - \exp\left(-\left(f_n/38.7\right)^{4.97}\right)\right) \tag{12.103}$$

$$P_4 = 1 + 2.751 \cdot \left(1 - \exp\left(-\left(\varepsilon_r/15.916\right)^8\right)\right) \tag{12.104}$$

$$P_5 = 0.334 \cdot \exp\left(-3.3 \cdot \left(\varepsilon_r/15\right)^3\right) + 0.746 \tag{12.105}$$

$$P_6 = P_5 \cdot \exp\left(-\left(f_n/18\right)^{0.368}\right) \tag{12.106}$$

$$P_7 = 1 + 4.069 \cdot P_6 \cdot g^{0.479} \cdot \exp\left(-1.347 \cdot g^{0.595} - 0.17 \cdot g^{2.5}\right) \tag{12.107}$$

The frequency dependence for the odd mode is

$$F_o\left(f_n\right) = P_1 \cdot P_2 \cdot \left(\left(P_3 \cdot P_4 + 0.1844\right) \cdot f_n \cdot P_{15}\right)^{1.5763} \tag{12.108}$$

with

$$P_8 = 0.7168 \cdot \left(1 + \frac{1.076}{1 + 0.0576 \cdot (\varepsilon_r - 1)}\right) \tag{12.109}$$

$$P_9 = P_8 - 0.7913 \cdot \left(1 - \exp\left(-\left(f_n/20\right)^{1.424}\right)\right) \cdot \arctan\left(2.481 \cdot \left(\varepsilon_r/8\right)^{0.946}\right) \tag{12.110}$$

$$P_{10} = 0.242 \cdot (\varepsilon_r - 1)^{0.55} \tag{12.111}$$

$$P_{11} = 0.6366 \cdot \left(\exp\left(-0.3401 \cdot f_n\right) - 1\right) \cdot \arctan\left(1.263 \cdot \left(u/3\right)^{1.629}\right) \tag{12.112}$$

$$P_{12} = P_9 + \frac{1 - P_9}{1 + 1.183 \cdot u^{1.376}} \tag{12.113}$$

$$P_{13} = 1.695 \cdot \frac{P_{10}}{0.414 + 1.605 \cdot P_{10}} \tag{12.114}$$

$$P_{14} = 0.8928 + 0.1072 \cdot \left(1 - \exp\left(-0.42 \cdot \left(f_n/20\right)^{3.215}\right)\right) \tag{12.115}$$

$$P_{15} = \left|1 - 0.8928 \cdot \left(1 + P_{11}\right) \cdot \exp\left(-P_{13} \cdot g^{1.092}\right) \cdot P_{12}/P_{14}\right| \tag{12.116}$$

Up to $f_n = 25$ the maximum error of these equations is 1.4%.

The static characteristic impedance for the even mode writes as follows.

$$Z_{L,e}(0) = \sqrt{\frac{\varepsilon_{eff}(0)}{\varepsilon_{eff,e}(0)}} \cdot \frac{Z_L(0)}{1 - \dfrac{Z_L(0)}{377\Omega} \cdot \sqrt{\varepsilon_{eff}(0)} \cdot Q_4} \tag{12.117}$$

with

$$Q_1 = 0.8695 \cdot u^{0.194} \tag{12.118}$$

$$Q_2 = 1 + 0.7519 \cdot g + 0.189 \cdot g^{2.31} \tag{12.119}$$

$$Q_3 = 0.1975 + \left(16.6 + \left(8.4/g\right)^6\right)^{-0.387} + \frac{1}{241} \cdot \ln\left(\frac{g^{10}}{1 + \left(g/3.4\right)^{10}}\right) \tag{12.120}$$

$$Q_4 = \frac{Q_1}{Q_2} \cdot \frac{2}{\exp\left(-g\right) \cdot u^{Q_3} + \left(2 - \exp\left(-g\right)\right) \cdot u^{-Q_3}} \tag{12.121}$$

with $Z_L(0)$ and $\varepsilon_{eff}(0)$ being again quantities for a zero-thickness single microstrip line of width W according to [37] (see also eq. (12.15) and (12.5)).

The static characteristic impedance for the odd mode writes as follows.

$$Z_{L,o}(0) = \sqrt{\frac{\varepsilon_{eff}(0)}{\varepsilon_{eff,o}(0)}} \cdot \frac{Z_L(0)}{1 - \dfrac{Z_L(0)}{377\Omega} \cdot \sqrt{\varepsilon_{eff}(0)} \cdot Q_{10}} \tag{12.122}$$

with

$$Q_5 = 1.794 + 1.14 \cdot \ln\left(1 + \frac{0.638}{g + 0.517 \cdot g^{2.43}}\right) \tag{12.123}$$

$$Q_6 = 0.2305 + \frac{1}{281.3} \cdot \ln\left(\frac{g^{10}}{1 + (g/5.8)^{10}}\right) + \frac{1}{5.1} \cdot \ln\left(1 + 0.598 \cdot g^{1.154}\right) \tag{12.124}$$

$$Q_7 = \frac{10 + 190 \cdot g^2}{1 + 82.3 \cdot g^3} \tag{12.125}$$

$$Q_8 = \exp\left(-6.5 - 0.95 \cdot \ln(g) - (g/0.15)^5\right) \tag{12.126}$$

$$Q_9 = \ln(Q_7) \cdot (Q_8 + 1/16.5) \tag{12.127}$$

$$Q_{10} = \frac{Q_2 \cdot Q_4 - Q_5 \cdot \exp\left(\ln(u) \cdot Q_6 \cdot u^{-Q_9}\right)}{Q_2} = Q_4 - \frac{Q_5}{Q_2} \cdot u^{Q_6 \cdot u^{-Q_9}} \tag{12.128}$$

The accuracy of the static impedances is better than 0.6%.

Dispersion of the characteristic impedance for the even mode can be modeled by the following equations.

$$Z_{L,e}(f_n) = Z_{L,e}(0) \cdot \left(\frac{0.9408 \cdot (\varepsilon_{eff}(f_n))^{C_e} - 0.9603}{(0.9408 - d_e) \cdot (\varepsilon_{eff}(0))^{C_e} - 0.9603}\right)^{Q_0} \tag{12.129}$$

with

$$\begin{aligned} C_e = 1 + 1.275 \cdot \left(1 - \exp\left(-0.004625 \cdot p_e \cdot \varepsilon_r^{1.674} \cdot (f_n/18.365)^{2.745}\right)\right) \\ -Q_{12} + Q_{16} - Q_{17} + Q_{18} + Q_{20} \end{aligned} \tag{12.130}$$

$$d_e = 5.086 \cdot q_e \cdot \frac{r_e}{0.3838 + 0.386 \cdot q_e} \cdot \frac{\exp\left(-22.2 \cdot u^{1.92}\right)}{1 + 1.2992 \cdot r_e} \cdot \frac{(\varepsilon_r - 1)^6}{1 + 10 \cdot (\varepsilon_r - 1)^6} \tag{12.131}$$

$$p_e = 4.766 \cdot \exp\left(-3.228 \cdot u^{0.641}\right) \tag{12.132}$$

$$q_e = 0.016 + (0.0514 \cdot \varepsilon_r \cdot Q_{21})^{4.524} \tag{12.133}$$

$$r_e = (f_n/28.843)^{12} \tag{12.134}$$

and

$$Q_{11} = 0.893 \cdot \left(1 - \frac{0.3}{1 + 0.7 \cdot (\varepsilon_r - 1)}\right) \tag{12.135}$$

$$Q_{12} = 2.121 \cdot \frac{(f_n/20)^{4.91}}{1 + Q_{11} \cdot (f_n/20)^{4.91}} \cdot \exp\left(-2.87 \cdot g\right) \cdot g^{0.902} \tag{12.136}$$

$$Q_{13} = 1 + 0.038 \cdot (\varepsilon_r/8)^{5.1} \tag{12.137}$$

$$Q_{14} = 1 + 1.203 \cdot \frac{(\varepsilon_r/15)^4}{1 + \cdot (\varepsilon_r/15)^4} \tag{12.138}$$

$$Q_{15} = \frac{1.887 \cdot \exp\left(-1.5 \cdot g^{0.84}\right) \cdot g^{Q_{14}}}{1 + 0.41 \cdot (f_n/15)^3 \cdot \dfrac{u^{2/Q_{13}}}{0.125 + u^{1.626/Q_{13}}}} \tag{12.139}$$

$$Q_{16} = Q_{15} \cdot \left(1 + \frac{9}{1 + 0.403 \cdot (\varepsilon_r - 1)^2}\right) \tag{12.140}$$

$$Q_{17} = 0.394 \cdot \left(1 - \exp\left(-1.47 \cdot (u/7)^{0.672}\right)\right) \cdot \left(1 - \exp\left(-4.25\,(f_n/20)^{1.87}\right)\right) \tag{12.141}$$

$$Q_{18} = 0.61 \cdot \frac{1 - \exp\left(-2.13 \cdot (u/8)^{1.593}\right)}{1 + 6.544 \cdot g^{4.17}} \tag{12.142}$$

$$Q_{19} = \frac{0.21 \cdot g^4}{(1 + 0.18 \cdot g^{4.9}) \cdot (1 + 0.1 \cdot u^2) \cdot \left(1 + (f_n/24)^3\right)} \tag{12.143}$$

$$Q_{20} = Q_{19} \cdot \left(0.09 + \frac{1}{1 + 0.1 \cdot (\varepsilon_r - 1)^{2.7}}\right) \tag{12.144}$$

$$Q_{21} = \left|1 - 42.54 \cdot g^{0.133} \cdot \exp\left(-0.812 \cdot g\right) \cdot \frac{u^{2.5}}{1 + 0.033 \cdot u^{2.5}}\right| \tag{12.145}$$

With $\varepsilon_{eff}(f_n)$ being the single microstrip effective dielectric constant according to [39] (see eq. (12.26)) and Q_0 single microstrip impedance dispersion according to [40] (there denoted as R_{17}, see eq. (12.49)).

Dispersion of the characteristic impedance for the odd mode can be modeled by the following equations.

$$Z_{L,o}(f_n) = Z_L(f_n) + \frac{Z_{L,o}(0) \cdot \left(\dfrac{\varepsilon_{eff,o}(f_n)}{\varepsilon_{eff,o}(0)}\right)^{Q_{22}} - Z_L(f_n) \cdot Q_{23}}{1 + Q_{24} + (0.46 \cdot g)^{2.2} \cdot Q_{25}} \tag{12.146}$$

with

$$Q_{22} = 0.925 \cdot \frac{\left(f_n/Q_{26}\right)^{1.536}}{1 + 0.3 \cdot \left(f_n/30\right)^{1.536}} \tag{12.147}$$

$$Q_{23} = 1 + \frac{0.005 \cdot f_n \cdot Q_{27}}{\left(1 + 0.812 \cdot \left(f_n/15\right)^{1.9}\right) \cdot \left(1 + 0.025 \cdot u^2\right)} \tag{12.148}$$

$$Q_{24} = \frac{2.506 \cdot Q_{28} \cdot u^{0.894}}{3.575 + u^{0.894}} \cdot \left(\frac{(1 + 1.3 \cdot u) \cdot f_n}{99.25}\right)^{4.29} \tag{12.149}$$

$$Q_{25} = \frac{0.3 \cdot f_n^2}{10 + f_n^2} \cdot \left(1 + \frac{2.333 \cdot \left(\varepsilon_r - 1\right)^2}{5 + \left(\varepsilon_r - 1\right)^2}\right) \tag{12.150}$$

$$Q_{26} = 30 - \frac{22.2 \cdot \left(\dfrac{\varepsilon_r - 1}{13}\right)^{12}}{1 + 3 \cdot \left(\dfrac{\varepsilon_r - 1}{13}\right)^{12}} - Q_{29} \tag{12.151}$$

$$Q_{27} = 0.4 \cdot g^{0.84} \cdot \left(1 + \frac{2.5 \cdot \left(\varepsilon_r - 1\right)^{1.5}}{5 + \left(\varepsilon_r - 1\right)^{1.5}}\right) \tag{12.152}$$

$$Q_{28} = 0.149 \cdot \frac{\left(\varepsilon_r - 1\right)^3}{94.5 + 0.038 \cdot \left(\varepsilon_r - 1\right)^3} \tag{12.153}$$

$$Q_{29} = \frac{15.16}{1 + 0.196 \cdot \left(\varepsilon_r - 1\right)^2} \tag{12.154}$$

with $Z_L(f_n)$ being the frequency-dependent power-current characteristic impedance formulation of a single microstrip with width W according to [40] (see eq. (12.50)). Up to $f_n = 20$, the numerical error of $Z_{L,o}(f_n)$ and $Z_{L,e}(f_n)$ is less than 2.5%.

Hammerstad and Jensen

The equations given by E. Hammerstad and Ø. Jensen [37] represent the first generally valid model of coupled microstrips with an acceptable accuracy. The model equations have been validated in the range $0.1 \le u \le 10$ and $g \ge 0.01$, a range which should cover that used in practice.

The homogeneous mode impedances are

$$Z_{L,e,o}\left(u, g\right) = \frac{Z_L(u)}{1 - Z_L(u) \cdot \Phi_{e,o}\left(u, g\right)/Z_{F0}} \tag{12.155}$$

The effective dielectric constants are

$$\varepsilon_{eff,e,o}\left(u, g, \varepsilon_r\right) = \frac{\varepsilon_r + 1}{2} + \frac{\varepsilon_r - 1}{2} \cdot F_{e,o}\left(u, g, \varepsilon_r\right) \tag{12.156}$$

with

$$F_e\left(u, g, \varepsilon_r\right) = \left(1 + \frac{10}{\mu\left(u, g\right)}\right)^{-a(\mu) \cdot b(\varepsilon_r)} \tag{12.157}$$

$$F_o\left(u, g, \varepsilon_r\right) = f_o\left(u, g, \varepsilon_r\right) \cdot \left(1 + \frac{10}{u}\right)^{-a(u) \cdot b(\varepsilon_r)} \tag{12.158}$$

whence $a(u)$ and $b\left(\varepsilon_r\right)$ denote eqs. (12.16) and (12.17) of the single microstrip line. The characteristic impedance of the single microstrip line $Z_L\left(u\right)$ also defined in [37] is given by eq. (12.5).

The modifying equations for the even mode are as follows

$$\Phi_e\left(u,g\right) = \frac{\varphi(u)}{\Psi(g)\cdot\left(\alpha(g)\cdot u^{m(g)} + \left(1-\alpha(g)\right)\cdot u^{-m(g)}\right)} \tag{12.159}$$

$$\varphi(u) = 0.8645\cdot u^{0.172} \tag{12.160}$$

$$\Psi(g) = 1 + \frac{g}{1.45} + \frac{g^{2.09}}{3.95} \tag{12.161}$$

$$\alpha(g) = 0.5\cdot e^{-g} \tag{12.162}$$

$$m(g) = 0.2175 + \left(4.113 + \left(20.36/g\right)^6\right)^{-0.251} + \frac{1}{323}\cdot\ln\left(\frac{g^{10}}{1+\left(g/13.8\right)^{10}}\right) \tag{12.163}$$

The modifying equations for the odd mode are as follows

$$\Phi_o\left(u,g\right) = \Phi_e\left(u,g\right) - \frac{\theta(g)}{\Psi(g)}\cdot\exp\left(\beta(g)\cdot u^{-n(g)}\cdot\ln u\right) \tag{12.164}$$

$$\theta(g) = 1.729 + 1.175\cdot\ln\left(1 + \frac{0.627}{g + 0.327\cdot g^{2.17}}\right) \tag{12.165}$$

$$\beta(g) = 0.2306 + \frac{1}{301.8}\cdot\ln\left(\frac{g^{10}}{1+\left(g/3.73\right)^{10}}\right) + \frac{1}{5.3}\cdot\ln\left(1 + 0.646\cdot g^{1.175}\right) \tag{12.166}$$

$$n(g) = \left(\frac{1}{17.7} + \exp\left(-6.424 - 0.76\cdot\ln g - \left(g/0.23\right)^5\right)\right)\cdot\ln\left(\frac{10+68.3\cdot g^2}{1+32.5\cdot g^{3.093}}\right) \tag{12.167}$$

Furthermore

$$\mu\left(u,g\right) = g\cdot e^{-g} + u\cdot\frac{20+g^2}{10+g^2} \tag{12.168}$$

$$f_o\left(u,g,\varepsilon_r\right) = f_{o1}\left(g,\varepsilon_r\right)\cdot\exp\left(p(g)\cdot\ln u + q(g)\cdot\sin\left(\pi\cdot\log u\right)\right) \tag{12.169}$$

$$p(g) = \frac{\exp\left(-0.745\cdot g^{0.295}\right)}{\cosh\left(g^{0.68}\right)} \tag{12.170}$$

$$q(g) = \exp\left(-1.366 - g\right) \tag{12.171}$$

$$f_{o1}\left(g,\varepsilon_r\right) = 1 - \exp\left(-0.179\cdot g^{0.15} - \frac{0.328\cdot g^{r(g,\varepsilon_r)}}{\ln\left(e + \left(g/7\right)^{2.8}\right)}\right) \tag{12.172}$$

$$r\left(g,\varepsilon_r\right) = 1 + 0.15\cdot\left(1 - \frac{\exp\left(1-\left(\varepsilon_r-1\right)^2/8.2\right)}{1+g^{-6}}\right) \tag{12.173}$$

The quasi-static characteristic impedance $Z_L(u)$ of a zero-thickness single microstrip line denoted in eq. (12.155) can either be calculated using the below equations with $\varepsilon_{r_{eff}}$ being the quasi-static effective dielectric constant defined by eq. (12.15) or using eqs. (12.5) and (12.15).

$$Z_{L1}(u) = \frac{Z_{F0}}{u + 1.98\cdot u^{0.172}} \tag{12.174}$$

$$Z_L(u) = \frac{Z_{L1}(u)}{\sqrt{\varepsilon_{r_{eff}}}} \tag{12.175}$$

The errors in the even and odd mode impedances $Z_{L,e}$ and $Z_{L,e}$ were found to be less than 0.8% and less than 0.3% for the wavelengths.

The model does not include the effect of non-zero strip thickness or asymmetry. Dispersion is also not included. W. J. Getsinger [52] has proposed modifications to his single strip dispersion model, but unfortunately it is easily shown that the results are asymptotically wrong for extreme values of gap width.

In fact he correctly assumes that in the even mode the two strips are at the same potential, and the total current is twice that on a single strip, and dispersion for even-mode propagation is computed by substituting $Z_{L,e}/2$ for Z_L in eqs. (12.60) and (12.61). In the odd mode the two strips are at opposite potentials, and the voltage between strips is twice that of a single strip to ground. Thus the total mode impedance is twice that of a single strip, and the dispersion for odd-mode propagation is computed substituting $2Z_{L,o}$ for Z_L in eqs. (12.60) and (12.61).

$$\varepsilon_{r,e,o}\left(f\right) = \varepsilon_r - \frac{\varepsilon_r - \varepsilon_{r_{eff,e,o}}}{1 + G \cdot \left(\dfrac{f}{f_p}\right)^2} \tag{12.176}$$

with

$$f_p = \begin{cases} \dfrac{Z_{L,e}}{4\mu_0 h} & \text{even mode} \\ \\ \dfrac{Z_{L,o}}{\mu_0 h} & \text{odd mode} \end{cases} \tag{12.177}$$

$$G = \begin{cases} 0.6 + Z_{L,e} \cdot 0.0045 & \text{even mode} \\ \\ 0.6 + Z_{L,o} \cdot 0.018 & \text{odd mode} \end{cases} \tag{12.178}$$

12.2.2 Strip thickness correction

According to R.H. Jansen [53] corrected strip width values have been found in the range of technologically meaningful geometries to be

$$W_{t,e} = W + \Delta W \cdot \left(1 - 0.5 \cdot \exp\left(-0.69 \cdot \frac{\Delta W}{\Delta t}\right)\right) \tag{12.179}$$

$$W_{t,o} = W_{t,e} + \Delta t \tag{12.180}$$

with

$$\Delta t = \frac{2 \cdot t \cdot h}{s \cdot \varepsilon_r} \qquad \text{for} \qquad s \gg 2t \tag{12.181}$$

The author refers to the modifications of the strip width of a single microstrip line ΔW given by Hammerstad and Bekkadal. See also eq. (12.21) on page 175.

$$\Delta W = \begin{cases} \dfrac{t}{\pi} \cdot \left(1 + \ln\left(\dfrac{2h}{t}\right)\right) & \text{for } W > \dfrac{h}{2\pi} > 2t \\ \\ \dfrac{t}{\pi} \cdot \left(1 + \ln\left(\dfrac{4\pi W}{t}\right)\right) & \text{for } \dfrac{h}{2\pi} \geq W > 2t \end{cases} \tag{12.182}$$

For large spacings s the single line formulae (12.182) applies.

12.2.3 Transmission losses

The loss equations given by E. Hammerstad and Ø. Jensen [37] for the single microstrip line are also valid for coupled microstrips, provided that the dielectric filling factor, homogeneous impedance, and current distribution factor of the actual mode are used. The following approximation gives good results for odd and even current distribution factors (modification of eq. (12.83)).

$$K_{i,e} = K_{i,o} = \exp\left(-1.2 \cdot \left(\frac{Z_{L,e} + Z_{L,o}}{2 \cdot Z_{F0}}\right)^{0.7}\right) \tag{12.183}$$

12.3 Microstrip open

A microstrip open end can be modeled by a longer effective microstrip line length Δl as described by M. Kirschning, R.H. Jansen and N.H.L. Koster [54].

$$\frac{\Delta l}{h} = \frac{Q_1 \cdot Q_3 \cdot Q_5}{Q_4} \tag{12.184}$$

with

$$Q_1 = 0.434907 \cdot \frac{\varepsilon_{r,eff}^{0.81} + 0.26}{\varepsilon_{r,eff}^{0.81} - 0.189} \cdot \frac{(W/h)^{0.8544} + 0.236}{(W/h)^{0.8544} + 0.87} \tag{12.185}$$

$$Q_2 = 1 + \frac{(W/h)^{0.371}}{2.358 \cdot \varepsilon_r + 1} \tag{12.186}$$

$$Q_3 = 1 + \frac{0.5274}{\varepsilon_{r,eff}^{0.9236}} \cdot \arctan\left(0.084 \cdot (W/h)^{\frac{1.9413}{Q_2}}\right) \tag{12.187}$$

$$Q_4 = 1 + 0.0377 \cdot (6 - 5 \cdot \exp(0.036 \cdot (1 - \varepsilon_r))) \cdot \arctan\left(0.067 \cdot (W/h)^{1.456}\right) \tag{12.188}$$

$$Q_5 = 1 - 0.218 \cdot \exp(-7.5 \cdot W/h) \tag{12.189}$$

The numerical error is less than 2.5% for $0.01 \leq W/h \leq 100$ and $1 \leq \varepsilon_r \leq 50$.

Another microstrip open end model was published by E. Hammerstad [55]:

$$\frac{\Delta l}{h} = 0.102 \cdot \frac{W/h + 0.106}{W/h + 0.264} \cdot \left(1.166 + \frac{\varepsilon_r + 1}{\varepsilon_r} \cdot (0.9 + \ln(W/h + 2.475))\right) \tag{12.190}$$

Here the numerical error is less than 1.7% for $W/h < 20$.

In order to simplify calculations, the equivalent additional line length Δl can be transformed into an equivalent open end capacitance C_{end}:

$$C_{end} = C' \cdot \Delta l = \frac{\sqrt{\varepsilon_{r,eff}}}{c_0 \cdot Z_L} \Delta l \tag{12.191}$$

With C' being the capacitance per length and $c_0 = 299\ 792\ 458$ m/s being the vacuum light velocity.

12.4 Microstrip gap

A symmetrical microstrip gap can be modeled by two open ends with a capacitive series coupling between the two ends. The physical layout is shown in fig. 12.5.

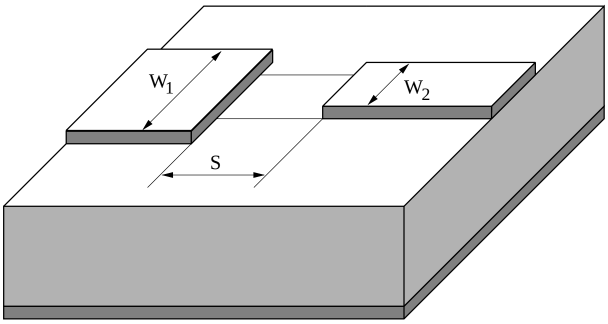


Figure 12.5: symmetrical microstrip gap layout

The equivalent π-network of a microstrip gap is shown in figure 12.6. The values of the components are according to [56] and [57].

$$C_S\ [\text{pF}] = 500 \cdot h \cdot \exp\left(-1.86 \cdot \frac{s}{h}\right) \cdot Q_1 \cdot \left(1 + 4.19\left(1 - \exp\left(-0.785 \cdot \sqrt{\frac{h}{W_1}} \cdot \frac{W_2}{W_1}\right)\right)\right) \tag{12.192}$$

$$C_{P1} = C_1 \cdot \frac{Q_2 + Q_3}{Q_2 + 1} \tag{12.193}$$

$$C_{P2} = C_2 \cdot \frac{Q_2 + Q_4}{Q_2 + 1} \tag{12.194}$$

with

$$Q_1 = 0.04598 \cdot \left(0.03 + \left(\frac{W_1}{h}\right)^{Q_5}\right) \cdot (0.272 + 0.07 \cdot \varepsilon_r) \tag{12.195}$$

$$Q_2 = 0.107 \cdot \left(\frac{W_1}{h} + 9\right) \cdot \left(\frac{s}{h}\right)^{3.23} + 2.09 \cdot \left(\frac{s}{h}\right)^{1.05} \cdot \frac{1.5 + 0.3 \cdot W_1/h}{1 + 0.6 \cdot W_1/h} \tag{12.196}$$

$$Q_3 = \exp\left(-0.5978 \cdot \left(\frac{W_2}{W_1}\right)^{1.35}\right) - 0.55 \tag{12.197}$$

$$Q_4 = \exp\left(-0.5978 \cdot \left(\frac{W_1}{W_2}\right)^{1.35}\right) - 0.55 \tag{12.198}$$

$$Q_5 = \frac{1.23}{1 + 0.12 \cdot (W_2/W_1 - 1)^{0.9}} \tag{12.199}$$

with C_1 and C_2 being the open end capacitances of a microstrip line (see eq. (12.191)). The numerical error of the capacitive admittances is less than 0.1mS for

$$\begin{aligned} 0.1 \le W_1/h \le 3 \\ 0.1 \le W_2/h \le 3 \\ 1 \le W_2/W_1 \le 3 \\ 6 \le \varepsilon_r \le 13 \\ 0.2 \le s/h \le \infty \\ 0.2\text{GHz} \le f \le 18\text{GHz} \end{aligned}$$

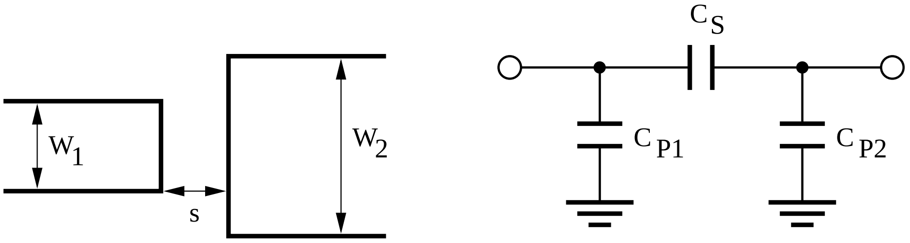


Figure 12.6: microstrip gap and its equivalent circuit

The Y-parameters for the given equivalent small signal circuit can be written as stated in eq. (12.200) and are easy to convert to scattering parameters.

$$Y = \begin{bmatrix} j\omega \cdot (C_{P1} + C_S) & -j\omega C_S \\ -j\omega C_S & j\omega \cdot (C_{P2} + C_S) \end{bmatrix} \tag{12.200}$$

12.5 Microstrip corner

The equivalent circuit of a microstrip corner is shown in fig. 12.7. The values of the components are as follows [57].

$$C \text{ [pF]} = W \cdot \left((10.35 \cdot \varepsilon_r + 2.5) \cdot \frac{W}{h} + (2.6 \cdot \varepsilon_r + 5.64) \right) \tag{12.201}$$

$$L \text{ [nH]} = 220 \cdot h \cdot \left(1 - 1.35 \cdot \exp\left(-0.18 \cdot \left(\frac{W}{h} \right)^{1.39} \right) \right) \tag{12.202}$$

The values for a 50% mitered bend are [57].

$$C \text{ [pF]} = W \cdot \left((3.93 \cdot \varepsilon_r + 0.62) \cdot \frac{W}{h} + (7.6 \cdot \varepsilon_r + 3.80) \right) \tag{12.203}$$

$$L \text{ [nH]} = 440 \cdot h \cdot \left(1 - 1.062 \cdot \exp\left(-0.177 \cdot \left(\frac{W}{h} \right)^{0.947} \right) \right) \tag{12.204}$$

With W being width of the microstrip line and h height of the substrate. These formulas are valid for $W/h = 0.2$ to 6.0 and for $\varepsilon_r = 2.36$ to 10.4 and up to 14 GHz. The precision is approximately 0.3%.

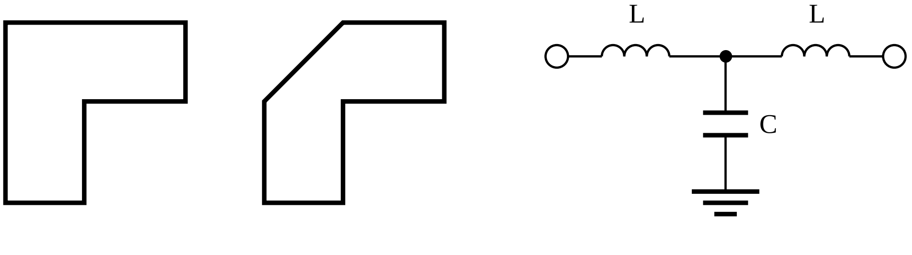


Figure 12.7: microstrip corner (left), mitered corner (middle) and equivalent circuit (right)

As could be seen the mitered bend reduces the capacitance and increases the inductance. Thus, the bend can be optimized in a way that the reflection is minimized [58]. Figure 12.8 and the following equations give the design rules.

$$X = \sqrt{2} \cdot W \cdot (0.52 + 0.65 \cdot \exp\left(-1.35 \cdot \frac{W}{h}\right) \tag{12.205}$$

$$A = \sqrt{2} \cdot X - W \tag{12.206}$$

The miter does not depend on the relative permittivity of the substrate.

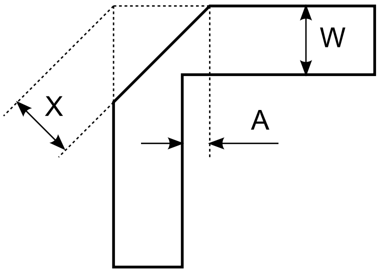


Figure 12.8: optimized microstrip bend

12.6 Microstrip impedance step

The equivalent circuit of a microstrip impedance step is quite simple: a capacitance C_S to ground and a series inductance L_S. The values are according to [59]:

$$C_S \text{ [pF] } = \sqrt{W_1 \cdot W_2} \cdot \left((10.1 \cdot \lg \varepsilon_r + 2.33) \cdot \frac{W_1}{W_2} - 12.6 \cdot \lg \varepsilon_r - 3.17 \right) \tag{12.207}$$

for $\varepsilon_r \le 10$ and $1.5 \le W_1/W_2 \le 3.5$ the error is $< 10\%$. Another formula determines C_S from the step geometry:

$$C_S = \frac{W_1 - W_2}{2} \cdot \left(\frac{\varepsilon_{r,eff1}}{Z_{L1} \cdot c_0} - \varepsilon_0 \cdot \varepsilon_r \cdot \frac{W_1}{h} \right) \tag{12.208}$$

The series inductance located at the port with the broader width is according to [60]:

$$\begin{aligned} L_S = & \frac{Z_{L1} \cdot W_{eff1}}{\pi \cdot f \cdot \lambda_1} \cdot \\ & \left[\ln\left(\frac{1-a^2}{4a} \cdot \left(\frac{1+a}{1-a} \right)^{0.5 \cdot (a + 1/a)} \right) + 2 \cdot \frac{A + B + 2 \cdot D}{A \cdot B - D^2} \right. \\ & \left. + \left(\frac{W_{eff1}}{4 \cdot \lambda_1} \right)^2 \cdot \left(\frac{1-a}{1+a} \right)^{4a} \cdot \left(\frac{5a^2 - 1}{1 - a^2} + \frac{4a^2 \cdot D}{3 \cdot A} \right)^2 \right] \end{aligned} \tag{12.209}$$

$$A = \left(\frac{1+a}{1-a} \right)^{2a} \cdot \left[\frac{1 + \sqrt{1 - (W_{eff1}/\lambda_1)^2}}{1 - \sqrt{1 - (W_{eff1}/\lambda_1)^2}} \right] - \frac{1 + 3a^2}{1 - a^2} \tag{12.210}$$

$$B = \left(\frac{1+a}{1-a} \right)^{2/a} \cdot \left[\frac{1 + \sqrt{1 - (W_{eff2}/\lambda_2)^2}}{1 - \sqrt{1 - (W_{eff2}/\lambda_2)^2}} \right] - \frac{3 + a^2}{1 - a^2} \tag{12.211}$$

$$D = \left(\frac{4a}{1 - a^2} \right)^2 \tag{12.212}$$

$$a = \frac{W_{eff2}}{W_{eff1}} \tag{12.213}$$

$$\lambda = \frac{c_0}{f \cdot \sqrt{\epsilon_{r,eff}}} \tag{12.214}$$

$$W_{eff} = \frac{W_{eff,static} + f_p \cdot W}{1 + f_p} \tag{12.215}$$

$$W_{eff,static} = \frac{Z_{F0} \cdot h}{Z_L \cdot \epsilon_{r,eff}} \tag{12.216}$$

$$f_p = \frac{2 \cdot W \cdot f \cdot \epsilon_{r,eff}}{c_0} \tag{12.217}$$

with f being frequency and c_0 being light velocity in vacuum.

12.7 Microstrip tee junction

A model of a microstrip tee junction is published in [55]. Figure 12.9 shows a unsymmetrical microstrip tee with the main arms consisting of port a and b and with the side arm consisting of port 2. The following model describes the gray area. The equivalent circuit is depicted in figure 12.10. It consists of a shunt reactance B_T, one transformer in each main arm (ratios T_a and T_b) and a microstrip line in each arm (width W_a, W_b and W_2).

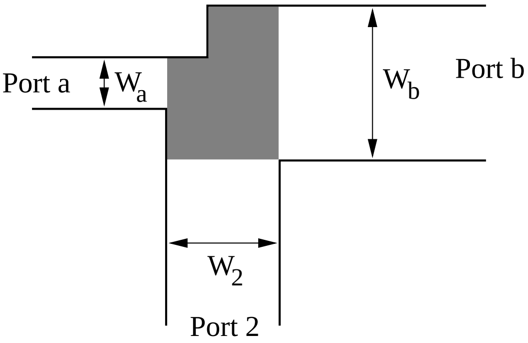


Figure 12.9: unsymmetrical microstrip tee (see text)

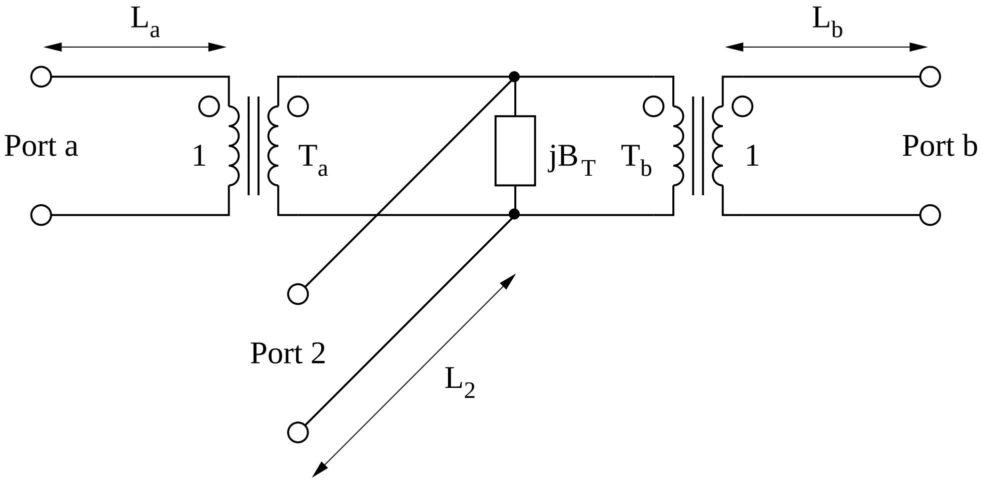


Figure 12.10: equivalent circuit of unsymmetrical microstrip tee

First, let us define some quantities. Each of them is used in the equations below with an index of the arm they belong to (a, b or 2).

$$\text{equivalent parallel plate line width:} \qquad D = \frac{Z_{F0}}{\sqrt{\varepsilon_{r,eff}}} \cdot \frac{h}{Z_L} \tag{12.218}$$

where Z_{F0} is vacuum field impedance, h height of substrate, $\varepsilon_{r,eff}$ effective, relative dielectric constant, Z_L microstrip line impedance.

$$\text{first higher order mode cut-off frequency:} \qquad f_p = 4 \cdot 10^5 \cdot \frac{Z_L}{h} \tag{12.219}$$

The main arm displacements of the reference planes from the center lines are (index x stand for a or b):

$$d_x = 0.055 \cdot D_2 \cdot \frac{Z_{L,x}}{Z_{L,2}} \cdot \left(1 - 2 \cdot \frac{Z_{L,x}}{Z_{L,2}} \cdot \left(\frac{f}{f_{p,x}}\right)^2\right) \tag{12.220}$$

The length of the line in the main arms is:

$$L_x = 0.5 \cdot W_2 - d_x \tag{12.221}$$

where f is frequency.

The side arm displacement of the reference planes from the center lines is:

$$d_2 = \sqrt{D_a \cdot D_b} \cdot \left(0.5 - R \cdot \left(0.05 + 0.7 \cdot \exp\left(-1.6 \cdot R\right) + 0.25 \cdot Q - 0.17 \cdot \ln R\right)\right) \tag{12.222}$$

The length of the line in the side arm is:

$$L_2 = 0.5 \cdot \max\left(W_a, W_b\right) - d_2 \tag{12.223}$$

where $\max\left(x, y\right)$ is the larger of the both quantities, R and Q are:

$$R = \frac{\sqrt{Z_{L,a} \cdot Z_{L,b}}}{Z_{L,2}} \qquad Q = \frac{f^2}{f_{p,a} \cdot f_{p,b}} \cdot R \tag{12.224}$$

Turn ratio of transformers in the side arms:

$$T_x^2 = 1 - \pi \cdot \left(\frac{f}{f_{p,x}}\right)^2 \cdot \left(\frac{1}{12} \cdot \left(\frac{Z_{L,x}}{Z_{L,2}}\right)^2 + \left(0.5 - \frac{d_2}{D_x}\right)^2\right) \tag{12.225}$$

The transformer ratios have to be limited to prevent them from becoming negative above a specific frequency, e.g.:

$$T_x^2 = \min(T_x^2, 10^{-12}) \tag{12.226}$$

Shunt susceptance:

$$\begin{aligned} B_T = 5.5 \cdot \frac{\mu_0 \cdot h \cdot f}{\sqrt{Z_{L,a} \cdot Z_{L,b}}} \cdot \frac{\varepsilon_r + 2}{\varepsilon_r} \cdot \frac{1}{Z_{L,2} \cdot T_a \cdot T_b} \cdot \frac{\sqrt{d_a \cdot d_b}}{D_2} \\ \cdot \left(1 + 0.9 \cdot \ln R + 4.5 \cdot Q - 4.4 \cdot \exp\left(-1.3 \cdot R\right) - 20 \cdot \left(\frac{Z_{L,2}}{Z_{F0}}\right)^2\right) \end{aligned} \tag{12.227}$$

Please note that the main arm displacements in eq. (12.220) yield two small microstrip lines at each main arm and the side arm displacement of eq. (12.222) results in a small microstrip strip line as well, but with negative length, i.e. kind of phaseshifter here.

For better implementation of the microstrip tee (figure 12.10) the device parameter of the equivalent circuit (three microstrip lines, two transformers and the shunt susceptance) are given below. First the cascade of transmission line Y_{ij} and transformer (turn ratio T) is put together:

$$(\underline{Y}_T) = \begin{bmatrix} Y_{11} & \dfrac{Y_{12}}{T} \\ \dfrac{Y_{21}}{T} & \dfrac{Y_{22}}{T^2} \end{bmatrix} \tag{12.228}$$

The y-parameters of the complete microstrip tee can now be calculated using the y-parameters $Y_{ij}^{(1\ldots3)}$ of the three microstrip arms.

$$Y_{ii}^G = Y_{11}^{(i)} - \frac{Y_{12}^{(i)} \cdot Y_{21}^{(i)}}{X} \tag{12.229}$$

$$Y_{ij}^G = -\frac{Y_{12}^{(i)} \cdot Y_{21}^{(j)}}{X} \qquad \text{with} \quad i \neq j \tag{12.230}$$

$$\text{with} \quad X = Y_{22}^{(1)} + Y_{22}^{(2)} + Y_{22}^{(3)} + j \cdot B \tag{12.231}$$

12.8 Microstrip cross

The most useful model of a microstrip cross have been published in [61, 62]. Fig. 12.11 shows the equivalent circuit (right-hand side) and the scheme with dimensions (left-hand side). The hatched area in the scheme marks the area modeled by the equivalent circuit. As can be seen the model require the microstrip width of line 1 and 3, as well as the one of line 2 and 4 to equal each other. Furthermore the permittivity of the substrat must be $\epsilon_r = 9.9$. The component values are calculated as follows:

$$X = \ln\left(\frac{W_1}{h}\right) \cdot \left(86.6 \cdot \frac{W_2}{h} - 30.9 \cdot \sqrt{\frac{W_2}{h}} + 367\right) + \left(\frac{W_2}{h}\right)^3 + 74 \cdot \frac{W_2}{h} + 130 \tag{12.232}$$

$$\begin{aligned} C_1 = C_2 = C_3 = C_4 \\ = 10^{-12} \cdot W_1 \cdot \left(0.25 \cdot X \cdot \left(\frac{h}{W_1}\right)^{1/3} - 60 + \frac{h}{2 \cdot W_2} - 0.375 \cdot \frac{W_1}{h} \cdot \left(1 - \frac{W_2}{h}\right)\right) \end{aligned} \tag{12.233}$$

$$Y = 165.6 \cdot \frac{W_2}{h} + 31.2\sqrt{\frac{W_2}{h}} - 11.8 \cdot \left(\frac{W_2}{h}\right)^2 \tag{12.234}$$

$$L_1 = L_3 = 10^{-9} \cdot h \cdot \left(Y \cdot \frac{W_1}{h} - 32 \cdot \frac{W_2}{h} + 3\right) \cdot \left(\frac{h}{W_1}\right)^{1.5} \tag{12.235}$$

$$L_5 = 10^{-9} \cdot h \cdot \left(5 \cdot \frac{W_2}{h} \cdot \cos\left(\frac{\pi}{2} \cdot \left(1.5 - \frac{W_1}{h}\right)\right) - \left(1 + \frac{7 \cdot h}{W_1}\right) \cdot \frac{h}{W_2} - 337.5\right) \tag{12.236}$$

The equation of L_2 and L_4 is obtained from the one of L_1 by exchanging the indices (W_1 and W_2). Note that L_5 is negative, so the model is unphysical without external microstrip lines. The above-mentioned equations are accurate to within 5% for $0.3 \leq W_1/h \leq 3$ and $0.1 \leq W_2/h \leq 3$ (value of $C_1 \ldots C_4$) or for $0.5 \leq W_{1,2}/h \leq 2$ (value of $L_1 \ldots L_3$), respectively.

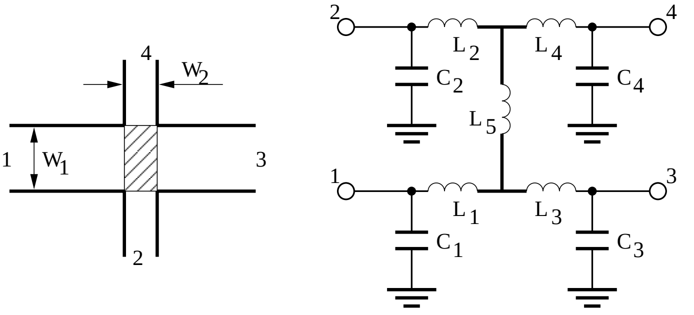


Figure 12.11: single-symmetrical microstrip cross and its model

Some improvement should be added to the original model:

1. Comparisons with real life show that the value of L_5 is too large. Multiplying it by 0.8 leads to much better results.

2. The model can be expanded for substrates with $\epsilon_r \neq 9.9$ by modifying the values of the capacitances:

$$C_x = C_x(\epsilon_r = 9.9) \cdot \frac{Z_0(\epsilon_r = 9.9, W = W_x)}{Z_0(\epsilon_r = \epsilon_{r,sub}, W = W_x)} \cdot \sqrt{\frac{\epsilon_{eff}(\epsilon_r = \epsilon_{r,sub}, W = W_x)}{\epsilon_{eff}(\epsilon_r = 9.9, W = W_x)}} \tag{12.237}$$

   The equations of Z_0 and ϵ_{eff} are the ones from the microstrip lines.

A useful model for an unsymmetrical cross junction has never been published. Nonetheless, as long as the lines that lie opposite are not to different in width, the model described here can be used as a first order approximation. This is performed by replacing W_1 and W_2 by the arithmetic mean of the line widths that lie opposite. This is done:

- In equation (12.232) and (12.233) for W_2 only, whereas W_1 is replaced by the width of the line.

- In equation (12.234) and (12.235) for W_2 only, whereas W_1 is replaced by the width of the line.

- In equation (12.236) for W_1 and W_2.

Another closed-form expression describing the non-ideal behaviour of a microstrip cross junction was published by [63]. Additionally there have been published papers [64, 65, 66] giving analytic (but not closed-form) expressions or just simple equivalent circuits with only a few expressions for certain topologies and dielectric constants which are actually of no pratical use.

12.9 Microstrip radial stub

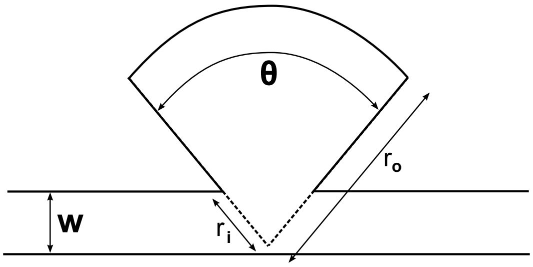


Figure 12.12: microstrip radial stub

Figure 12.12 shows a radial stub that is often used in microwave circuits to create an RF short. Its input impedance can be calculated as follows [67]:

$$Z_{in} = j \cdot \frac{Z_{F0} \cdot h}{r_i \theta \sqrt{\epsilon_{r,eff}}} \cdot \frac{J_0(k \cdot r_i) \cdot N_1(k \cdot r_o) - J_1(k \cdot r_o) \cdot N_0(k \cdot r_i)}{J_1(k \cdot r_i) \cdot N_1(k \cdot r_o) - J_1(k \cdot r_o) \cdot N_1(k \cdot r_i)} \tag{12.238}$$

with $Z_{F0} \approx 377\Omega$ being vacuum field impedance, h height of substrate, r_i inner radius, r_o outer radius, θ angle of stub in radians, $\epsilon_{r,eff}$ effective dielectric constant of a microstrip line with width $w = (r_i + r_o) \cdot \sin(0.5 \cdot \theta)$ and propagation constant $k = \alpha + j\beta$. J_0 and J_1 are the Bessel function of first kind and order zero and one, respectively. N_0 and N_1 are the Bessel function of second kind and order zero and one, respectively. It is further:

$$\alpha = \frac{R_s \cdot \sqrt{\epsilon_{r,eff}}}{Z_{F0} \cdot h} = \frac{\sqrt{\rho \pi f \mu \cdot \epsilon_{r,eff}}}{Z_{F0} \cdot h} \tag{12.239}$$

and

$$\beta = \frac{2\pi f \sqrt{\epsilon_{r,eff}}}{c_0} \tag{12.240}$$

with f frequency and c_0 light velocity in vacuum, ρ conductivity of the metal and μ magnetic permeability of metal. As the Bessel functions have complex arguments, it's better to split real and imaginary part:

$$J_0(k \cdot r) = J_0(\beta \cdot r) + j \cdot \alpha \cdot r \cdot J_1(\beta \cdot r) \tag{12.241}$$

$$J_1(k \cdot r) = J_1(\beta \cdot r) - j \cdot \alpha \cdot r \cdot J_0(\beta \cdot r) + j \cdot \frac{\alpha}{\beta} \cdot J_1(\beta \cdot r) \tag{12.242}$$

$$N_0(k \cdot r) = N_0(\beta \cdot r) + j \cdot \alpha \cdot r \cdot N_1(\beta \cdot r) \tag{12.243}$$

$$N_1(k \cdot r) = N_1(\beta \cdot r) - j \cdot \alpha \cdot r \cdot N_0(\beta \cdot r) + j \cdot \frac{\alpha}{\beta} \cdot N_1(\beta \cdot r) \tag{12.244}$$

12.10 Microstrip interdigital capacitor

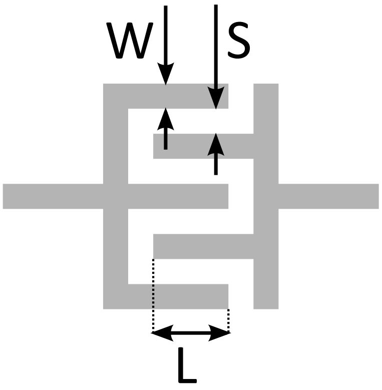


Figure 12.13: microstrip interdigital capacitor with 5 fingers

Figure 12.13 shows an interdigital capacitor (IDC). Its capacitance consists of three parts [68], the capacitance at the end of the fingers C_{end}, the one of a three-finger structure C_{3F} and the one of a periodical $N-3$ structure C_N:

$$C_{IDC} = C_{end} + C_{3F} + C_N \tag{12.245}$$

For an IDC with N fingers, the above-mentioned terms can be calculated as follows:

$$C_{end} = N \cdot C_{CPW,open} \tag{12.246}$$

$$C_{3F} = 4 \cdot L \cdot \epsilon_0 \cdot \left(\frac{K(k_1)}{K(k_1')} + \frac{\epsilon_r - 1}{2} \cdot \frac{K(k_2)}{K(k_2')} \right) \tag{12.247}$$

$$C_N = (N-3) \cdot L \cdot \epsilon_0 \cdot \left(\frac{K(k_3)}{K(k_3')} + \frac{\epsilon_r - 1}{2} \cdot \frac{K(k_4)}{K(k_4')} \right) \tag{12.248}$$

$$a = \frac{\pi}{4 \cdot h} \tag{12.249}$$

$$k_1 = \frac{W}{W + 2 \cdot S} \cdot \sqrt{\frac{(3W+2S)^2 - (W+2S)^2}{(3W+2S)^2 - W^2}} \tag{12.250}$$

$$k_2 = \frac{sinh(a \cdot W)}{sinh(a \cdot (W + 2 \cdot S))} \cdot \sqrt{\frac{sinh^2(a \cdot (3W+2S)) - sinh^2(a \cdot (W+2S))}{sinh^2(a \cdot (3W+2S)) - sinh^2(a \cdot W)}} \tag{12.251}$$

$$k_3 = \frac{W}{W+S} \tag{12.252}$$

$$k_4 = \frac{sinh(a \cdot W)}{sinh(a \cdot (W+S))} \cdot \sqrt{\frac{cosh^2(a \cdot (W+S)) - sinh^2(a \cdot (W+S))}{cosh^2(a \cdot W) - sinh^2(a \cdot (W+S))}} \tag{12.253}$$

$$\tag{12.254}$$

with $C_{CPW,open}$ being the open-end capacitance of a coplanar waveguide according to equation 13.28 and 13.29. $K(k)$ and $K(k')$ are the complete elliptic integral of the first kind and its complement.

A discrete capacitor is a very good equivalent circuit for an IDC at low frequencies. At higher frequencies the different transit times of the inner and the outer fingers creates an additional loss. This can be modelled by parallel transmission lines.

12.11 Microstrip via hole

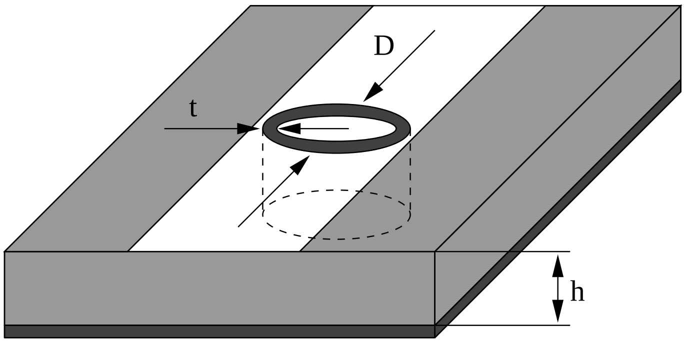


Figure 12.14: microstrip via hole to ground

According to Marc E. Goldfarb and Robert A. Pucel [69] a via hole ground in microstrip is a series of a resistance and an inductance. The given model for a cylindrical via hole has been verified numerically and experimentally for a range of $h < 0.03 \cdot \lambda_0$.

$$L = \frac{\mu_0}{2\pi} \cdot \left(h \cdot \ln\left(\frac{h + \sqrt{r^2 + h^2}}{r} \right) + \frac{3}{2} \cdot \left(r - \sqrt{r^2 + h^2} \right) \right) \tag{12.255}$$

whence h is the via length (substrate height) and $r = D/2$ the via's radius.

$$R = R\left(f = 0\right) \cdot \sqrt{1 + \frac{f}{f_\delta}} \tag{12.256}$$

with

$$f_\delta = \frac{\rho}{\pi \cdot \mu_0 \cdot t^2} \tag{12.257}$$

The relationship for the via resistance can be used as a close approximation and is valid independent of the ratio of the metalization thickness t to the skin depth. In the formula ρ denotes the specific resistance of the conductor material.

12.12 Bondwire

Wire inductors, so called bond wire connections, are used to connect active and passive circuit components as well as micro devices to the real world.

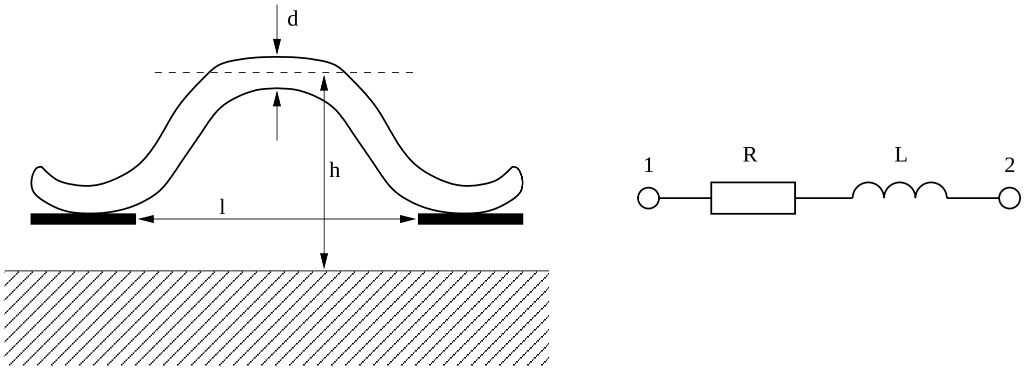


Figure 12.15: bond wire and its equivalent circuit

12.12.1 Freespace model

The freespace inductance L of a wire of diameter d and length l is given [70, 71] by

$$L = \frac{\mu_0}{2\pi} \cdot l \left[\ln \left\{ \frac{2l}{d} + \sqrt{1 + \left(\frac{2l}{d} \right)^2} \right\} + \frac{d}{2l} - \sqrt{1 + \left(\frac{d}{2l} \right)^2} + C \right] \tag{12.258}$$

where the frequency-dependent correction factor C is a function of bond wire diameter and its material skin depth δ is expressed as

$$C = \frac{\mu_r}{4} \cdot \tanh \left(\frac{4\delta}{d} \right) \tag{12.259}$$

$$\delta = \frac{1}{\sqrt{\pi \cdot \sigma \cdot f \cdot \mu_0 \cdot \mu_r}} \tag{12.260}$$

where σ is the conductivity of the wire material. When δ/d is small, $C = \delta/d$. Due to the skin effect the wire resistance R is given by

$$\begin{aligned} R &= \frac{l}{\sigma \cdot \pi \cdot \delta \cdot (d - \delta)} && \text{for} \quad d > 2 \cdot \delta && (12.261) \\ R &= \frac{l}{\sigma \cdot \pi \cdot 0.25 \cdot d^2} && \text{for} \quad d \leq 2 \cdot \delta && (12.262) \\ & && && (12.263) \end{aligned}$$

12.12.2 Mirror model

The effect of the ground plane on the inductance value of a wire has also been considered. If the wire is at a distance h above the ground plane, it sees its image at $2h$ from it. The wire and its image result in a mutual inductance. Since the image wire carries a current opposite to the current flow in the bond wire, the effective inductance of the bond wire becomes

$$L = \frac{\mu_0}{2\pi} \cdot l \left[\ln \left(\frac{4h}{d} \right) + \ln \left(\frac{l + \sqrt{l^2 + d^2/4}}{l + \sqrt{l^2 + 4h^2}} \right) + \sqrt{1 + \frac{4h^2}{l^2}} - \sqrt{1 + \frac{d^2}{4l^2}} - 2\frac{h}{l} + \frac{d}{2l} \right] \tag{12.264}$$

Mirror is a strange model that is frequency independent. Whereas computations are valid, hypothesis are arguable. Indeed, they did the assumption that the ground plane is perfect that is really a zero order model in the high frequency domain.

Chapter 13

Coplanar components

13.1 Coplanar waveguides (CPW)

13.1.1 Definition

A *coplanar line* is a structure in which all the conductors supporting wave propagation are located on the same plane, i.e. generally the top of a dielectric substrate. There exist two main types of coplanar lines: the first, called *coplanar waveguide (CPW)*, that we will study here, is composed of a median metallic strip separated by two narrow slits from a infinite ground plane, as may be seen on the figure below.

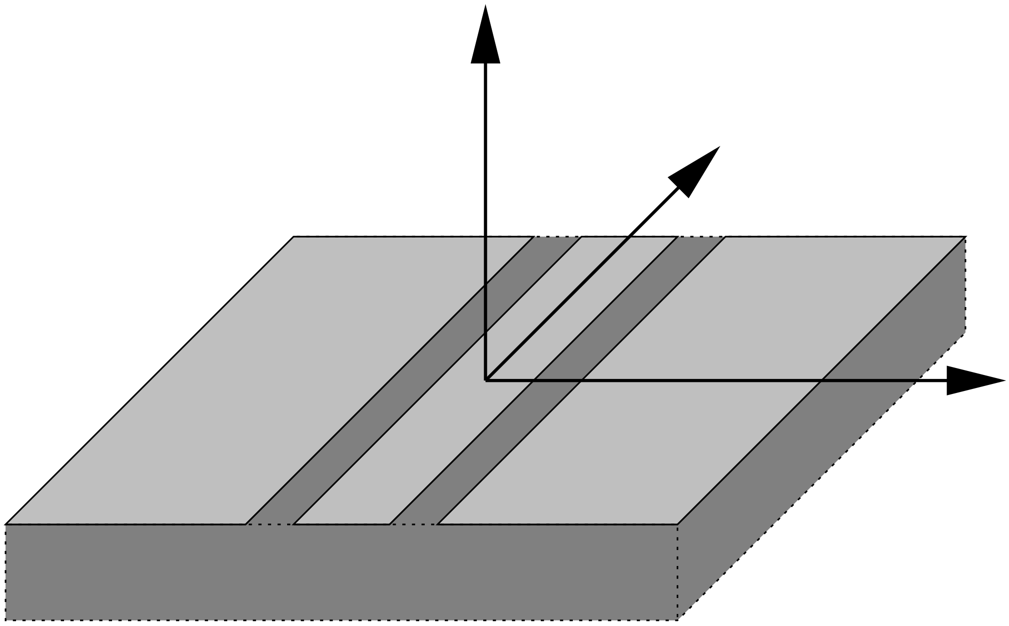
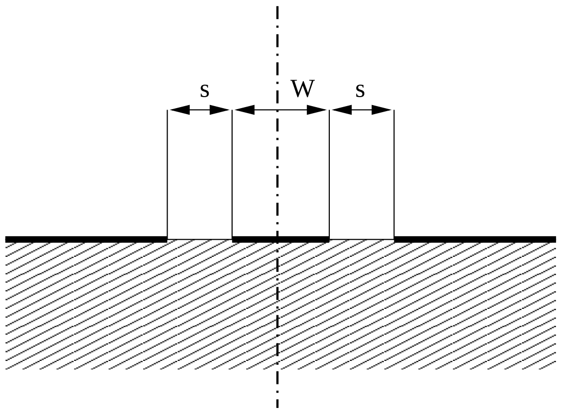


Figure 13.1: coplanar waveguide line

The characteristic dimensions of a CPW are the central strip width W and the width of the slots s. The structure is obviously symmetrical along a vertical plane running in the middle of the central strip.



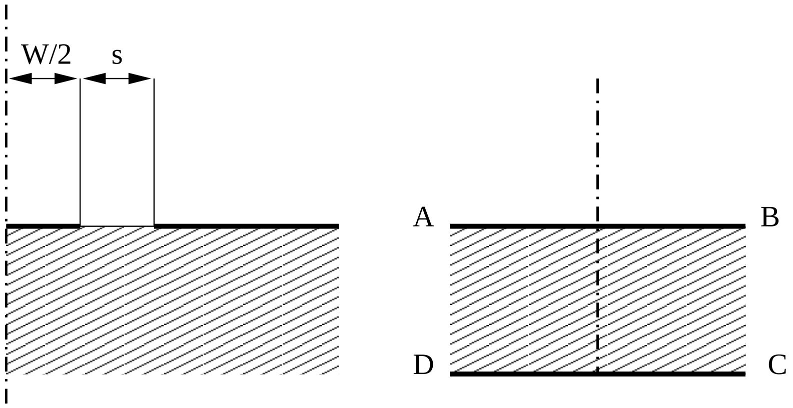
The other coplanar line, called a *coplanar slot (CPS)* is the complementary of that topology, consisting of two strips running side by side.

13.1.2 Quasi-static analysis by conformal mappings

A CPW can be quasi-statically analysed by the use of *conformal mappings.* Briefly speaking, it consists in transforming the geometry of the PCB into another conformation, whose properties make the computations straightforward. The interested reader can consult the pp. 886 - 910 of [72] which has a correct coverage of both the theoretical and applied methods. The French reader interested in the mathematical arcanes involved is referred to the second chapter of [73] (which may be out of print nowadays), for an extensive review of all the theoretical framework. The following analysis is mainly borrowed from [61], pp. 375 *et seq.* with additions from [72].

The CPW of negligible thickness located on top of an infinitely deep substrate, as shown on the left of the figure below, can be mapped into a parallel plate capacitor filled with dielectric $ABCD$ using the conformal function:

$$w = \int_{z_0}^{z} \frac{dz}{\sqrt{(z - W/2)(z - W/2 - s)}}. \tag{13.1}$$



To further simplify the analysis, the original dielectric boundary is assumed to constitute a magnetic wall, so that BC and AD become magnetic walls too and there is no resulting fringing field in the resulting capacitor. With that assumption, the capacitance per unit length is merely the sum of the top (air filled) and bottom (dielectric filled) partial capacitances. The latter is given by:

$$C_d = 2 \cdot \varepsilon_0 \cdot \varepsilon_r \cdot \frac{K(k_1)}{K'(k_1)} \tag{13.2}$$

while the former is:

$$C_a = 2 \cdot \varepsilon_0 \cdot \frac{K(k_1)}{K'(k_1)} \tag{13.3}$$

In both formulae $K(k)$ and $K'(k)$ represent the complete elliptic integral of the first kind and its complement, and $k_1 = \frac{W}{W+2s}$. While the separate evaluation of K and K' is more or less tricky, the K/K' ratio lets itself compute efficiently through the following formulae:

$$\frac{K(k)}{K'(k)} = \frac{\pi}{\ln\left(2\frac{1+\sqrt{k'}}{1-\sqrt{k'}}\right)} \qquad \text{for} \qquad 0 \leq k \leq \frac{1}{\sqrt{2}} \tag{13.4}$$

$$\frac{K(k)}{K'(k)} = \frac{\ln\left(2\frac{1+\sqrt{k}}{1-\sqrt{k}}\right)}{\pi} \qquad \text{for} \qquad \frac{1}{\sqrt{2}} \leq k \leq 1 \tag{13.5}$$

with k' being the complementary modulus: $k' = \sqrt{1-k^2}$. While [72] states that the accuracy of the above formulae is close to 10^{-5}, [61] claims it to be $3 \cdot 10^{-6}$. It can be considered as exact for any practical purposes.

The total line capacitance is thus the sum of C_d and C_a. The effective permittivity is therefore:

$$\varepsilon_{re} = \frac{\varepsilon_r + 1}{2} \tag{13.6}$$

and the impedance:

$$Z = \frac{0.25 \cdot Z_{F0}}{\sqrt{\varepsilon_{re}}} \cdot \frac{K'(k_1)}{K(k_1)} \tag{13.7}$$

with Z_{F0} being the field impedance of vacuum, i.e. $\approx 377\Omega$.

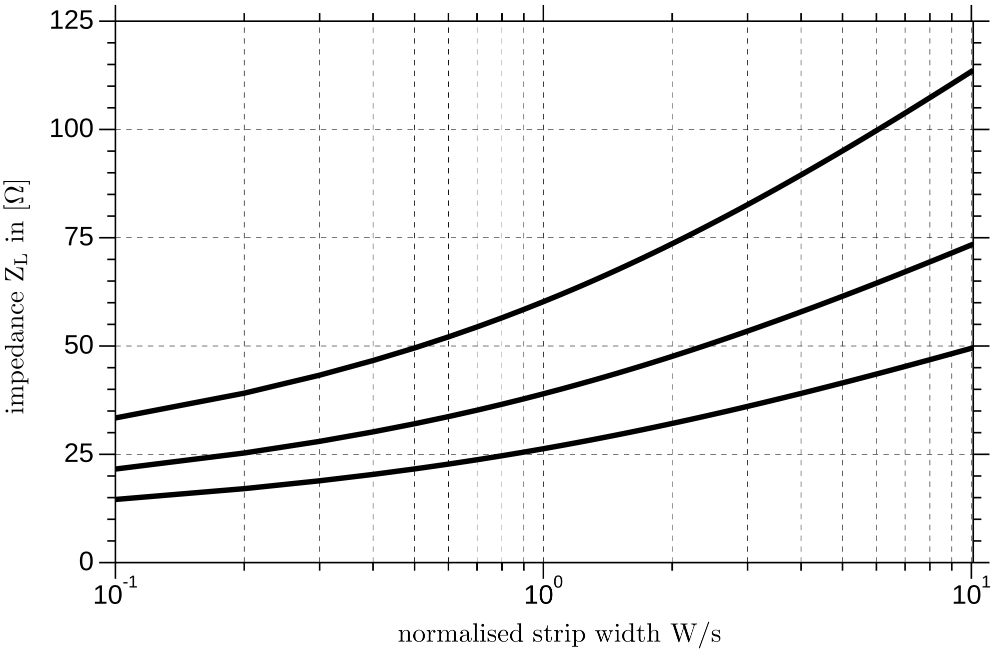


Figure 13.2: characteristic impedance as approximated by eq. (13.7) for $\varepsilon_r = 1.0$ (air), 3.78 (quartz) and 9.5 (alumina)

In practical cases, the substrate has a finite thickness h. To carry out the analysis of this conformation, a preliminary conformal mapping transforms the finite thickness dielectric into an infinite thickness one. Only the effective permittivity is altered; it becomes:

$$\varepsilon_{re} = 1 + \frac{\varepsilon_r - 1}{2} \cdot \frac{K(k_2)}{K'(k_2)} \cdot \frac{K'(k_1)}{K(k_1)} \tag{13.8}$$

where k_1 is given above and

$$k_2 = \frac{\sinh\left(\dfrac{\pi W}{4h}\right)}{\sinh\left(\dfrac{\pi \cdot (W + 2s)}{4h}\right)}. \tag{13.9}$$

Finally, let us consider a CPW over a finite thickness dielectric backed by an infinite ground plane. In this case, the quasi-TEM wave is an hybrid between microstrip and true CPW mode. The equations then become:

$$\varepsilon_{re} = 1 + q \cdot (\varepsilon_r - 1) \tag{13.10}$$

where q, called *filling factor* is given by:

$$q = \frac{\dfrac{K(k_3)}{K'(k_3)}}{\dfrac{K(k_1)}{K'(k_1)} + \dfrac{K(k_3)}{K'(k_3)}} \tag{13.11}$$

and

$$k_3 = \frac{\tanh\left(\dfrac{\pi W}{4h}\right)}{\tanh\left(\dfrac{\pi \cdot (W + 2s)}{4h}\right)} \tag{13.12}$$

The impedance of this line amounts to:

$$Z = \frac{0.5 \cdot Z_{F0}}{\sqrt{\varepsilon_{re}}} \cdot \frac{1}{\dfrac{K(k_1)}{K'(k_1)} + \dfrac{K(k_3)}{K'(k_3)}} \tag{13.13}$$

13.1.3 Effects of metalization thickness

In most practical cases, the strips are very thin, yet their thickness cannot be entirely neglected. A first order correction to take into account the non-zero thickness of the conductor is given by [61]:

$$s_e = s - \Delta \tag{13.14}$$

and

$$W_e = W + \Delta \tag{13.15}$$

where

$$\Delta = \frac{1.25t}{\pi} \cdot \left(1 + \ln\left(\frac{4\pi W}{t}\right)\right) \tag{13.16}$$

In the computation of the impedance, both the k_1 and the effective dielectric constant are affected, wherefore k_1 must be substituted by an "effective" modulus k_e, with:

$$k_e = \frac{W_e}{W_e + 2s_e} \approx k_1 + \left(1 - k_1^2\right) \cdot \frac{\Delta}{2s} \tag{13.17}$$

and

$$\varepsilon_{re}^t = \varepsilon_{re} - \frac{0.7 \cdot (\varepsilon_{re} - 1) \cdot \dfrac{t}{s}}{\dfrac{K(k_1)}{K'(k_1)} + 0.7 \cdot \dfrac{t}{s}} \tag{13.18}$$

13.1.4 Effects of dispersion

The effects of dispersion in CPW are similar to those encountered in the microstrip lines, though the net effect on impedance is somewhat different. [61] gives a closed form expression to compute $\varepsilon_{re}(f)$ from its quasi-static value:

$$\sqrt{\varepsilon_{re}(f)} = \sqrt{\varepsilon_{re}(0)} + \frac{\sqrt{\varepsilon_r} - \sqrt{\varepsilon_{re}(0)}}{1 + G \cdot \left(\dfrac{f}{f_{TE}}\right)^{-1.8}} \tag{13.19}$$

where:

$$G = e^{u \cdot \ln\left(\frac{W}{s}\right) + v} \tag{13.20}$$

$$u = 0.54 - 0.64p + 0.015p^2 \tag{13.21}$$

$$v = 0.43 - 0.86p + 0.54p^2 \tag{13.22}$$

$$p = \ln\left(\tfrac{W}{h}\right) \tag{13.23}$$

and f_{TE} is the cut-off frequency of the TE_0 mode, defined by:

$$f_{TE} = \frac{c}{4h \cdot \sqrt{\varepsilon_r - 1}}. \tag{13.24}$$

This dispersion expression was first reported by [74] and has been reused and extended in [75]. The accuracy of this expression is claimed to be better than 5% for $0.1 \le W/h \le 5$, $0.1 \le W/s \le 5$, $1.5 \le \varepsilon_r \le 50$ and $0 \le f/f_{TE} \le 10$.

13.1.5 Evaluation of losses

As for microstrip lines, the losses in CPW results of at least two factors: a dielectric loss α_d and conductor losses α_c^{CW}. The dielectric loss α_d is identical to the microstrip case, see eq. (12.80) on page 180.

The α_c^{CW} part of the losses is more complex to evaluate. As a general rule, it might be written:

$$\alpha_c^{CW} = 0.023 \cdot \frac{R_s}{Z_{0cp}} \left[\frac{\partial Z_{0cp}^a}{\partial s} - \frac{\partial Z_{0cp}^a}{\partial W} - \frac{\partial Z_{0cp}^a}{\partial t} \right] \text{ in dB/unit length} \tag{13.25}$$

where Z_{0cp}^a stands for the impedance of the coplanar waveguide with air as dielectric and R_s is the surface resistivity of the conductors (see eq. (12.82) on page 181).

Through a direct approach evaluating the losses by conformal mapping of the current density, one obtains [61], first reported in [76] and finally applied to coplanar lines by [77]:

$$\begin{aligned} \alpha_c^{CW} &= \frac{R_s \cdot \sqrt{\varepsilon_{re}}}{480\pi \cdot K(k_1) \cdot K'(k_1) \cdot (1 - k_1^2)} \cdot \\ &\left(\frac{1}{a} \left[\pi + \ln \frac{8\pi a \cdot (1 - k_1)}{t \cdot (1 + k_1)} \right] + \frac{1}{b} \left[\pi + \ln \frac{8\pi b \cdot (1 - k_1)}{t \cdot (1 + k_1)} \right] \right) \end{aligned} \tag{13.26}$$

In the formula above, $a = W/2, b = s + W/2$ and it is assumed that $t > 3\delta, t \ll W$ and $t \ll s$.

13.1.6 S- and Y-parameters of the single coplanar line

The computation of the coplanar waveguide lines S- and Y-parameters is equal to all transmission lines (see section section 10.23 on page 119).

13.2 Coplanar waveguide open

The behaviour of an open circuit as shown in fig. 13.3 is very similar to that in a microstrip line; that is, the open circuit is capacitive in nature.

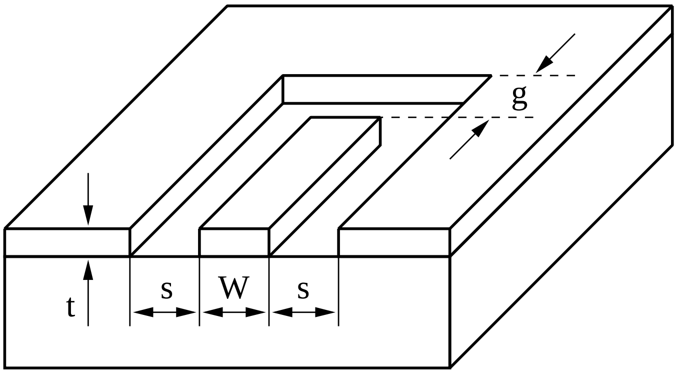


Figure 13.3: coplanar waveguide open-circuit

A very simple approximation for the equivalent length extension Δl associated with the fringing fields has been given by K.Beilenhoff [78].

$$\Delta l_{open} = \frac{C_{open}}{C'} \approx \frac{W + 2s}{4} \tag{13.27}$$

For the open end, the value of Δl is not influenced significantly by the metallization thickness and the gap g when $g > W + 2s$. Also, the effect of frequency and aspect ration $W/(W + 2s)$ is relatively weak. The above approximation is valid for $0.2 \leq W/(W + 2s) \leq 0.8$.

The open end capacitance C_{open} can be written in terms of the capacitance per unit length and the wave resistance.

$$C_{open} = C' \cdot \Delta l_{open} = \frac{\sqrt{\varepsilon_{r,eff}}}{c_0 \cdot Z_L} \cdot \Delta l_{open} \tag{13.28}$$

In order to also model a small end gap g, the above-mentioned capacitance is multiplied by the following factor:

$$k = 1 + 0.03 \cdot \frac{W + 2s}{g} \tag{13.29}$$

13.3 Coplanar waveguide short

There is a similar simple approximation for a coplanar waveguide short-circuit, also given in [78]. The short circuit is inductive in nature.

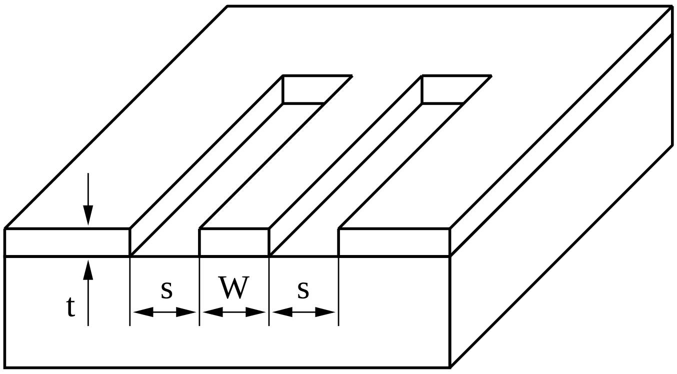


Figure 13.4: coplanar waveguide short-circuit

The equivalent length extension Δl associated with the fringing fields is

$$\Delta l_{short} = \frac{L_{short}}{L'} \approx \frac{W + 2s}{8} \tag{13.30}$$

Equation (13.30) is valid when the metalization thickness t does not become too large ($t < s/3$).

The short end inductance L_{short} can be written in terms of the inductance per unit length and the wave resistance.

$$L_{short} = L' \cdot \Delta l_{short} = \frac{\sqrt{\varepsilon_{r,eff}} \cdot Z_L}{c_0} \cdot \Delta l_{short} \tag{13.31}$$

According to W.J.Getsinger [79] the CPW short-circuit inductance per unit length can also be modeled by

$$L_{short} = \frac{2}{\pi} \cdot \varepsilon_0 \cdot \varepsilon_{r,eff} \cdot (W + s) \cdot Z_L^2 \cdot \left(1 - \operatorname{sech}\left(\frac{\pi \cdot Z_{F0}}{2 \cdot Z_L \cdot \sqrt{\varepsilon_{r,eff}}}\right)\right) \tag{13.32}$$

based on his duality [80] theory.

13.4 Coplanar waveguide gap

According to W.J.Getsinger [80] a coplanar series gap (see fig. 13.5) is supposed to be the dual problem of the inductance of a connecting strip between twin strip lines.

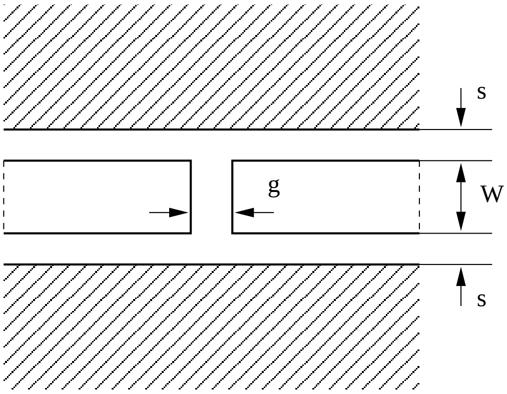


Figure 13.5: coplanar waveguide series gap

The inductance of such a thin strip with a width g and the length W is given to a good approximation by

$$L = \frac{\mu_0 \cdot W}{2\pi} \cdot \left(p - \sqrt{1+p^2} + \ln\left(\frac{1+\sqrt{1+p^2}}{p}\right)\right) \tag{13.33}$$

where $p = g/4W$ and $g, W \ll \lambda$. Substituting this inductance by its equivalent capacitance of the gap in CPW yields

$$\begin{aligned} C &= L \cdot \frac{4 \cdot \varepsilon_{r,eff}}{Z_{F0}^2} \\ &= \frac{2 \cdot \varepsilon_0 \cdot \varepsilon_{r,eff} \cdot W}{\pi} \cdot \left(p - \sqrt{1+p^2} + \ln\left(\frac{1+\sqrt{1+p^2}}{p}\right)\right) \end{aligned} \tag{13.34}$$

13.5 Coplanar waveguide step

The coplanar step discontinuity shown in figure 13.6 has been analysed by C. Sinclair [81].

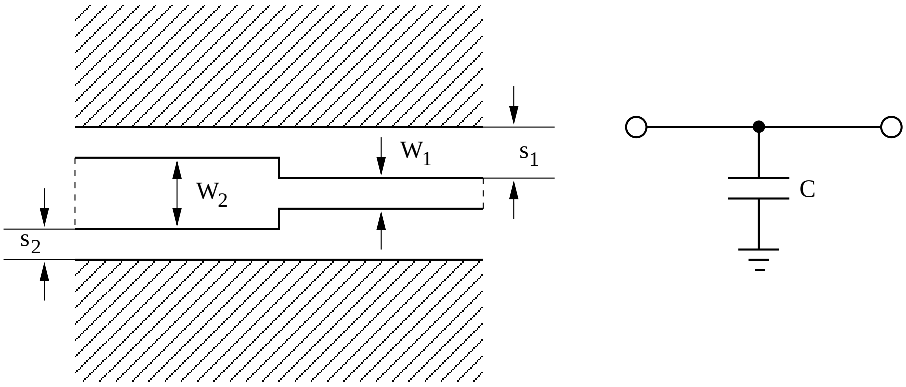


Figure 13.6: coplanar waveguide impedance step and equivalent circuit

The symmetric step change in width of the centre conductor is considered to have a similar equivalent circuit as a step of a parallel plate guide - this is a reasonable approximation to the CPW step as in the CPW the majority of the field is between the inner and outer conductors with some fringing.

The actual CPW capacitance can be expressed as

$$C = \overline{x} \cdot \frac{\varepsilon_0}{\pi} \cdot \left(\frac{\alpha^2 + 1}{\alpha} \cdot \ln \left(\frac{1+\alpha}{1-\alpha} \right) - 2 \cdot \ln \left(\frac{4 \cdot \alpha}{1 - \alpha^2} \right) \right) \tag{13.35}$$

where

$$\alpha = \frac{s_1}{s_2}, \alpha < 1 \quad \text{and} \quad \overline{x} = \frac{x_1 + x_2}{2} \tag{13.36}$$

The capacitance per unit length equivalence yields

$$x_1 = \frac{C'\left(W_1, s_1\right) \cdot s_1}{\varepsilon_0} \quad \text{and} \quad x_2 = \frac{C'\left(W_2, s_2\right) \cdot s_2}{\varepsilon_0} \tag{13.37}$$

with

$$C' = \frac{\sqrt{\varepsilon_{r,eff}}}{c_0 \cdot Z_L} \tag{13.38}$$

The average equivalent width $\overline{x}$ of the parallel plate guide can be adjusted with an expression that uses weighted average of the gaps s_1 and s_2. The final expression has not been discussed in [81]. The given equations are validated over the following ranges: $2 < \varepsilon_r < 14$, $h > W + 2s$ and $f < 40\text{GHz}$.

The Z-parameters of the equivalent circuit depicted in fig. 13.6 are

$$Z_{11} = Z_{21} = Z_{12} = Z_{22} = \frac{1}{j\omega C} \tag{13.39}$$

The MNA matrix representation for the AC analysis can be derived from the Z-parameters in the following way.

$$\begin{bmatrix} . & . & 1 & 0 \\ . & . & 0 & 1 \\ -1 & 0 & Z_{11} & Z_{12} \\ 0 & -1 & Z_{21} & Z_{22} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ I_{in} \\ I_{out} \end{bmatrix} = \begin{bmatrix} I_1 \\ I_2 \\ 0 \\ 0 \end{bmatrix} \tag{13.40}$$

The above expanded representation using the Z-parameters is necessary because the Y-parameters are infinite. During DC analysis the equivalent circuit is a voltage source between both terminals with zero voltage.

The S-parameters of the topology are

$$S_{11} = S_{22} = -\frac{Z_0}{2Z + Z_0} \tag{13.41}$$

$$S_{12} = S_{21} = 1 + S_{11} = \frac{2Z}{2Z + Z_0} \tag{13.42}$$

Chapter 14

Stripline components

14.1 Transmission line

A symmetrical stripline (also known as tri-plate line or sandwich line) can be seen in figure 14.1 on the left-handside. The signal line is centered between the upper and lower ground plane.

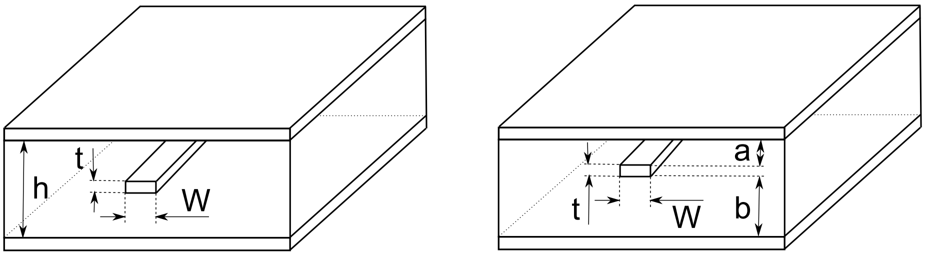


Figure 14.1: symmetrical stripline (left) and unsymmetrical stripline (right)

As the line is completely embedded into the dielectric substrate it is non-dispersive [60].

$$\epsilon_{r,eff} = \epsilon_r \tag{14.1}$$

$$Z_L = \frac{0.25 \cdot Z_{F0} \cdot (h-t)}{\sqrt{\epsilon_r} \cdot \left(W + \frac{1}{\pi} \cdot \left(2 \cdot h \cdot \ln\left(\frac{2 \cdot h - t}{h-t}\right) - t \cdot \ln\left(\frac{h^2}{(h-t)^2} - 1\right)\right)\right)} \qquad \text{for} \quad \frac{W}{h-t} \geq 0.35 \tag{14.2}$$

$$Z_L = \frac{Z_{F0}}{2 \cdot \pi \sqrt{\epsilon_r}} \cdot \ln\left(\frac{4 \cdot h}{\pi \cdot d_e}\right) \qquad \text{for} \quad \frac{W}{h-t} < 0.35 \tag{14.3}$$

$$\text{with} \qquad d_e = \frac{W}{2} \cdot \left(1 + \frac{t}{\pi \cdot W} \cdot \left(1 + \ln\left(\frac{4 \cdot \pi \cdot W}{t}\right)\right) + 0.236 \cdot \left(\frac{t}{W}\right)^{1.65}\right) \tag{14.4}$$

If $t/W > 1$ then in equation 14.4 t/W has to be replaced with W/t and d_e/W with d_e/t. The conductor losses are as follows [82].
For $\frac{W}{h-t} \geq 0.35$ it is:

$$\alpha_c = \frac{4 \cdot \epsilon_r \cdot Z_L \cdot R_{sh}}{Z_{F0}^2 \cdot (h-t)} \cdot \left(1 + \frac{2 \cdot W}{h-t} + \frac{h+t}{\pi \cdot (h-t)} \cdot \ln\left(\frac{2h-t}{t}\right)\right) \tag{14.5}$$

and for $\frac{W}{h-t} < 0.35$ it is:

$$\begin{aligned}\alpha_c =& \frac{R_{sh}}{2\pi \cdot Z_L \cdot d_e} \\ & \cdot \left(\frac{1}{2} + \frac{d_e}{h} + \frac{t}{2\pi \cdot W} + \frac{\ln\left(\frac{4\pi \cdot W}{t}\right)}{2\pi} + 0.1947 \cdot \left(\frac{t}{W}\right)^{0.65} - 0.0767 \cdot \left(\frac{t}{W}\right)^{1.65}\right)\end{aligned} \tag{14.6}$$

with R_{sh} being the sheet resistivity according to equation 12.82. The dielectric losses are:

$$\alpha_d = \frac{\pi \cdot f \cdot \sqrt{\epsilon_{r,eff}}}{c_0} \cdot \tan\delta \tag{14.7}$$

A non-symmetrical stripline is shown in figure 14.1 on the right-handside. Its properties can be calculated by the equations of the symmetrical stripline. The effective permittivity $\epsilon_{r,eff}$ and the dielectric loss factor α_d are the same. The characteristic impedance is calculated by the virtual parallel connection of two striplines with height h_1 and h_2, respectively.

$$Z_L = 2 \cdot \frac{Z_L(h_1 = 2a + t) \cdot Z_L(h_2 = 2b + t)}{Z_L(h_1 = 2a + t) + Z_L(h_2 = 2b + t)} \tag{14.8}$$

The conductor loss is the arithmetic mean of both factors.

$$\alpha_c = 0.5 \cdot (\alpha_c(h_1 = 2a + t) + \alpha_c(h_2 = 2b + t)) \tag{14.9}$$

Higher order modes occur in striplines if the half wavelength becomes as large as the line width w or the substrate height h. The cut-off frequencies for the first TE and TM modes are [82]:

$$f_{c,TE} = \frac{c_0}{(2 \cdot w + 0.5 \cdot \pi \cdot h) \cdot \sqrt{\epsilon_r}} \tag{14.10}$$

$$f_{c,TM} = \frac{c_0}{2 \cdot h \cdot \sqrt{\epsilon_r}} \tag{14.11}$$

14.2 Stripline open end

The open end effect is modeled with an additional line length Δl [83], [82]:

$$\Delta l = \frac{1}{k} \cdot \arctan\left(\frac{(L + 2 \cdot W) \cdot \tan(k \cdot L)}{4 \cdot L + 2 \cdot W}\right) \tag{14.12}$$

with

$$k = \frac{2 \cdot \pi \cdot f \cdot \sqrt{\epsilon_r}}{c_0} \tag{14.13}$$

$$L = \frac{h \cdot \ln(2)}{\pi} \tag{14.14}$$

with c_0 vacuum light velocity, W is line width, h substrate height, f frequency and ϵ_r relative permittivity of substrate.

14.3 Stripline gap

A gap of length s in the transmission line is modeled with a π circuit consisting of capacitances as shown in picture 12.5 [83], [82]:

$$\omega \cdot C_s = \frac{0.5}{Z_L} \cdot \frac{1 + B \cdot A}{A - B} - 0.5 \cdot \omega C_p \tag{14.15}$$

$$\omega \cdot C_p = \frac{1}{Z_L} \cdot \tan\left(\frac{2 \cdot \ln(2) \cdot h \cdot f \cdot \sqrt{\varepsilon_r}}{c_0}\right) \tag{14.16}$$

with

$$A = \cot\left(\frac{\pi \cdot s \cdot f \cdot \sqrt{\epsilon_r}}{c_0}\right) \tag{14.17}$$

$$B = \frac{2 \cdot h \cdot f \cdot \sqrt{\epsilon_r}}{c_0} \cdot \left[\ln\left(\coth\left(\frac{\pi \cdot s}{2 \cdot h}\right)\right) - \ln\left(\cosh\left(\frac{\pi \cdot s}{2 \cdot h}\right)\right)\right] \tag{14.18}$$

14.4 Stripline bend

The (uncompensated) stripline bend with angle θ (in radians) can be modeled with a T circuit consisting of two identical series inductances L_s and a parallel capacitance C_p in between [83]:

$$\omega \cdot L_s = \frac{2 \cdot D \cdot Z_L \cdot f \cdot \sqrt{\epsilon_r}}{c_0} \cdot \left[\Psi\left(\frac{\theta}{2\pi} - 0.5\right) - \Psi(-0.5)\right] \tag{14.19}$$

$$\omega \cdot C_p = \frac{2\pi \cdot D \cdot f \cdot \sqrt{\epsilon_r} \cdot \tan(0.5 \cdot \theta)}{c_0 \cdot Z_L} \tag{14.20}$$

with effective line width

$$D = W + h \cdot \frac{2 \cdot \ln(2)}{\pi} \tag{14.21}$$

with c_0 vacuum light velocity, W is line width, h substrate height, f frequency and ϵ_r relative permittivity of substrate. $\Psi(x)$ is the so-called digamma function. Here it can be precisely approximated by the following equation:

$$\Psi(x) = 0.144202 + 0.023591 \cdot (x + 1) + 0.870648 \cdot \ln(x + 1) - \frac{0.744174}{x + 1} - \frac{1}{x} \tag{14.22}$$

14.5 Optimal stripline bend

The parasitics of a 90 degree stripline bend can be compensated completely by cutting off the corner metallization. The model of this mitered bend is just the line length l of its curve [83]:

$$\begin{aligned} \frac{W}{l} =& 1.7854346 - 0.015055579 \cdot \frac{W}{h} - 0.11334006 \cdot \left(\frac{W}{h}\right)^2 \\ &+ 0.050432954 \cdot \left(\frac{W}{h}\right)^3 - 0.0097883487 \cdot \left(\frac{W}{h}\right)^4 \end{aligned} \tag{14.23}$$

14.6 Stripline step

A stripline impedance step (step in line width, i.e. $W_1 > W_2$) creates a series inductance L_s [83], [82]:

$$\omega \cdot L_s = -\frac{2 \cdot Z_L \cdot D_1 \cdot f \cdot \sqrt{\epsilon_r}}{c_0} \cdot \ln\left(\sin\left(\frac{\pi \cdot D_2}{2 \cdot D_1}\right)\right) \tag{14.24}$$

with D_1 and D_2 being the effective line width according to equation 14.21. The parallel capacitance at line W_2 is mentioned nowhere, but it can be approximated by the geometry of the corners:

$$C_p = \varepsilon_0 \cdot \varepsilon_r \cdot \frac{(W_1 - W_2)^2}{h} \tag{14.25}$$

14.7 Stripline Tee junction

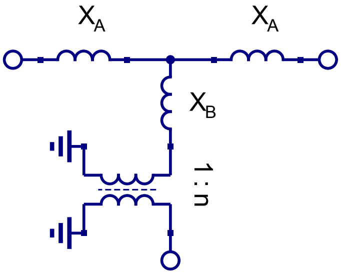


Figure 14.2: equivalent circuit of a stripline tee junction

The equivalent circuit of a stripline tee junction is depicted in figure 14.2. The component values are [83], [82]:

$$n = \frac{\lambda}{\pi \cdot D_2} \cdot \sin\left(\frac{\pi \cdot D_2}{\lambda}\right) \tag{14.26}$$

$$X_A = -\frac{(\pi \cdot n \cdot D_2)^2}{16 \cdot \lambda \cdot D_1} \tag{14.27}$$

for $D_2 < 0.5 \cdot D_1$ it is

$$\begin{aligned} X_B = &- 0.5 \cdot X_C + \frac{2 \cdot D_1^2}{\lambda \cdot n^2 \cdot D_2} \cdot \left[0.5 \cdot \left(\frac{D_1}{\lambda}\right)^2 \cdot \cos^4\left(\frac{\pi \cdot D_2}{2 \cdot D_1}\right)\right. \\ &\left. - \ln\left(\sin\left(\frac{\pi \cdot D_2}{2 \cdot D_1}\right)\right) + \frac{\pi \cdot D_2}{6 \cdot D_1} + 1.5 \cdot \left(\frac{D_1}{\lambda}\right)^2 + \ln(2)\right] \end{aligned} \tag{14.28}$$

and for $D_2 > 0.5 \cdot D_1$ it is

$$X_B = -0.5 \cdot X_C + \frac{2 \cdot D_1^2}{\lambda \cdot n^2 \cdot D_2} \cdot \left(\ln\left(1.43 \cdot \frac{D_1}{D_2}\right) + 2 \cdot \left(\frac{D_1}{\lambda}\right)^2\right) \tag{14.29}$$

with

$$\lambda = \frac{c_0}{f \cdot \sqrt{\epsilon_r}} \tag{14.30}$$

with D_1 and D_2 being the effective line width according to 14.21.

Chapter 15

Other types of transmission lines

The dielectric losses of an arbitrary quasi-TEM waveguide can be calculated as follows [84]:

$$\alpha_D \left[\frac{\text{dB}}{\text{m}}\right] = \frac{20 \cdot \pi}{\lambda \cdot \ln 10} \cdot \frac{\iiint \vec{E} \cdot \vec{D} \cdot \tan\delta \; dV}{C \cdot V^2} \tag{15.1}$$

$$= \frac{20 \cdot \pi}{\ln 10} \cdot \frac{f \cdot \sqrt{\epsilon_{r,eff}} \cdot \tan\delta}{c_0} \cdot \frac{\epsilon_r}{\epsilon_{r,eff}} \cdot \frac{\partial \epsilon_{r,eff}}{\partial \epsilon_r} \tag{15.2}$$

The conductor losses can be calculated by the Wheeler's formula:

$$\alpha_c \left[\frac{\text{dB}}{\text{m}}\right] = \frac{R_s \cdot \sqrt{\epsilon_r}}{2 \cdot Z_{F0}} \cdot \frac{1}{Z_L} \cdot \frac{\partial Z_L}{\partial n} \tag{15.3}$$

where R_{sh} is the sheet resistance according to 12.82 and n is the surface normal into the conductor. I.e. the conductor volume is reduced and its change to the characteristic transmission line impedance Z_L gives the loss.

15.1 Coaxial cable

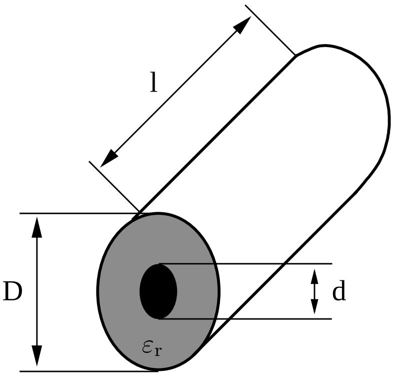


Figure 15.1: coaxial line

15.1.1 Characteristic impedance

The characteristic impedance of a coaxial line can be calculated as follows [85]:

$$Z_L = \frac{Z_{F0}}{2\pi \cdot \sqrt{\varepsilon_r}} \cdot \ln\left(\frac{D}{d}\right) \tag{15.4}$$

15.1.2 Losses

Overall losses in a coaxial cable consist of dielectric and conductor losses. The dielectric losses compute as follows:

$$\alpha_d = \frac{\pi}{c_0} \cdot f \cdot \sqrt{\varepsilon_r} \cdot \tan\delta \tag{15.5}$$

The conductor (i.e. ohmic) losses are specified by

$$\alpha_c = \sqrt{\varepsilon_r} \cdot \left(\frac{\dfrac{1}{D} + \dfrac{1}{d}}{\ln\left(\dfrac{D}{d}\right)} \right) \cdot \frac{R_S}{Z_{F0}} \tag{15.6}$$

with R_S denoting the sheet resistance of the conductor material, i.e. the skin resistance

$$R_S = \sqrt{\pi \cdot f \cdot \mu_r \cdot \mu_o \cdot \rho} \tag{15.7}$$

15.1.3 Cutoff frequencies

In normal operation a signal wave passes through the coaxial line as a TEM wave with no electrical or magnetic field component in the direction of propagation. Beyond a certain cutoff frequency additional (unwanted) higher order modes are excited.

$$f_{TE} \approx \frac{c_0}{\pi \cdot (D+d)} \quad \rightarrow \quad \text{TE(1,1) mode} \tag{15.8}$$

$$f_{TM} \approx \frac{c_0}{2 \cdot (D-d)} \quad \rightarrow \quad \text{TM(n,1) mode} \tag{15.9}$$

15.2 Twisted pair

The twisted pair configurations as shown in fig. 15.2 provides good low frequency shielding. Undesired signals tend to be coupled equally into eachline of the pair. A differential receiver will therefore completely cancel the interference.

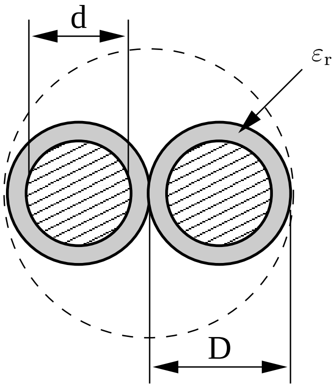


Figure 15.2: twisted pair configuration

15.2.1 Quasi-static model

According to P. Lefferson [86] the characteristic impedance and effective dielectric constant of a twisted pair can be calculated as follows.

$$Z_L = \frac{Z_{F0}}{\pi \cdot \sqrt{\varepsilon_{r,eff}}} \cdot \text{acosh}\left(\frac{D}{d}\right) \tag{15.10}$$

$$\varepsilon_{r,eff} = \varepsilon_{r,1} + q \cdot (\varepsilon_r - \varepsilon_{r,1}) \tag{15.11}$$

with

$$q = 0.25 + 0.0004 \cdot \theta^2 \qquad \text{and} \qquad \theta = \text{atan}\,(T \cdot \pi \cdot D) \tag{15.12}$$

whereas θ is the pitch angle of the twist; the angle between the twisted pair's center line and the twist. It was found to be optimal for θ to be between 20°and 45°. T denotes the twists per length. Eq. (15.12) is valid for film insulations, for the softer PTFE material it should be modified as follows.

$$q = 0.25 + 0.001 \cdot \theta^2 \tag{15.13}$$

Assuming air as dielectric around the wires yields 1's replacing $\varepsilon_{r,1}$ in eq. (15.11). The wire's total length before twisting in terms of the number of turns N is

$$l = N \cdot \pi \cdot D \cdot \sqrt{1 + \frac{1}{\tan^2 \theta}} \tag{15.14}$$

15.2.2 Transmission losses

The propagation constant γ of a general transmission line is given by

$$\gamma = \sqrt{(R' + j\omega L') \cdot (G' + j\omega C')} \tag{15.15}$$

Using some transformations of the formula gives an expression with and without the angular frequency.

$$\begin{aligned}\gamma &= \sqrt{(R' + j\omega L') \cdot (G' + j\omega C')} \\ &= \sqrt{L'C'} \cdot \sqrt{\frac{R'G'}{L'C'} + j\omega\left(\frac{R'}{L'} + \frac{G'}{C'}\right) - \omega^2} \\ &= \sqrt{L'C'} \cdot \sqrt{\left(\frac{1}{2} \cdot \left(\frac{R'}{L'} + \frac{G'}{C'}\right) + j\omega\right)^2 - \frac{1}{4} \cdot \left(\frac{R'}{L'} + \frac{G'}{C'}\right)^2 + \frac{R'G'}{L'C'}}\end{aligned} \tag{15.16}$$

For high frequencies eq.(15.16) can be approximated to

$$\gamma \approx \sqrt{L'C'} \cdot \left(\frac{1}{2} \cdot \left(\frac{R'}{L'} + \frac{G'}{C'}\right) + j\omega\right) \tag{15.17}$$

Thus the real part of the propagation constant γ yields

$$\alpha = Re\left\{\gamma\right\} = \sqrt{L'C'} \cdot \frac{1}{2} \cdot \left(\frac{R'}{L'} + \frac{G'}{C'}\right) \tag{15.18}$$

With

$$Z_L = \sqrt{\frac{L'}{C'}} \tag{15.19}$$

the expression in eq.(15.18) can be written as

$$\alpha = \alpha_c + \alpha_d = \frac{1}{2} \cdot \left(\frac{R'}{Z_L} + G' Z_L\right) \tag{15.20}$$

whereas α_c denotes the conductor losses and α_d the dielectric losses.

Conductor losses

The sheet resistance R' of a transmission line conductor is given by

$$R' = \frac{\rho}{A_{eff}} \tag{15.21}$$

whereas ρ is the specific resistance of the conductor material and A_{eff} the effective area of the conductor perpendicular to the propagation direction. At higher frequencies the area of the conductor is reduced by the skin effect. The skin depth is given by

$$\delta_s = \sqrt{\frac{\rho}{\pi \cdot f \cdot \mu}} \tag{15.22}$$

Thus the effective area of a single round wire yields

$$A_{eff} = \pi \cdot \left(r^2 - (r - \delta_s)^2\right) = \pi \cdot \delta_s \cdot (d - \delta_s) \tag{15.23}$$

whereas r denotes the radius of the wire. This means the overall conductor attenuation constant α_c for a single wire gives

$$\alpha_c = \frac{R'}{2 \cdot Z_L} = \frac{\rho}{2 \cdot Z_L \cdot \pi \cdot \delta_s \cdot (d - \delta_s)} \tag{15.24}$$

Dielectric losses

The dielectric losses are determined by the dielectric loss tangent.

$$\tan \delta_d = \frac{G'}{\omega C'} \quad \rightarrow \quad G' = \omega C' \cdot \tan \delta_d \tag{15.25}$$

With

$$C' = \frac{1}{\omega} \cdot Im\left\{\frac{\gamma}{Z_L}\right\} \tag{15.26}$$

the equation (15.25) can be rewritten to

$$\begin{aligned} G' &= \frac{\beta}{Z_L} \cdot \tan \delta_d = \frac{\omega}{v_{ph} \cdot Z_L} \cdot \tan \delta_d \\ &= \frac{2\pi \cdot f \cdot \sqrt{\varepsilon_{r,eff}}}{c_0 \cdot Z_L} \cdot \tan \delta_d = \frac{2\pi \cdot \sqrt{\varepsilon_{r,eff}}}{\lambda_0 \cdot Z_L} \cdot \tan \delta_d \end{aligned} \tag{15.27}$$

whereas v_{ph} denotes the phase velocity, c_0 the speed of light, $\varepsilon_{r,eff}$ the effective dielectric constant and λ_0 the freespace wavelength. With these expressions at hand it is possible to find a formula for the dielectric losses of the transmission line.

$$\alpha_d = \frac{1}{2} \cdot G' Z_L = \frac{\pi \cdot \sqrt{\varepsilon_{r,eff}}}{\lambda_0} \cdot \tan \delta_d \tag{15.28}$$

Overall losses of the twisted pair configuration

Transmission losses consist of conductor losses, dielectric losses as well as radiation losses. The above expressions for the conductor and dielectric losses are considered to be first order approximations. The conductor losses have been derived for a single round wire. The overall conductor losses due to the twin wires must be doubled. The dielectric losses can be used as is. Radiation losses are neglected.

Chapter 16

Synthesizing circuits

16.1 Attenuators

Attenuators are used to damp a signal. Using pure ohmic resistors the circuit can be realized for a very high bandwidth, i.e. from DC to many GHz. The power attenuation $0 < L \leq 1$ is defined as:

$$L = \frac{P_{in}}{P_{out}} = \frac{V_{in}^2}{Z_{in}} \cdot \frac{Z_{out}}{V_{out}^2} = \left(\frac{V_{in}}{V_{out}}\right)^2 \cdot \frac{Z_{out}}{Z_{in}} \tag{16.1}$$

where P_{in} and P_{out} are the input and output power and V_{in} and V_{out} are the input and output voltages.

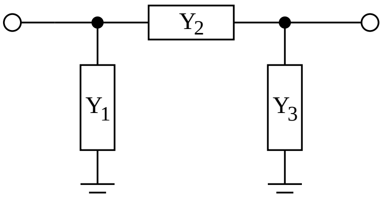


Figure 16.1: π-topology of an attenuator

Fig. 16.1 shows an attenuator using the π-topology. The conductances can be calculated as follows.

$$Y_2 = \frac{L-1}{2 \cdot \sqrt{L \cdot Z_{in} \cdot Z_{out}}} \tag{16.2}$$

$$Y_1 = Y_2 \cdot \left(\sqrt{\frac{Z_{out}}{Z_{in}} \cdot L} - 1\right) \tag{16.3}$$

$$Y_3 = Y_2 \cdot \left(\sqrt{\frac{Z_{in}}{Z_{out}} \cdot L} - 1\right) \tag{16.4}$$

where Z_{in} and Z_{out} are the input and output reference impedances, respectively. The π-attenuator can be used for an impedance ratio of:

$$\frac{1}{L} \leq \frac{Z_{out}}{Z_{in}} \leq L \tag{16.5}$$

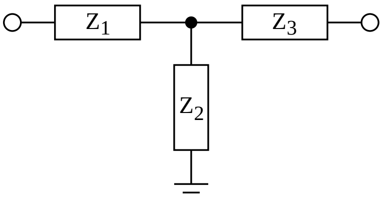


Figure 16.2: T-topology of an attenuator

Fig. 16.2 shows an attenuator using the T-topology. The resistances can be calculated as follows.

$$Z_2 = \frac{2\cdot\sqrt{L\cdot Z_{in}\cdot Z_{out}}}{L-1} \tag{16.6}$$

$$Z_1 = Z_{in}\cdot A - Z_2 \tag{16.7}$$

$$Z_3 = Z_{out}\cdot A - Z_2 \tag{16.8}$$

$$\text{with} \qquad A = \frac{L+1}{L-1} \tag{16.9}$$

where L is the attenuation ($0 < L \leq 1$) according to equation 16.1 and Z_{in} and Z_{out} are the input and output reference impedance, respectively. The T-attenuator can be used for an impedance ratio of:

$$\frac{Z_{out}}{Z_{in}} \leq \frac{(L+1)^2}{4\cdot L} \tag{16.10}$$

16.2 Filters

One of the most common tasks in microwave technologies is to extract a frequency band from others. Optimized filters exist in order to easily create a filter with an appropriate characteristic. The most popular ones are:

Name	Property
Bessel filter (Thomson filter)	as constant group delay as possible
Butterworth filter (power-term filter)	as constant amplitude transfer function as possible
Legendre filter (Optimum L filter)	fastest roll-off with monotonic frequency response
Chebychev filter type I	constant ripple in pass band
Chebychev filter type II	constant ripple in stop band
Cauer filter (elliptical filter)	constant ripple in pass and stop band

From top to bottom the following properties increase:

- ringing of step response
- phase distortion
- variation of group delay
- steepness of amplitude transfer function at the beginning of the pass band

The order n of a filter denotes the number of poles of its (voltage) transfer function. It is:

$$\text{slope of asymptote} = \pm\, n\cdot 20\text{dB/decade} \tag{16.11}$$

Note that this equation holds for all filter characteristics, but there are big differences concerning the attenuation near the pass band.

16.2.1 Transfer functions

The transfer functions of the filter types is usually given for lowpass filters. They can be transformed into the other filter classes by replacing the normalized frequency $S = j\frac{\omega}{\omega_B}$ by the following terms.

$$\text{lowpass} \rightarrow \text{highpass}: \qquad S' = \frac{1}{S} \tag{16.12}$$

$$\text{lowpass} \rightarrow \text{bandpass}: \qquad S' = \frac{S + \frac{1}{S}}{\Delta\Omega} \tag{16.13}$$

$$\text{lowpass} \rightarrow \text{bandstop}: \qquad S' = \frac{\Delta\Omega}{S + \frac{1}{S}} \tag{16.14}$$

$$\text{with} \qquad \Delta\Omega = \frac{\omega_{up} - \omega_{low}}{\omega_{mid}} = \frac{\omega_{up} - \omega_{low}}{\sqrt{\omega_{up} \cdot \omega_{low}}} \tag{16.15}$$

Bessel

The transfer function of an n-th order Bessel lowpass filter writes as follows:

$$A = \frac{A_0}{1 + \sum_{i=1}^{n} c_i \cdot S^i} \tag{16.16}$$

$$\text{with} \quad c_1 = 1 \tag{16.17}$$

$$c_i = \frac{2 \cdot (n - i + 1)}{i \cdot (2n - i + 1)} \cdot c_{i-1} \qquad \text{for} \quad i = 2 \dots n \tag{16.18}$$

Butterworth

The transfer function of an n-th order Butterworth lowpass filter writes as follows:

$$A = \frac{A_0}{\prod_i (1 + a_i \cdot S + b_i \cdot S^2)} \tag{16.19}$$

$$\text{with (even order)} \quad a_i = 2 \cdot \cos\frac{(2i - 1) \cdot \pi}{2n} \qquad \text{for} \quad i = 1 \dots \frac{n}{2} \tag{16.20}$$

$$b_i = 1 \tag{16.21}$$

$$\text{with (odd order)} \quad a_1 = 1 \tag{16.22}$$

$$b_1 = 0 \tag{16.23}$$

$$a_i = 2 \cdot \cos\frac{(i - 1) \cdot \pi}{n} \qquad \text{for} \quad i = 2 \dots \frac{n + 1}{2} \tag{16.24}$$

$$b_i = 1 \tag{16.25}$$

Chebyshev I

The transfer function of an n-th order Chebyshev lowpass filter writes as follows:

$$A = \frac{A_0}{\prod_i (1 + a_i \cdot S + b_i \cdot S^2)} \tag{16.26}$$

$$\text{with (even order)} \quad a_i = 2 \cdot b_i \cdot \sinh\gamma \cdot \cos\frac{(2i-1)\cdot\pi}{2n} \qquad \text{for} \quad i = 1 \ldots \frac{n}{2} \tag{16.27}$$

$$b_i = \frac{1}{\cosh^2\gamma - \cos^2\frac{(2i-1)\cdot\pi}{2n}} \tag{16.28}$$

$$\text{with (odd order)} \quad a_1 = \frac{1}{\sinh\gamma} \tag{16.29}$$

$$b_1 = 0 \tag{16.30}$$

$$a_i = 2 \cdot b_i \cdot \sinh\gamma \cdot \cos\frac{(i-1)\cdot\pi}{n} \qquad \text{for} \quad i = 2 \ldots \frac{n+1}{2} \tag{16.31}$$

$$b_i = \frac{1}{\cosh^2\gamma - \cos^2\frac{(i-1)\cdot\pi}{n}} \tag{16.32}$$

$$\text{with} \quad \gamma = \frac{1}{n} \cdot \mathrm{arsinh}\frac{1}{\sqrt{10^{R_{dB}/10} - 1}} \tag{16.33}$$

The corner frequency f_c of Chebyshev filters is not the -3dB frequency f_{-3dB}. But this can be easily transformed by the following equation:

$$f_c = f_{-3dB} \cdot \cosh\left(\frac{1}{n} \cdot \mathrm{arcosh}\frac{1}{\sqrt{10^{R_{dB}/10} - 1}}\right) \tag{16.34}$$

16.2.2 LC ladder filters

The best possibility to realize a filters in VHF and UHF bands are LC ladder filters. The usual way to synthesize them is to first calculate a low-pass (LP) filter and afterwards transform it into a high-pass (HP), band-pass (BP) or band-stop (BS) filter. To do so, each component must be transformed into another.

In a low-pass filter, there are parallel capacitors C_{LP} and series inductors L_{LP} in alternating order. The other filter classes can be derived from it:

In a high-pass filter:

$$C_{LP} \quad \rightarrow \quad L_{HP} = \frac{1}{\omega_B^2 \cdot C_{LP}} \tag{16.35}$$

$$L_{LP} \quad \rightarrow \quad C_{HP} = \frac{1}{\omega_B^2 \cdot L_{LP}} \tag{16.36}$$

In a band-pass filter:

$$\begin{aligned}
C_{LP} \quad &\rightarrow \quad \text{parallel resonance circuit with} && (16.37)\\
&C_{BP} = \frac{C_{LP}}{\Delta\Omega} && (16.38)\\
&L_{BP} = \frac{\Delta\Omega}{\omega_1 \cdot \omega_2 \cdot C_{LP}} && (16.39)\\
L_{LP} \quad &\rightarrow \quad \text{series resonance circuit with} && (16.40)\\
&C_{BP} = \frac{\Delta\Omega}{\omega_1 \cdot \omega_2 \cdot L_{LP}} && (16.41)\\
&L_{BP} = \frac{L_{LP}}{\Delta\Omega} && (16.42)
\end{aligned}$$

In a band-stop filter:

$$\begin{aligned}
C_{LP} \quad &\rightarrow \quad \text{series resonance circuit with} && (16.43)\\
&C_{BP} = \frac{C_{LP}}{2 \cdot \left|\frac{\omega_2}{\omega_1} - \frac{\omega_1}{\omega_2}\right|} && (16.44)\\
&L_{BP} = \frac{1}{\omega^2 \cdot \Delta\Omega \cdot C_{LP}} && (16.45)\\
L_{LP} \quad &\rightarrow \quad \text{parallel resonance circuit with} && (16.46)\\
&C_{BP} = \frac{1}{\omega^2 \cdot \Delta\Omega \cdot L_{LP}} && (16.47)\\
&L_{BP} = \frac{L_{LP}}{2 \cdot \left|\frac{\omega_2}{\omega_1} - \frac{\omega_1}{\omega_2}\right|} && (16.48)
\end{aligned}$$

Where

$$\begin{aligned}
\omega_1 \quad &\rightarrow \quad \text{lower corner frequency of frequency band} && (16.49)\\
\omega_2 \quad &\rightarrow \quad \text{upper corner frequency of frequency band} && (16.50)\\
\omega \quad &\rightarrow \quad \text{center frequency of frequency band} \quad \omega = 0.5 \cdot (\omega_1 + \omega_2) && (16.51)\\
\Delta\Omega \quad &\rightarrow \quad \Delta\Omega = \frac{|\omega_2 - \omega_1|}{\omega} && (16.52)
\end{aligned}$$

Butterworth

The k-th element of an n order Butterworth low-pass ladder filter is:

$$\begin{aligned}
\text{capacitance:} \quad C_k &= \frac{X_k}{Z_0} && (16.53)\\
\text{inductance:} \quad L_k &= X_k \cdot Z_0 && (16.54)\\
\text{with} \quad X_k &= \frac{2}{\omega_B} \cdot \sin \frac{(2 \cdot k + 1) \cdot \pi}{2 \cdot n} && (16.55)
\end{aligned}$$

The order of the Butterworth filter is dependent on the specifications provided by the user. These specifications include the edge frequencies and gains.

$$n = \frac{\log\left(\frac{10^{-0.1 \cdot \alpha_{stop}} - 1}{10^{-0.1 \cdot \alpha_{pass}} - 1}\right)}{2 \cdot \log\left(\frac{\omega_{stop}}{\omega_{pass}}\right)} \tag{16.56}$$

Chebyshev I

The equations for a Chebyshev type I filter are defined recursivly. With R_{dB} being the passband ripple in decibel, the k-th element of an n order low-pass ladder filter is:

$$\text{capacitance:} \quad C_k = \frac{X_k}{Z_0} \tag{16.57}$$

$$\text{inductance:} \quad L_k = X_k \cdot Z_0 \tag{16.58}$$

$$\text{with} \quad X_k = \frac{2}{\omega_B} \cdot g_k \tag{16.59}$$

$$r = \sinh\left(\frac{1}{n} \cdot \text{arsinh} \frac{1}{\sqrt{10^{R_{dB}/10} - 1}}\right) \tag{16.60}$$

$$a_k = \sin \frac{(2 \cdot k + 1) \cdot \pi}{2 \cdot n} \tag{16.61}$$

$$g_k = \begin{cases} \dfrac{a_k}{r} & \text{for} \quad k = 0 \\ \dfrac{a_{k-1} \cdot a_k}{g_{k-1} \cdot \left(r^2 + \sin^2 \dfrac{k \cdot \pi}{n}\right)} & \text{for} \quad k \geq 1 \end{cases} \tag{16.62}$$

The ripple of even order Chebyshev filters is above the 0dB line. Thus, they can't be realized with passive circuits (at least not with input and output impedance being equal). The -3dB frequency transformation is done by equation 16.34.

The order of the Chebychev filter is dependent on the specifications provided by the user. The general form of the calculation for the order is the same as for the Butterworth, except that the inverse hyperbolic cosine function is used in place of the common logarithm function.

$$n = \frac{\text{sech}\left(\dfrac{10^{-0.1 \cdot \alpha_{stop}} - 1}{10^{-0.1 \cdot \alpha_{pass}} - 1}\right)}{2 \cdot \text{sech}\left(\dfrac{\omega_{stop}}{\omega_{pass}}\right)} \tag{16.63}$$

Chebyshev II

Because of the nature of the derivation of the inverse Chebychev approxiation function from the standard Chebychev approximation the calculation of the order (16.63) is the same.

16.2.3 End-coupled transmission line bandpass filters

Filters for the lower microwave bands (4 to 12 GHz) can be easily realized by transmission line filters. Figure 16.3 shows a capacitive-coupled, half-wavelength resonator bandpass filter. The necessary design steps are described now [87].

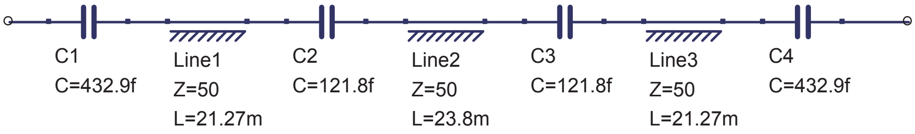


Figure 16.3: 3rd order half-wavelength bandpass filters

First the (normalized) characteristic admittances of the J-inverters are calculated:

$$J_k = \sqrt{\frac{\pi \cdot \Delta f}{2 \cdot g_k \cdot g_{k+1}}} \qquad \text{for} \quad k = -1 \quad \text{and} \quad k = n-1 \tag{16.64}$$

$$J_k = \frac{\pi \cdot \Delta f}{2 \cdot \sqrt{g_k \cdot g_{k+1}}} \qquad \text{for} \quad k = 0 \ldots n-2 \tag{16.65}$$

$$\tag{16.66}$$

where n is the order of the filter, $g_0 \ldots g_{n-1}$ are the normalized elements of a LC ladder-type lowpass filter ($g_k = X_k \cdot \omega_B$, see section 16.2.2), g_{-1} and g_n equals 1, $\Delta f = (f_{up} - f_{low})/f_{mid}$ is the relative bandwidth.

Secondly, the susceptances B_k are computed.

$$B_k \cdot Z_0 = \frac{J_k}{1 - J_k^2} \tag{16.67}$$

Finally, the coupling capacitances C_k and the length l_k of the transmission lines can be determined.

$$C_k = \frac{B_k}{\omega_{mid}} \tag{16.68}$$

$$l_k = \frac{c_0}{2\omega_{mid}} \cdot (2\pi - \arctan(2 \cdot B_{k-1} \cdot Z_0) - \arctan(2 \cdot B_k \cdot Z_0)) \tag{16.69}$$

The characteristic impedance of the transmission lines equals the system impedance Z_0. Note that all transmission line filters are periodic and thus, they also show passband behaviour at the harmonic frequencies ($2\omega_{mid}, \quad 3\omega_{mid}, \quad 4\omega_{mid}, \quad \ldots$).

This kind of bandpass filter can be realized in microstrip topology. The transmission lines are replaced by microstrip lines and the capacitors are replaced by microstrip gaps. The microstrip width is calculated to fit the system impedance and the gap length is calculated to fit the coupling capacitance C_k (see section 12.4). The length of the microstrip lines must be shortened according to the effective relative permittivity and according to the end-effect of the microstrip open (see section 12.4 and 12.3). Because the coupling capacitance of a microstrip gap is quite small the relative bandwidth that can be achieved with this filter topology is small, too (up to approximately 3%).

16.2.4 Lateral-coupled transmission line bandpass filters

Coupled transmission lines are well suited to design bandpass filters for frequencies from 1 to 40 GHz. Figure 16.4 shows an example. The design steps follow below [87].

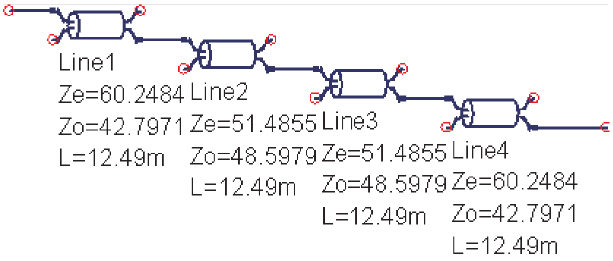


Figure 16.4: 3rd order coupled transmission line bandpass filters

The calculation again starts with the (normalized) characteristic admittances of the J-inverters (see equation 16.65 and 16.66). They are used to determine the characteristic impedances of the even and the odd mode ($k = 0 \dots n$).

$$Z_{0e,k} = Z_0 \cdot \left(1 + J_k + J_k^2\right) \tag{16.70}$$

$$Z_{0o,k} = Z_0 \cdot \left(1 - J_k + J_k^2\right) \tag{16.71}$$

The length of the transmission lines is a quarter of the wavelength.

$$l_k = \frac{c_0}{4 \cdot f_{mid}} \tag{16.72}$$

This kind of bandpass filter can nicely be realized in microstrip topology. The characteristic impedances of even and odd mode are used to calculate the widths and the gaps of the lines. The lengths have to be reduced by the effective relative permittivity ϵ_{re} and ϵ_{ro} and by the open-end effect of the microstrip lines.

$$l_k = \frac{c_0}{4 \cdot f_{mid} \cdot \sqrt[4]{\epsilon_{re,k} \cdot \epsilon_{ro,k}}} - \Delta l_{open} \tag{16.73}$$

16.2.5 Stepped-impedance lowpass filters

The z-parameters of a transmission line are

$$Z_{11} = Z_{22} = -j \cdot Z_0 \cdot \cot(\beta l) \tag{16.74}$$

$$Z_{12} = Z_{21} = -j \cdot Z_0 \cdot \operatorname{cosec}(\beta l) \tag{16.75}$$

Because Z_{12} and Z_{21} are equal, this can replace a T-topology (see section 19.1.3) with the series elements being each

$$j \cdot X = Z_{11} - Z_{12} = j \cdot Z_0 \cdot \frac{1 - \operatorname{cosec}(\beta l)}{\sin(\beta l)} = j \cdot Z_0 \cdot \tan\left(\frac{\beta l}{2}\right) \tag{16.76}$$

For short lines the impedance X of the series elements and admittance B of the shunt element gives the following.

$$X \approx Z_0 \cdot \beta l \tag{16.77}$$

$$B \approx Y_0 \cdot \beta l \tag{16.78}$$

For a transmission line with large characteristic impedance Z_h the series elements dominate. Thus, it behaves like an inductance. For a transmission line with small characteristic impedance Z_l the shunt element dominates. Thus, it behaves like a capacitance. A sequence of short transmission lines with alternating low and high impedances results in a lowpass filter. The line lengthes are

$$l_{high} = g_k \cdot \frac{Z_0}{Z_h} \cdot \frac{c_0}{\omega_B} \tag{16.79}$$

$$l_{low} = g_k \cdot \frac{Z_l}{Z_0} \cdot \frac{c_0}{\omega_B} \tag{16.80}$$

with Z_0 being the impedance that the filter is designed for and $g_k = X_k \cdot \omega_B$ (see section 16.2.2). To get good performance Z_l must be as low as possible and Z_h must be as high as possible. Nonetheless, the stop-band attenuation won't reach the typical value of the filter type used. This is because of the approximations done above. Sometimes, the result can be improved by adding the inductance of the previous and/or following low-impedance line to the one of the high-impedance line. The length then becomes:

$$l_{high} = \frac{Z_0}{Z_h} \cdot \frac{c_0}{\omega_B} \cdot \left(g_k - 0.5 \cdot g_{k-1} \cdot \left(\frac{Z_l}{Z_0}\right)^2 - 0.5 \cdot g_{k+1} \cdot \left(\frac{Z_l}{Z_0}\right)^2\right) \tag{16.81}$$

Sometimes, performance is also improved by lowing the capacitance of the low-impedance line in order to compensate for the capacitance of the adjacent high-impedance lines. The line length becomes for examples:

$$l_{low} = \frac{c_0}{\omega_B} \cdot \left(g_k \cdot \frac{Z_l}{Z_0} - 0.1 \cdot \frac{Z_l}{Z_h} \right) \tag{16.82}$$

16.2.6 Active filters

Active RC filters are a good choice for frequencies below 1MHz. Here, the multi-feedback topology will be discussed because it shows better performance in reality than the Sallen-Key topology.

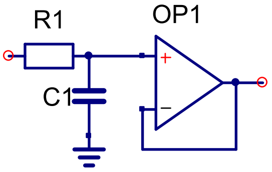


Figure 16.5: active 1st order lowpass filter

Figure 16.5 shows a 1st order active low-pass filter. The number of components leaves some degree of freedom. The following procedure is recommended:

$$\text{choose} \quad C_1 \qquad \text{(e.g. E6 serie)} \tag{16.83}$$

$$R_1 = \frac{a_1}{\omega_B \cdot C1} \tag{16.84}$$

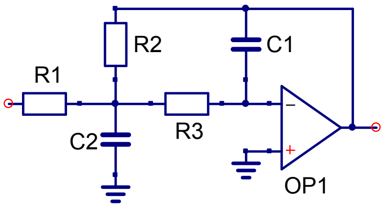


Figure 16.6: active 2nd order lowpass filter

Figure 16.6 shows a 2nd order active low-pass filter. The number of components leaves some degree of freedom. The following procedure is recommended:

$$\text{choose} \quad C_1 \qquad \text{(e.g. E6 serie)} \tag{16.85}$$

$$\text{choose} \quad C_2 \geq \frac{4 \cdot b_1 \cdot (1 + A_0)}{a_1^2} \cdot C_1 \qquad \text{(e.g. E6 serie)} \tag{16.86}$$

$$R_2 = \frac{a_1 \cdot C_2 - \sqrt{a_1^2 \cdot C_2^2 - 4 \cdot C_1 \cdot C_2 \cdot b_1 \cdot (1 + A_0)}}{2 \cdot \omega_B \cdot C_1 \cdot C_2} \tag{16.87}$$

$$R_1 = \frac{R_2}{A_0} \tag{16.88}$$

$$R_3 = \frac{b_1}{\omega_B^2 \cdot C_1 \cdot C_2 \cdot R_2} \tag{16.89}$$

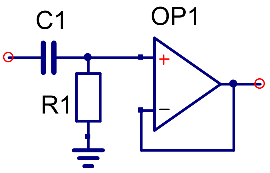


Figure 16.7: active 1st order highpass filter

Figure 16.7 shows a 1st order active high-pass filter. The number of components leaves some degree of freedom. The following procedure is recommended:

$$\text{choose} \quad C_1 \qquad \text{(e.g. E6 serie)} \tag{16.90}$$

$$R_1 = \frac{1}{\omega_B \cdot a_1 \cdot C1} \tag{16.91}$$

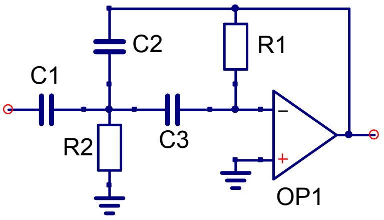


Figure 16.8: active 2nd order highpass filter

Figure 16.8 shows a 2nd order active high-pass filter. The number of components leaves some degree of freedom. The following procedure is recommended:

$$\text{choose} \quad C_1 \qquad \text{(e.g. E6 serie)} \tag{16.92}$$

$$C_2 = C_3 = \frac{C_1}{A_0} \tag{16.93}$$

$$R_1 = \frac{2 + A_0}{a_1 \cdot \omega_B \cdot C_2} \tag{16.94}$$

$$R_2 = \frac{a_1}{\omega_B \cdot b_1 \cdot C_2 \cdot (2 + A_0)} \tag{16.95}$$

The terms a_1 and b_1 in the equations above are the constants for the wanted filter type (see 16.2.1). Several cascaded filters can create all filter types with even order by replacing these terms with a_2 and b_2 for the second stage etc. Odd order filters need a first order filter as first stage.

Chapter 17

System Simulation

System simulations are used to analyze a transmission system on the modulation level. The principle is quite simple. The data consists of an array of complex numbers. Each number represents the amplitude and phase of the modulated carrier at a specific time. A data block is sent to a component which modifies it according to its function. Afterwards the array is sent to the next component. This procedure repeats until all componets are done.

17.1 Component models

17.1.1 Mach-Zehnder Modulator

A Mach-Zehnder modulator (MZM) is an interferometer that is often used to modulate optical signals. It consists of a power splitter, two waveguides and a power combiner. The delay caused by the waveguides changes according to the modulation voltage and thus, creating a constructive or destructive interference at the output E_{out}, i.e. the transfer function is a cosine. The MZM can be used as amplitude modulator (when biased at quadrature) or as phase modulator (when biased at minimum). An MZM in push-pull configuration can be modeled by the following equation:

$$E_{out}(t) = \frac{j \cdot E_{in}(t)}{\sqrt{L} \cdot (1 + \sqrt{ER})} \cdot \exp\left(j \cdot \alpha \cdot \frac{\pi}{2V_\pi}\right) \left(\exp(-X) - \frac{\sqrt{ER} - 1}{\sqrt{ER} + 1} \cdot \exp(X)\right) \tag{17.1}$$

$$\text{with} \qquad X = j\frac{\pi}{2V_\pi} \cdot (V_{bias} - V_{mod}) \tag{17.2}$$

with V_{bias} DC bias voltage, V_{mod} modulation voltage, V_π voltage for full modulation, α chirp coefficient, ER extinction rate, L power loss. Note a finite extiction ratio creates a chirp. The chirp coefficient here produces an additional chirp.

17.1.2 Delay-Line Interferometer

A delay-line interferometer (DLI) is used in optical telecommunication systems to convert a phase modulation into an amplitude modulation. It consists of a power splitter, two waveguides and a power combiner, which a constructive and a destructive output. The delays of the waveguides differ by the symbol duration T. It can be modeled by the following equation:

$$E_{constructive}(t) = \frac{E_{in}(t) + \gamma \cdot E_{in}(t - 1/FSR)}{\sqrt{2 \cdot L \cdot (1 + \gamma^2)}} \tag{17.3}$$

$$E_{destructive}(t) = \frac{E_{in}(t) - \gamma \cdot E_{in}(t - 1/FSR)}{\sqrt{2 \cdot L \cdot (1 + \gamma^2)}} \tag{17.4}$$

$$\text{with} \qquad \gamma = \exp(j \cdot \phi) \cdot \frac{\sqrt{ER} - 1}{\sqrt{ER} + 1} \tag{17.5}$$

with $FSR = 1/T$ free spectral range, L power loss, ER extinction rate, ϕ phase of delayed waveguide.

Chapter 18

Electromagnetic field simulations

18.1 Introduction

Circuit simulations are very powerful analysis tools in electrical engineering, but of course limits exist for their application. A typical example is an arbitrary printed circuit board (PCB) layout at very high frequencies where circuit models for the components are not precise enough or not available at all. At this point, computational electromagnetics (CEM) plays an important role. It solves the differential equation for the electromagnetic field, and therefore use the most fundamental approach in physics. The most common methods are the following ones:

- **FDTD** (finite-difference time-domain) is the most popular method in CEM. It is a partial differential equation (PDE) technique that is easy to understand, easy to implement and very powerful.
  The discretization is restricted to structured rectangular hexahedrons (known as Yee grid). Thus, the mesh creation is a difficult task, especially with complex or multiscale geometries, because the smallest detail determines the cell size.
  In contrast to all frequency-domain methods, FDTD doesn't build a matrix, but solves the domain by simple cell-by-cell operations. Hence, the memory and computation complexity rises linearly with the problem size, i.e. $O(n)$. FDTD is therefore the best choice for very large structures.
  For calculating the frequency response, the circuit is usually excited by a gaussian-shaped pulse. Then, the solver iterates over time until the energy has decreased to a specific level.
  A different approach is to use the **FDFD** (finite-difference frequency-domain) method.
  Famous projects are:
  Meep → http://ab-initio.mit.edu/wiki/index.php/Meep
  openEMS → http://www.openEMS.de/

- **TLM** (transmission line matrix) is a time domain analysis that uses Huygen's principle of wave propagation in space and time. Furthermore, frequency domain variants exist, too. It is very similar to FDTD, and hence both methods share most of its advantages and disadvantages. The most famous projects are:
  Yatpac → http://www.yatpac.org
  emGine → http://www.petr-lorenz.com/emgine/

- **FEM** (finite element method) is the most popular frequency domain method. It is a PDE method and is very universal and powerful. The domain can be discretized with unstructured grids (usually tetrahedrons). Nonetheless, it usually needs more resources (memory and computation time) than FDTD.
  The matrix to be solved is very sparse, so the computation complexity rises quadratically with the problem size, i.e. $O(n^2)$.
  Promising projects are:

GetDP → http://geuz.org/getdp/
Elmer → http://www.csc.fi/english/pages/elmer

- **MoM** (method of moments) or **BEM** (boundary element method) is mostly a frequency domain analysis, even though a time-domain version exists. It's a surface integral technique and is especially successful in simulating classical antennas, but also very useful for planar structures like PCBs.
  This method needs to discretize the surfaces (or metal layers) only and includes the environment (like the PCB substrate) by use of the Green's function. Hence, It's the best choice for small and middle-sized planar structures.
  The matrix to be solved is dense but small. Filling the matrix consumes by far the most time, so the computation complexity rises very strongly with the problem size. Anyway, an extension called **MLFMM** (multilevel fast multipole method) exists that can reduce the complexity to $O(N \cdot \log(N))$.
  The most famous projects are:
  NEC2 → http://www.si-list.net/swindex.html
  GLMoM → http://glmom.sourceforge.net/
  SCUFF-EM → http://homerreid.dyndns.org/scuff-EM/
  Puma-EM → http://puma-em.sourceforge.net/

As usual each method has its advantages and disadvantages. PDE techniques are very successful in simulating closed volumes with inhomogeneous dielectrics (e.g. cavity resonators or shielded transmission lines), whereas integral techniques are well suited to model conducting surfaces in open domain problems (e.g. radiation and scattering). Modern EM simulators often use hybrid methods that combine the advantages of both types of formulations. This way it is possible to precisely calculate almost every property of almost every structure. The disadvantage of field solvers is the huge amount of memory and computation time that are consumed.

Some terms often used in electromagnetics should also be mentioned:

- **PDE** - partial differential equation
- **ABC** - absorbing boundary condition
- **PML** - perfectly matched layer (a special ABC)
- **PEC** - perfect electrical conductor
- **PMC** - perfect magnetical conductor
- **EMI** - electromagnetic interference
- **EMC** - electromagnetic compatibility

Boundary conditions (B.C.) are needed to solve a differential equation. The most important ones are the following:

- **Dirichlet B.C.** → $V = 0$, e.g. tangential electric field on metallic surfaces
- **inhomogeneous Dirichlet B.C.** → $V = Q_b$, e.g. electric potential on metallic surfaces
- **Neumann B.C.** → $\nabla V \cdot \vec{n} = \frac{\partial V}{\partial n} = 0$ with n being orthonormal to boundary, e.g. dynamic magnetic field on metallic surfaces
- **inhomogeneous Neumann B.C.** → $\frac{\partial V}{\partial n} = Q_b$, e.g. surface charge in electrostatics
- **Impedance B.C.** → $\frac{\partial V}{\partial n} + a \cdot V = 0$, e.g. first order absorbing condition

18.2 Finite-Difference Time-Domain (FDTD) Method

The FDTD method [88] discretizes the simulation domain into a structured mesh with rectangular blocks. Then, it calculates the electric field along the edges of each block. The grid for the magnetic field lies half a block displaced and is calculated for half a time step later. This discretization is called Yee cell, see figure 18.1.
The electric field is computed from the magnetic field by Ampere's law:

$$\oint_{\partial A} \vec{H} \cdot d\vec{s} = +\frac{\partial}{\partial t} \iint_A \epsilon \vec{E} \cdot d\vec{A} + \iint_A \kappa \vec{E} \cdot d\vec{A} \tag{18.1}$$

The result is stored and used to compute the magnetic field for the next half time step by Faraday's law:

$$\oint_{\partial A} \vec{E} \cdot d\vec{s} = -\frac{\partial}{\partial t} \iint_A \mu \vec{H} \cdot d\vec{A} + \iint_A \sigma \vec{H} \cdot d\vec{A} \tag{18.2}$$

The result is stored and used to repeat the calculation of the electric field for the next time step. This is done again and again, so the incident field moves through the domain according to Huygens' principle.

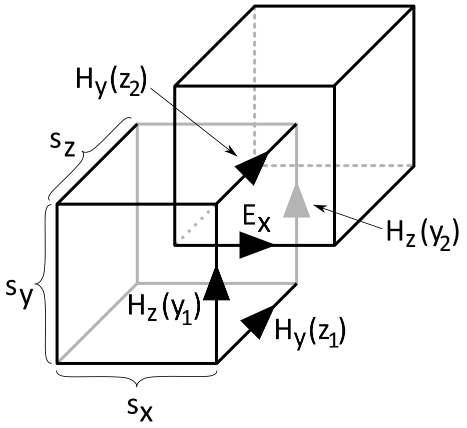


Figure 18.1: Yee cell showing discretization of Ampere's law for E_x

According to figure 18.1, the discretization of Ampere's law yields:

$$H_y(z_2) \cdot s_y - H_y(z_1) \cdot s_y - H_z(y_2) \cdot s_y + H_z(y_1) \cdot s_y \approx \epsilon \cdot \frac{\partial}{\partial t} E_x \cdot s_y \cdot s_z + \kappa E_x \cdot s_y \cdot s_z \tag{18.3}$$

It's now the better choice to use voltages and currents instead of the field variables, i.e. we set:

$$i_y = H_y \cdot s_y \qquad i_z = H_z \cdot s_z \qquad v_x = E_x \cdot s_x \tag{18.4}$$

$$C_x = \frac{\epsilon \cdot s_y \cdot s_z}{s_x} \qquad G_x = \frac{\epsilon \cdot s_y \cdot s_z}{s_x} \tag{18.5}$$

The method then is called equivalent-circuit FDTD (EC-FDTD). Next, the time derivative is replaced by its difference quotient and the voltages at half a time step is approximated by the average of its values from the time step before and after. This gives:

$$\begin{aligned} &i_y(z_2, t_{0.5}) - i_y(z_1, t_{0.5}) - i_z(y_2, t_{0.5}) + i_z(y_1, t_{0.5}) \\ &\qquad \approx C_x \cdot \frac{v_x(t_1) - v_x(t_0)}{\Delta t} + G_x \cdot \frac{v_x(t_1) - v_x(t_0)}{2} \end{aligned} \tag{18.6}$$

Finally, the equation can be solved for $v_x(t_1)$ which yields:

$$\begin{aligned} v_x(t_1) \approx{} & \frac{2 \cdot C_x - \Delta t \cdot G_x}{2 \cdot C_x + \Delta t \cdot G_x} \cdot v_x(t_0) \\ & + \frac{2 \cdot \Delta t}{2 \cdot C_x + \Delta t \cdot G_x} \cdot \left(i_y(z_2, t_{0.5}) - i_y(z_1, t_{0.5}) - i_z(y_2, t_{0.5}) + i_z(y_1, t_{0.5})\right) \end{aligned} \tag{18.7}$$

This equation calculates the voltage $v_x(t_1)$ from the results of prior time steps. The same can be performed for all other variables v_y, v_z, i_x, i_y and i_z. So, we have got a simple algorithm to perform an electro-magnetic field simulation in time domain.

18.2.1 Discretization

A disadvantage of the FDTD algorithm is the need for a structured mesh. A small cell size at any position causes a dense grid over the whole domain. The method also has problems with high field densities, that for example appear at the edges of microstrip lines. Very small cell sizes are necessary here. Laterally, this can be relaxed by the so-called 1/3 2/3 rule (see figure 18.2). I.e. the grid isn't set precisely on the edge, but shifted by one third of the cell size into the metal strip.

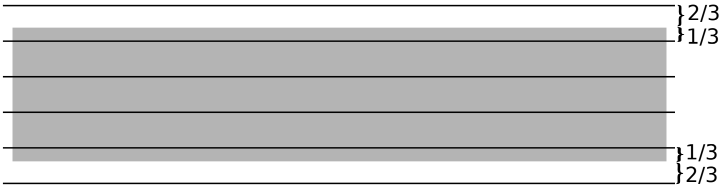


Figure 18.2: A typical FDTD mesh for a microstrip line using the 1/3 2/3 rule

18.2.2 Excitation

Usually, the structure is excited by a short pulse, the simulation is performed until the field energy has decayed below a certain limit and the result is transformed into frequency domain. Obviously, the pulse must excite the frequency range of interest. Furthermore, two problems may encounter. First, many structures tend to store DC charges that prevent the energy to decay if the excitation contains DC energy. And second, if the structure creates high-Q resonances, the energy decay is strongly delayed.

The most popular pulse shape is Gaussian, because it covers a broad frequency range and afterwards exhibits a very steep roll-off. DC-free variants exists, too. So, Gaussian pulses can be trimmed to excite the wanted frequency range only (omit DC and/or resonances if needed). The following pulse shape is quite popular:

$$v(t) = \cos\left(2 \cdot \pi \cdot f_0 \cdot (t - t_0)\right) \cdot \exp\left(-9 \cdot \left(\frac{t}{t_0} - 1\right)^2\right) \tag{18.8}$$

$$\text{with} \qquad t_0 = \frac{9}{2 \cdot \pi \cdot f_c} \tag{18.9}$$

with f_0 being the center frequency, i.e. the one with the highes energy. The pulse reaches its maximum at t_0. It has to be applied to the simulation domain from time 0 to $2 \cdot t_0$. The parameter f_c determines frequency decay. Three parameter pairs are used often (see figure 18.3 and 18.4):

- broadband: $f_0 = 0$ and $f_c = 1.8 \cdot f_{max}$
  Fastest FDTD convergence, most energy at DC

- middle: $f_0 = 2 \cdot f_{max}$ and $f_c = f_{max}$
  Slower FDTD convergency, steeper frequency roll-off, most energy at DC

- DC min: $f_0 = 0.5 \cdot (f_{max} + f_{min})$ and $f_c = 0.5 \cdot (f_{max} - f_{min})$
  Slowest FDTD convergency, steepest frequency roll-off, small DC energy
  (If $f_{min} > 0.33 \cdot (f_{max} - f_{min})$, the DC energy is below -40dB, so it can be considered as DC free.)

If a DC free pulse and a low frequency excitation is required, the above-mentioned function isn't the best choice. A better suited example is the Ricker wavelet (see figure 18.3 and 18.4):

$$v(t) = \left(1 - 2 \cdot x^2\right) \cdot \exp\left(-x^2\right) \tag{18.10}$$

$$x = \pi \cdot f_0 \cdot (t - t_0) \tag{18.11}$$

$$\text{with} \qquad t_0 = \frac{1}{2 \cdot \pi \cdot f_0} \tag{18.12}$$

with f_0 being the frequency with the highest energy. Again, the pulse reaches its maximum at t_0. It has to be applied to the simulation domain from time 0 to $2 \cdot t_0$. The usable frequency range is about $0.22 \cdot f_0$ to $2 \cdot f_0$.

A function with an even lower cut-off frequency is the following pulse:

$$v(t) = 8.1 \cdot x \cdot \exp\left(-12.0 \cdot x^2\right) \tag{18.13}$$

$$x = f_0 \cdot (t - t_0) \tag{18.14}$$

$$\text{with} \qquad t_0 = \frac{1.5}{f_0} \tag{18.15}$$

Once more,it has to be applied to the simulation domain from time 0 to $2 \cdot t_0$. The usable frequency range is about $0.06 \cdot f_0$ to $2 \cdot f_0$.

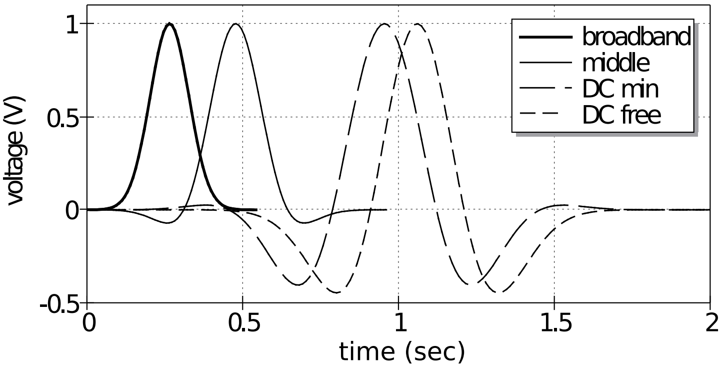


Figure 18.3: Pulse shapes often used for FDTD simulations ($f_{min} = 0.1$Hz, $f_{max} = 3$Hz)

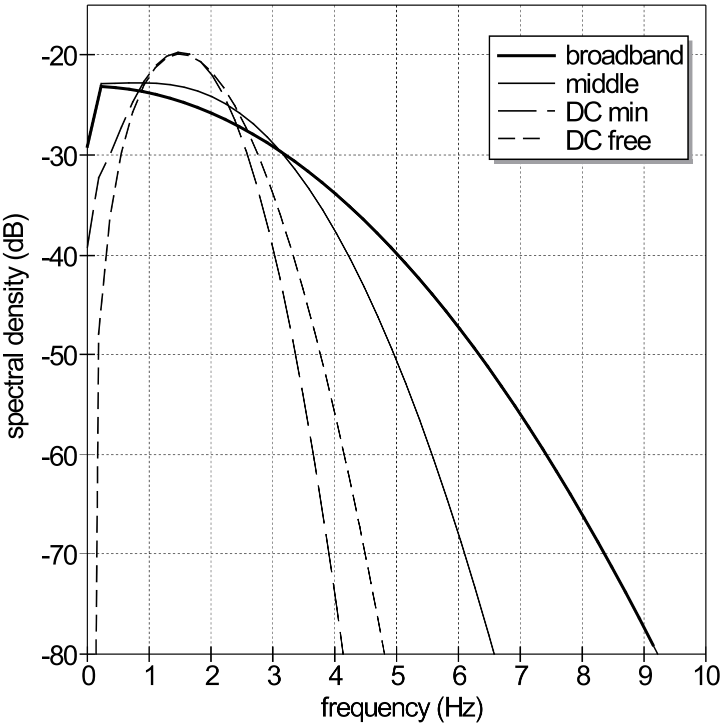


Figure 18.4: Frequency domain representation of pulse shapes ($f_{min} = 0.1\text{Hz}$, $f_{max} = 3\text{Hz}$)

18.3 Finite Element Method (FEM)

18.3.1 Overview

FEM is a universal method to solve partial differential equations (PDE). It is used in many scientific fields like electro- and magnetostatics, heat transfer, mechanical stress analysis, fluid flows etc. The steps to perform an FEM analysis can be shortly summerized as follows:

1. The user defines the properties and the boundary of the geometry under investigation (size, material etc.). He also specifies the excitation (e.g. current source, incident electric field).

2. Discretization: The preprocessor creates the mesh, i.e. it divides the user-defined domain into spatial pieces (finite elements) that are small enough to guarantee accuracy, but that are still few enough to keep the computation time low.

3. The PDE, the boundary conditions and the excitations are applied to every element in the domain. This way, the matrix A and the right-hand side vector b are created.

4. The simulator solves the matrix equation $A \cdot x = b$ to get the unknown quantity x (e.g. electric field) at every element of the domain.

5. The postprocessor uses the result x to calculate all wanted quantities (e.g. s-parameters, capacitance, impedance etc.)

Creating the discretization (step 2) is a very complex task that needs sophisticated algorithms. Usually, stand-alone programs perform this meshing. The most famous projects are as follows:

Gmsh → http://www.geuz.org/gmsh/
Tetgen → http://wias-berlin.de/software/tetgen/

Netgen → http://netgen-mesher.sourceforge.net/

The next subsections explain step 3 in more detail. The technical background for performing an FEM analysis is described by typical examples.

18.3.2 Electrostatics

Because electrostatic fields are curl-free ($\nabla \times \vec{E} = \vec{0}$), they can be fully represented by a scalar potential V, defined as:

$$\vec{E} = -\nabla V \tag{18.16}$$

The PDE describing all electrostatic phenomena is quite simple therefore:

$$-\nabla \cdot (\epsilon \nabla V) = q_V \tag{18.17}$$

with permittivity ϵ and volume charge density in As/m^3. In order to solve the equation numerically, the computational domain needs to be split into a finite number of small pieces, most often into tetrahedrons (3D space) or triangles (2D space), respectively. Within these elements, a shape (or basis) function W_k interpolates the electrostatic potential:

$$V = \sum_k V_k \cdot W_k \tag{18.18}$$

Scalar quantities are usually modeled with node-based shape functions, i.e. the unknowns V_k are the values at the nodes of the elements. Because a tetrahedron has four nodes (figure 18.5), it creates four unknowns in the matrix equation ($k = 4$).

The basic concept of FEM is to multiply the PDE with a weighting funkction W and integrate the product over the volume Ω of each element:

$$\iiint\limits_{\Omega} PDE(\vec{r}) \cdot W(\vec{r})\ dv \tag{18.19}$$

Hence, the PDE is not forced to be valid everywhere. Instead, a weighted average is used, which is named weak form of the PDE. In case of equation 18.17 this leads to:

$$-\iiint\limits_{\Omega} \nabla \cdot (\epsilon \nabla V) \cdot W\ dv = \iiint\limits_{\Omega} q_V \cdot W\ dv \tag{18.20}$$

By use of Green's first identity, the double differential operator can be avoided, leading to:

$$\iiint\limits_{\Omega} \epsilon \nabla V \cdot \nabla W\ dv - \oiint\limits_{\partial\Omega} \epsilon \nabla V \cdot W\ d\vec{S} = \iiint\limits_{\Omega} q_V \cdot W\ dv \tag{18.21}$$

By replacing V with the interpolation formula (equation 18.18) the unknowns can be separated from the integrals. Using the same function W as basis and as weighting function (Galerkin's method) leads to the final matrix equation:

$$(\boldsymbol{P} - \boldsymbol{B}) \cdot \boldsymbol{V} = \boldsymbol{Q} \tag{18.22}$$

with the following matrix and vector entries:

$$\begin{aligned}
\text{potential} \qquad \boldsymbol{P}_{nm} &= \iiint \epsilon \cdot \nabla W_n \cdot \nabla W_m\ dv && (18.23)\\
\text{boundaries} \qquad \boldsymbol{B}_n &= \oiint \vec{D} \cdot W_n\ d\vec{S} + \oiint q_S \cdot W_n\ dS && (18.24)\\
\text{sources} \qquad \boldsymbol{Q}_n &= \iiint q_V \cdot W_n\ dv && (18.25)
\end{aligned}$$

These integral equations can be solved directly, because all terms are known:

1. The user defines material properties (permittivity ϵ, volume charge density q_V) and the boundary conditions (incoming flux $\vec{n} \cdot \vec{D}$, surface charge density q_S, fixed potential on metallic surfaces).
2. The pre-processor determines the integration domains by creating the mesh.
3. The basis functions W_n are pre-defined and pre-calculated formulas.

Thus, the matrix $(\boldsymbol{P} - \boldsymbol{B})$ and the right-hand side vector $\boldsymbol{Q}$ are filled by stamping. A matrix entry is non-zero only if it belongs to two nodes that share the same element. Hence, the matrix is very sparse. Furthermore, it can be seen that an FEM matrix is always structurally symmetric.

In order to save resources, it is usual practice to exploit symmetries. Through a symmetry plane, no electrical flux is flowing. Therefore, it is sufficient to simulate one half of the domain only by keeping the new boundary open (no boundary condition at all). If a symmetry plane exists, but charge and potential have different signs on each side, the symmetry plane can be replaced by a fixed potential of zero. Again one half of the domain needs to be simulated only.

18.3.3 Heat Transfer

The heat equation for thermal equilibrium equals exactly the PDE for electrostatics:

$$-\nabla \cdot (k \cdot \nabla T) = h_V \tag{18.26}$$

with k being the heat conductivity and h_V being the volume heating in W/m^3. Solving this equation goes the same way as described for electrostatics.
The heat equation for non-equilibrium writes as follows:

$$\rho \cdot c_p \cdot \frac{\partial T}{\partial t} - \nabla \cdot (k \cdot \nabla T) = h_V \tag{18.27}$$

where ρ is mass density in kg/m^3, c_p is specific heat capacity in J/kg/K and t is time.

18.3.4 Static Current Conduction

The equation for static current conduction equals exactly the PDE for electrostatics:

$$-\nabla \cdot (\sigma \cdot \nabla V) = \frac{\partial q_V}{\partial t} \tag{18.28}$$

with σ being the electric conductivity and q_V being the volume charge density. Solving this equation goes the same way as described for electrostatics.

18.3.5 Electromagnetic Waves

Im microwave engineering, field solvers are often used to calculate the RF behaviour of a specific structure [89]. Thus, it is used to solve the vector wave equation for the complex-valued, time-harmonic electric field $\vec{E}$ (i.e. in frequency domain with $\partial/\partial t = j\omega$):

$$\omega^2 \cdot \epsilon \cdot \mu \cdot \vec{E} + \nabla^2 \vec{E} - \nabla \cdot (\nabla \vec{E}) = j \cdot \omega \cdot \mu \cdot \vec{J} + \nabla \times \vec{M} \tag{18.29}$$

Note that in source-free regions it is $\nabla \vec{E} = 0$, $\vec{J} = 0$ and $\vec{M} = 0$. In order to find a solution to this equation, boundary conditions are required:

$$\vec{n}_1 \times (\vec{E}_1 - \vec{E}_2) = 0 \tag{18.30}$$

$$\vec{n}_1 \cdot (\epsilon_1 \cdot \vec{E}_1 - \epsilon_2 \cdot \vec{E}_2) = \rho_s \tag{18.31}$$

Note again that in source-free regions it is $\rho_s = 0$. By resolving the Laplace operator, the differential equation 18.29 can also be written as:

$$\omega^2 \cdot \epsilon \cdot \mu \cdot \vec{E} - \nabla \times \nabla \times \vec{E} = j \cdot \omega \cdot \mu \cdot \vec{J} + \nabla \times \vec{M} \tag{18.32}$$

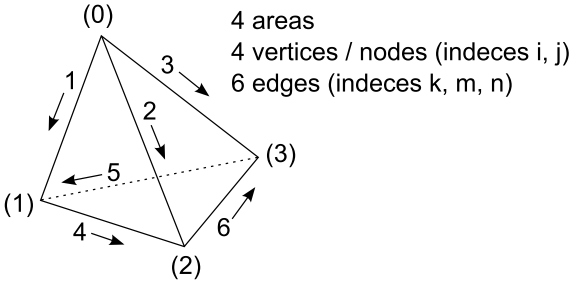


Figure 18.5: tetrahedron with node and edge numbering

For 3D vector quantities the prefered discretization use tetrahedral elements (see figure 18.5) with edge-based shape function (TVFE - tangential vector finite elements). That is, the computational domain is divided into small tetrahedra. The unknowns in vector x of the matrix equation are the average values of the electric field E_k in parallel to the edges of these tetrahedral-shaped elements. The electric field $\vec{E}$ within them is interpolated with a so-called shape (or basis) function $\vec{W}_k$, which is non-zero within its element only:

$$\vec{E} = \sum_{k=1}^{6} E_k \cdot \vec{W}_k \tag{18.33}$$

This way, the electric field can be expressed in the matrix equation with one unknown for each edge in the mesh. The basic concept of FEM is to cast the PDE into its weak form:

$$\iiint PDE(\vec{r}) \cdot \vec{W}(\vec{r})\, dv \tag{18.34}$$

For equation 18.32 this gives:

$$\iiint \omega^2 \cdot \epsilon \cdot \mu \cdot \vec{E} \cdot \vec{W} - \vec{W} \cdot \nabla \times \nabla \times \vec{E}\, dv = \iiint j \cdot \omega \cdot \mu \cdot \vec{J} \cdot \vec{W} + \vec{W} \cdot \nabla \times \vec{M}\, dv \tag{18.35}$$

That is, the PDE is not enforced everywhere. Instead a weighted average (over each element) is solved. Most often, the shape function from equation 18.33 is used as weighting function $\vec{W}(\vec{r})$ (Galerkin's method). This finally gives the following matrix equation:

$$(\boldsymbol{S} - \epsilon_0 \cdot \mu_0 \cdot \omega^2 \cdot \boldsymbol{T} - \boldsymbol{B}) \cdot E = J \tag{18.36}$$

with:

$$\text{"stiffness"} \quad \boldsymbol{S}_{nm} = \iiint \frac{1}{\mu_r} \cdot (\nabla \times \vec{W}_n) \cdot (\nabla \times \vec{W}_m)\, dv \tag{18.37}$$

$$\text{"mass"} \quad \boldsymbol{T}_{nm} = \iiint \epsilon_r \cdot \vec{W}_n \cdot \vec{W}_m\, dv \tag{18.38}$$

$$\text{boundaries} \quad \boldsymbol{B}_n = j\omega\mu_0 \cdot \left(\oiint \vec{W}_n \cdot (\vec{n} \times \vec{H})\, dS - \iint \sigma_S \cdot (\vec{n} \times \vec{W}_n) \cdot (\vec{n} \times \vec{W}_m)\, dS \right) \tag{18.39}$$

$$\text{sources} \quad \boldsymbol{J}_n = -\iiint \left(j\omega\mu_0 \cdot \vec{J}_n + \frac{1}{\mu_r} \cdot \nabla \times \vec{M}_n \right) \cdot \vec{W}_n\, dv \tag{18.40}$$

with σ_S being the surface conductivity (or admittance) at a boundary. Each edge of the mesh gets a row and a column within the matrices S, T and B. In order to create the matrices S and T, a tetrahedral element inserts the 6×6 coefficients of its edges at the corresponding position. Thus, a matrix entry is non-zero only if it belongs to two edges that share the same finite element. The overall matrix is very sparse therefore.

If now the coefficients of the wave equation (i.e. the material properties ϵ_r and μ_r) are kept constant within an element, the integrals 18.37 and 18.38 can be solved in advance. With first order shape functions applied to a tetrahedron the results are [90]:

$$\boldsymbol{S}_{nm} = \frac{1}{9 \cdot V \cdot \mu_r} \cdot [\vec{b}_n \cdot \vec{b}_m] \qquad (n = 0 \ldots 5, \quad m = 0 \ldots 5) \tag{18.41}$$

$$\text{element volume} \quad V = \frac{1}{6} \cdot |(\vec{r}_1 - \vec{r}_0) \cdot ((\vec{r}_2 - \vec{r}_0) \times (\vec{r}_3 - \vec{r}_0))| \tag{18.42}$$

$$\text{edge vector} \quad \vec{b}_k = \vec{b}_{ij} = \vec{r}_j - \vec{r}_i \tag{18.43}$$

where $\vec{r}_i$ is the position vector of node i. Furthermore:

$$\boldsymbol{T} = \frac{\epsilon_r}{180 \cdot V} \cdot [\boldsymbol{T}_{nm}] \qquad (n = 0 \ldots 5, \quad m = 0 \ldots 5) \tag{18.44}$$

$$\text{with} \quad T_{00} = 2 \cdot (\vec{A}_0 \cdot \vec{A}_0 - \vec{A}_0 \cdot \vec{A}_1 + \vec{A}_1 \cdot \vec{A}_1) \tag{18.45}$$

$$T_{11} = 2 \cdot (\vec{A}_0 \cdot \vec{A}_0 - \vec{A}_0 \cdot \vec{A}_2 + \vec{A}_2 \cdot \vec{A}_2) \tag{18.46}$$

$$T_{22} = 2 \cdot (\vec{A}_0 \cdot \vec{A}_0 - \vec{A}_0 \cdot \vec{A}_3 + \vec{A}_3 \cdot \vec{A}_3) \tag{18.47}$$

$$T_{33} = 2 \cdot (\vec{A}_1 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_2 + \vec{A}_2 \cdot \vec{A}_2) \tag{18.48}$$

$$T_{44} = 2 \cdot (\vec{A}_1 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_3 + \vec{A}_3 \cdot \vec{A}_3) \tag{18.49}$$

$$T_{55} = 2 \cdot (\vec{A}_2 \cdot \vec{A}_2 - \vec{A}_2 \cdot \vec{A}_3 + \vec{A}_3 \cdot \vec{A}_3) \tag{18.50}$$

$$T_{10} = T_{01} = \vec{A}_0 \cdot \vec{A}_0 - \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_0 \cdot \vec{A}_2 + 2 \cdot \vec{A}_1 \cdot \vec{A}_2 \tag{18.51}$$

$$T_{20} = T_{02} = \vec{A}_0 \cdot \vec{A}_0 - \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_0 \cdot \vec{A}_3 + 2 \cdot \vec{A}_1 \cdot \vec{A}_2 \tag{18.52}$$

$$T_{21} = T_{12} = \vec{A}_0 \cdot \vec{A}_0 - \vec{A}_0 \cdot \vec{A}_2 - \vec{A}_0 \cdot \vec{A}_3 + 2 \cdot \vec{A}_2 \cdot \vec{A}_3 \tag{18.53}$$

$$T_{30} = T_{03} = \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_1 - 2 \cdot \vec{A}_0 \cdot \vec{A}_2 + \vec{A}_1 \cdot \vec{A}_2 \tag{18.54}$$

$$T_{31} = T_{13} = 2 \cdot \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_2 - \vec{A}_0 \cdot \vec{A}_2 + \vec{A}_2 \cdot \vec{A}_2 \tag{18.55}$$

$$T_{32} = T_{23} = \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_3 - \vec{A}_0 \cdot \vec{A}_2 + \vec{A}_2 \cdot \vec{A}_3 \tag{18.56}$$

$$T_{40} = T_{04} = \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_1 - 2 \cdot \vec{A}_0 \cdot \vec{A}_3 + \vec{A}_1 \cdot \vec{A}_3 \tag{18.57}$$

$$T_{41} = T_{14} = \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_2 - \vec{A}_0 \cdot \vec{A}_3 + \vec{A}_2 \cdot \vec{A}_3 \tag{18.58}$$

$$T_{42} = T_{24} = 2 \cdot \vec{A}_0 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_3 - \vec{A}_0 \cdot \vec{A}_3 + \vec{A}_3 \cdot \vec{A}_3 \tag{18.59}$$

$$T_{43} = T_{34} = \vec{A}_1 \cdot \vec{A}_1 - \vec{A}_1 \cdot \vec{A}_2 - \vec{A}_1 \cdot \vec{A}_3 + 2 \cdot \vec{A}_2 \cdot \vec{A}_3 \tag{18.60}$$

$$T_{50} = T_{05} = \vec{A}_0 \cdot \vec{A}_2 - \vec{A}_1 \cdot \vec{A}_2 - \vec{A}_0 \cdot \vec{A}_3 + \vec{A}_1 \cdot \vec{A}_3 \tag{18.61}$$

$$T_{51} = T_{15} = \vec{A}_0 \cdot \vec{A}_2 - \vec{A}_2 \cdot \vec{A}_2 - 2 \cdot \vec{A}_0 \cdot \vec{A}_3 + \vec{A}_2 \cdot \vec{A}_3 \tag{18.62}$$

$$T_{52} = T_{25} = 2 \cdot \vec{A}_0 \cdot \vec{A}_2 - \vec{A}_2 \cdot \vec{A}_3 - \vec{A}_0 \cdot \vec{A}_3 + \vec{A}_3 \cdot \vec{A}_3 \tag{18.63}$$

$$T_{53} = T_{35} = \vec{A}_1 \cdot \vec{A}_2 - \vec{A}_2 \cdot \vec{A}_2 - 2 \cdot \vec{A}_1 \cdot \vec{A}_3 + \vec{A}_2 \cdot \vec{A}_3 \tag{18.64}$$

$$T_{54} = T_{45} = 2 \cdot \vec{A}_1 \cdot \vec{A}_2 - \vec{A}_2 \cdot \vec{A}_3 - \vec{A}_1 \cdot \vec{A}_3 + \vec{A}_3 \cdot \vec{A}_3 \tag{18.65}$$

where $\vec{A}_i$ is the area normal of the triangle lying opposite to node i and pointing to the inside of the tetrahedron, e.g.

$$\vec{A}_0 = \frac{1}{2} \cdot (\vec{r}_2 - \vec{r}_1) \times (\vec{r}_3 - \vec{r}_1) \tag{18.66}$$

18.4 FE-BI Method

The finite element boundary integral (FE-BI) method is a hybrid technique that uses FEM (for volume electric field) and MoM (for surface electric and magnetic field). This way the advantages of both methods are combined and create one of the most powerful techniques in computational electromagnetics. Anyway, a disadvantage of the MoM part is its dense matrix structure.

Chapter 19

Mathematical background

19.1 N-port matrix conversions

When dealing with n-port parameters it may be necessary or convenient to convert them into other matrix representations. All equations in this section uses the definition of power waves. The following matrices and notations are used in the transformation equations.

$$
\begin{aligned}
\left[\underline{X}\right]^{-1} &= \text{inverted matrix of } \left[\underline{X}\right] \\
\left[\underline{X}\right]^{*} &= \text{complex conjugated matrix of } \left[\underline{X}\right] \\
\left[\underline{E}\right] &= \begin{bmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1 \end{bmatrix} \text{identity matrix} \\
\left[\underline{S}\right] &= \text{S-parameter matrix} \\
\left[\underline{Z}\right] &= \text{impedance matrix} \\
\left[\underline{Y}\right] &= \text{admittance matrix} \\
\left[\underline{Z}_{ref}\right] &= \begin{pmatrix} \underline{Z}_{0,1} & 0 & \dots & 0 \\ 0 & \underline{Z}_{0,2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \underline{Z}_{0,N} \end{pmatrix} \\
\underline{Z}_{0,n} &= \text{reference impedance of port } n \\
\left[G_{ref}\right] &= \begin{pmatrix} G_1 & 0 & \dots & 0 \\ 0 & G_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & G_N \end{pmatrix} \\
G_n &= \frac{1}{\sqrt{\text{Re}\left|\underline{Z}_{0,n}\right|}}
\end{aligned}
$$

19.1.1 Renormalization of S-parameters to different port impedances

During S-parameter usage it sometimes appears to have not all components in a circuit normalized to the same impedance. In the field of RF techniques, it is usually 50Ω. In order to transform the S-parameter matrix $[\underline{S}]$ to the one $[\underline{S'}]$ with different port impedances, the following computation must be applied.

$$[\underline{S'}] = [\underline{A}]^{-1} \cdot \left([\underline{S}] - [\underline{R}]^*\right) \cdot \left([\underline{E}] - [\underline{R}] \cdot [\underline{S}]\right)^{-1} \cdot [\underline{A}]^* \tag{19.1}$$

With

$$Z_n = \text{reference impedance of port } n \text{ after the normalizing process}$$

$$Z_{n,before} = \text{reference impedance of port } n \text{ before the normalizing process}$$

$$[\underline{R}] = \begin{pmatrix} \underline{r}_1 & 0 & \dots & 0 \\ 0 & \underline{r}_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \underline{r}_N \end{pmatrix} \text{reflection coefficient matrix}$$

$$\underline{r}_n = \frac{Z_n - Z_{n,before}^*}{Z_n + Z_{n,before}}$$

$$[\underline{A}] = \begin{pmatrix} A_1 & 0 & \dots & 0 \\ 0 & A_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & A_N \end{pmatrix}$$

$$A_n = \frac{1-\underline{r}_n^*}{|1-\underline{r}_n|} \cdot \sqrt{|1 - \underline{r}_n \cdot \underline{r}_n^*|}$$

$$A_n = \frac{\sqrt{Z_n \cdot Z_{n,before}}}{Z_n + Z_{n,before}} \text{ for real-valued reference impedances}$$

19.1.2 Transformations of n-Port matrices

S-parameter, admittance and impedance matrices are not limited to one- or two-port definitions. They are defined for an arbitrary number of ports. The following section contains transformation formulas forth and back each matrix representation.

Converting a scattering parameter matrix to an impedance matrix is done by the following formula.

$$[\underline{Z}] = \left[\underline{G}_{ref}\right]^{-1} \cdot \left([\underline{E}] - [\underline{S}]\right)^{-1} \cdot \left([\underline{S}] \cdot \left[\underline{Z}_{ref}\right] + \left[\underline{Z}_{ref}\right]^*\right) \cdot \left[\underline{G}_{ref}\right] \tag{19.2}$$

Converting a scattering parameter matrix to an admittance matrix can be achieved by computing the following formula.

$$[\underline{Y}] = \left[\underline{G}_{ref}\right]^{-1} \cdot \left([\underline{S}] \cdot \left[\underline{Z}_{ref}\right] + \left[\underline{Z}_{ref}\right]^*\right)^{-1} \cdot \left([\underline{E}] - [\underline{S}]\right) \cdot \left[\underline{G}_{ref}\right] \tag{19.3}$$

Converting an impedance matrix to a scattering parameter matrix is done by th following formula.

$$[\underline{S}] = \left[\underline{G}_{ref}\right] \cdot \left([\underline{Z}] - \left[\underline{Z}_{ref}\right]^*\right) \cdot \left([\underline{Z}] + \left[\underline{Z}_{ref}\right]\right)^{-1} \cdot \left[\underline{G}_{ref}\right]^{-1} \tag{19.4}$$

Converting an admittance matrix to a scattering parameter matrix is done by the following formula.

$$[\underline{S}] = \left[\underline{G}_{ref}\right] \cdot \Big([\underline{E}] - \left[\underline{Z}_{ref}\right]^{*} \cdot [\underline{Y}]\Big) \cdot \left([\underline{E}] + \left[\underline{Z}_{ref}\right] \cdot [\underline{Y}]\right)^{-1} \cdot \left[\underline{G}_{ref}\right]^{-1} \tag{19.5}$$

Converting an impedance matrix to an admittance matrix is done by the following simple formula.

$$[\underline{Y}] = [\underline{Z}]^{-1} \tag{19.6}$$

Converting an admittance matrix to an impedance matrix is done by the following simple formula.

$$[\underline{Z}] = [\underline{Y}]^{-1} \tag{19.7}$$

19.1.3 Two-Port transformations

Two-Port matrix conversion based on current and voltage

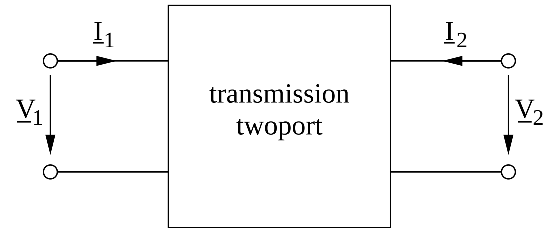


Figure 19.1: twoport definition using current and voltage

There are five different matrix forms for the correlations between the quantities at the transmission twoport shown in fig. 19.1, each having its special meaning when connecting twoports with each other.

- Y-parameters (also called admittance parameters)

$$\begin{pmatrix}\underline{I}_1\\ \underline{I}_2\end{pmatrix} = \begin{pmatrix}\underline{Y}_{11} & \underline{Y}_{12}\\ \underline{Y}_{21} & \underline{Y}_{22}\end{pmatrix} \cdot \begin{pmatrix}\underline{V}_1\\ \underline{V}_2\end{pmatrix} \tag{19.8}$$

- Z-parameters (also called impedance parameters)

$$\begin{pmatrix}\underline{V}_1\\ \underline{V}_2\end{pmatrix} = \begin{pmatrix}\underline{Z}_{11} & \underline{Z}_{12}\\ \underline{Z}_{21} & \underline{Z}_{22}\end{pmatrix} \cdot \begin{pmatrix}\underline{I}_1\\ \underline{I}_2\end{pmatrix} \tag{19.9}$$

- H-parameters (also called hybrid parameters)

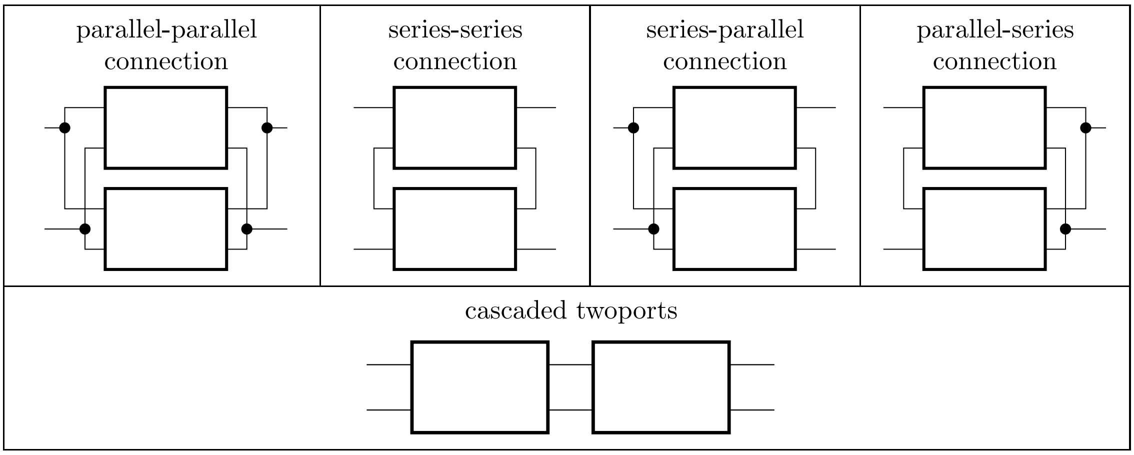
$$\begin{pmatrix}\underline{V}_1\\ \underline{I}_2\end{pmatrix} = \begin{pmatrix}\underline{H}_{11} & \underline{H}_{12}\\ \underline{H}_{21} & \underline{H}_{22}\end{pmatrix} \cdot \begin{pmatrix}\underline{I}_1\\ \underline{V}_2\end{pmatrix} \tag{19.10}$$

- G-parameters (also called C-parameters)

$$\begin{pmatrix}\underline{I}_1\\ \underline{V}_2\end{pmatrix} = \begin{pmatrix}\underline{G}_{11} & \underline{G}_{12}\\ \underline{G}_{21} & \underline{G}_{22}\end{pmatrix} \cdot \begin{pmatrix}\underline{V}_1\\ \underline{I}_2\end{pmatrix} \tag{19.11}$$

- A-parameters (also called chain or ABCD-parameters)

$$\begin{pmatrix}\underline{V}_1\\ \underline{I}_1\end{pmatrix} = \begin{pmatrix}\underline{A}_{11} & \underline{A}_{12}\\ \underline{A}_{21} & \underline{A}_{22}\end{pmatrix} \cdot \begin{pmatrix}\underline{V}_2\\ -\underline{I}_2\end{pmatrix} \tag{19.12}$$


parallel-parallel connection
series-series connection
series-parallel connection
parallel-series connection
cascaded twoports

Basically there are five different kinds of twoport connections. Using the corresponding twoport matrix representations, complicated networks can be analysed by connecting elementary twoports. The linear correlations between the complex currents and voltages rms values of a twoport are described by four complex twoport parameters (i.e. the twoport matrix). These parameters are used to describe the AC behaviour of the twoport.

- parallel-parallel connection: use Y-parameters: $Y = Y_1 + Y_2$
- series-series connection: use Z-parameters: $Z = Z_1 + Z_2$
- series-parallel connection: use H-parameters: $H = H_1 + H_2$
- parallel-series connection: use G-parameters: $G = G_1 + G_2$
- chain connection: use A-parameters: $A = A_1 \cdot A_2$

	A	Y	Z	H	G
A	$\begin{matrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{matrix}$	$\begin{matrix} \frac{-Y_{22}}{Y_{21}} & \frac{-1}{Y_{21}} \\ \frac{-\Delta Y}{Y_{21}} & \frac{-Y_{11}}{Y_{21}} \end{matrix}$	$\begin{matrix} \frac{Z_{11}}{Z_{21}} & \frac{\Delta Z}{Z_{21}} \\ \frac{1}{Z_{21}} & \frac{Z_{22}}{Z_{21}} \end{matrix}$	$\begin{matrix} \frac{-\Delta H}{H_{21}} & \frac{-H_{11}}{H_{21}} \\ \frac{-H_{22}}{H_{21}} & \frac{-1}{H_{21}} \end{matrix}$	$\begin{matrix} \frac{1}{G_{21}} & \frac{G_{22}}{G_{21}} \\ \frac{G_{11}}{G_{21}} & \frac{\Delta G}{G_{21}} \end{matrix}$
Y	$\begin{matrix} \frac{A_{22}}{A_{12}} & \frac{-\Delta A}{A_{12}} \\ \frac{-1}{A_{12}} & \frac{A_{11}}{A_{12}} \end{matrix}$	$\begin{matrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{matrix}$	$\begin{matrix} \frac{Z_{22}}{\Delta Z} & \frac{-Z_{12}}{\Delta Z} \\ \frac{-Z_{21}}{\Delta Z} & \frac{Z_{11}}{\Delta Z} \end{matrix}$	$\begin{matrix} \frac{1}{H_{11}} & \frac{-H_{12}}{H_{11}} \\ \frac{H_{21}}{H_{11}} & \frac{\Delta H}{H_{11}} \end{matrix}$	$\begin{matrix} \frac{\Delta G}{G_{22}} & \frac{G_{12}}{G_{22}} \\ \frac{-G_{21}}{G_{22}} & \frac{1}{G_{22}} \end{matrix}$
Z	$\begin{matrix} \frac{A_{11}}{A_{21}} & \frac{\Delta A}{A_{21}} \\ \frac{1}{A_{21}} & \frac{A_{22}}{A_{21}} \end{matrix}$	$\begin{matrix} \frac{Y_{22}}{\Delta Y} & \frac{-Y_{12}}{\Delta Y} \\ \frac{-Y_{21}}{\Delta Y} & \frac{Y_{11}}{\Delta Y} \end{matrix}$	$\begin{matrix} Z_{11} & Z_{12} \\ Z_{21} & Z_{22} \end{matrix}$	$\begin{matrix} \frac{\Delta H}{H_{22}} & \frac{H_{12}}{H_{22}} \\ \frac{-H_{21}}{H_{22}} & \frac{1}{H_{22}} \end{matrix}$	$\begin{matrix} \frac{1}{G_{11}} & \frac{-G_{12}}{G_{11}} \\ \frac{G_{21}}{G_{11}} & \frac{\Delta G}{G_{11}} \end{matrix}$
H	$\begin{matrix} \frac{A_{12}}{A_{22}} & \frac{\Delta A}{A_{22}} \\ \frac{-1}{A_{22}} & \frac{A_{21}}{A_{22}} \end{matrix}$	$\begin{matrix} \frac{1}{Y_{11}} & \frac{-Y_{12}}{Y_{11}} \\ \frac{Y_{21}}{Y_{11}} & \frac{\Delta Y}{Y_{11}} \end{matrix}$	$\begin{matrix} \frac{\Delta Z}{Z_{22}} & \frac{Z_{12}}{Z_{22}} \\ \frac{-Z_{21}}{Z_{22}} & \frac{1}{Z_{22}} \end{matrix}$	$\begin{matrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{matrix}$	$\begin{matrix} \frac{G_{22}}{\Delta G} & \frac{-G_{12}}{\Delta G} \\ \frac{-G_{21}}{\Delta G} & \frac{G_{11}}{\Delta G} \end{matrix}$
G	$\begin{matrix} \frac{A_{21}}{A_{11}} & \frac{-\Delta A}{A_{11}} \\ \frac{1}{A_{11}} & \frac{A_{12}}{A_{11}} \end{matrix}$	$\begin{matrix} \frac{\Delta Y}{Y_{22}} & \frac{Y_{12}}{Y_{22}} \\ \frac{-Y_{21}}{Y_{22}} & \frac{1}{Y_{22}} \end{matrix}$	$\begin{matrix} \frac{1}{Z_{11}} & \frac{-Z_{12}}{Z_{11}} \\ \frac{Z_{21}}{Z_{11}} & \frac{\Delta Z}{Z_{11}} \end{matrix}$	$\begin{matrix} \frac{H_{22}}{\Delta H} & \frac{-H_{12}}{\Delta H} \\ \frac{-H_{21}}{\Delta H} & \frac{H_{11}}{\Delta H} \end{matrix}$	$\begin{matrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{matrix}$

Two-Port matrix conversion based on signal waves

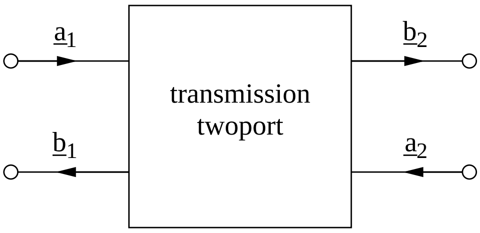


Figure 19.2: twoport definition using signal waves

There are two different matrix forms for the correlations between the quantities at the transmission twoport shown in fig. 19.2.

- S-parameters (also called scattering parameters)

$$\begin{pmatrix} \underline{b}_1 \\ \underline{b}_2 \end{pmatrix} = \begin{pmatrix} \underline{S}_{11} & \underline{S}_{12} \\ \underline{S}_{21} & \underline{S}_{22} \end{pmatrix} \cdot \begin{pmatrix} \underline{a}_1 \\ \underline{a}_2 \end{pmatrix} \tag{19.13}$$

- T-parameters (also called transfer scattering parameters)

$$\begin{pmatrix} \underline{b}_1 \\ \underline{a}_1 \end{pmatrix} = \begin{pmatrix} \underline{T}_{11} & \underline{T}_{12} \\ \underline{T}_{21} & \underline{T}_{22} \end{pmatrix} \cdot \begin{pmatrix} \underline{a}_2 \\ \underline{b}_2 \end{pmatrix} \tag{19.14}$$

Note that this is only one of two different definitions for T-parameters appearing in literature! When connecting cascaded twoports it is possible to compute the resulting transfer scattering parameters by the following equation.

$$T = T_1 \cdot T_2 \tag{19.15}$$

According to Janusz A. Dobrowolski [91] the following table contains the matrix transformation formulas.

	S	T
S	$\begin{matrix} S_{11} & S_{12} \\ S_{21} & S_{22} \end{matrix}$	$\begin{matrix} \dfrac{T_{12}}{T_{22}} & \dfrac{\Delta T}{T_{22}} \\ \dfrac{1}{T_{22}} & \dfrac{-T_{21}}{T_{22}} \end{matrix}$
T	$\begin{matrix} \dfrac{-\Delta S}{S_{21}} & \dfrac{S_{11}}{S_{21}} \\ \dfrac{-S_{22}}{S_{21}} & \dfrac{1}{S_{21}} \end{matrix}$	$\begin{matrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{matrix}$

Mixed Two-Port matrix conversions

Sometimes it may be useful to have a twoport matrix representation based on signal waves in a representation based on voltage and current and the other way around. There are two more parameters involved in this case: The reference impedance at port 1 (denoted as Z_1) and the reference impedance at port 2 (denoted as Z_2).

Converting from scattering parameters to chain parameters results in

$$A_{11} = \frac{(1 - S_{22}) \cdot (Z_1^* + Z_1 \cdot S_{11}) + Z_1 \cdot S_{12} \cdot S_{21}}{2 \cdot S_{21} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}} \tag{19.16}$$

$$A_{12} = \frac{(Z_1^* + Z_1 \cdot S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) - Z_1 \cdot Z_2 \cdot S_{12} \cdot S_{21}}{2 \cdot S_{21} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}} \tag{19.17}$$

$$A_{21} = \frac{(1 - S_{11}) \cdot (1 - S_{22}) - S_{12} \cdot S_{21}}{2 \cdot S_{21} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}} \tag{19.18}$$

$$A_{22} = \frac{(1 - S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_2 \cdot S_{12} \cdot S_{21}}{2 \cdot S_{21} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}} \tag{19.19}$$

Converting from chain parameters to scattering parameters results in

$$S_{11} = \frac{A_{11} \cdot Z_2 + A_{12} - A_{21} \cdot Z_1^* \cdot Z_2 - A_{22} \cdot Z_1^*}{A_{11} \cdot Z_2 + A_{12} + A_{21} \cdot Z_1 \cdot Z_2 + A_{22} \cdot Z_1} \tag{19.20}$$

$$S_{12} = \frac{\Delta A \cdot 2 \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}}{A_{11} \cdot Z_2 + A_{12} + A_{21} \cdot Z_1 \cdot Z_2 + A_{22} \cdot Z_1} \tag{19.21}$$

$$S_{21} = \frac{2 \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}}{A_{11} \cdot Z_2 + A_{12} + A_{21} \cdot Z_1 \cdot Z_2 + A_{22} \cdot Z_1} \tag{19.22}$$

$$S_{22} = \frac{-A_{11} \cdot Z_2^* + A_{12} - A_{21} \cdot Z_1 \cdot Z_2^* + A_{22} \cdot Z_1}{A_{11} \cdot Z_2 + A_{12} + A_{21} \cdot Z_1 \cdot Z_2 + A_{22} \cdot Z_1} \tag{19.23}$$

Converting from scattering parameters to hybrid parameters results in

$$H_{11} = \frac{(Z_1^* + Z_1 \cdot S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) - Z_1 \cdot Z_2 \cdot S_{12} \cdot S_{21}}{(1 - S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.24}$$

$$H_{12} = \frac{2 \cdot S_{12} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}}{(1 - S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.25}$$

$$H_{21} = \frac{-2 \cdot S_{21} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}}{(1 - S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.26}$$

$$H_{22} = \frac{(1 - S_{11}) \cdot (1 - S_{22}) - S_{12} \cdot S_{21}}{(1 - S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.27}$$

Converting from hybrid parameters to scattering parameters results in

$$S_{11} = \frac{(H_{11} - Z_1^*) \cdot (1 + Z_2 \cdot H_{22}) - Z_2 \cdot H_{12} \cdot H_{21}}{(H_{11} + Z_1) \cdot (1 + Z_2 \cdot H_{22}) - Z_2 \cdot H_{12} \cdot H_{21}} \tag{19.28}$$

$$S_{12} = \frac{2 \cdot H_{12} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}}{(H_{11} + Z_1) \cdot (1 + Z_2 \cdot H_{22}) - Z_2 \cdot H_{12} \cdot H_{21}} \tag{19.29}$$

$$S_{21} = \frac{-2 \cdot H_{21} \cdot \sqrt{Re\,(Z_1) \cdot Re\,(Z_2)}}{(H_{11} + Z_1) \cdot (1 + Z_2 \cdot H_{22}) - Z_2 \cdot H_{12} \cdot H_{21}} \tag{19.30}$$

$$S_{22} = \frac{(H_{11} + Z_1) \cdot (1 - Z_2^* \cdot H_{22}) + Z_2^* \cdot H_{12} \cdot H_{21}}{(H_{11} + Z_1) \cdot (1 + Z_2 \cdot H_{22}) - Z_2 \cdot H_{12} \cdot H_{21}} \tag{19.31}$$

Converting from scattering parameters to the second type of hybrid parameters results in

$$G_{11} = \frac{(1 - S_{11}) \cdot (1 - S_{22}) - S_{12} \cdot S_{21}}{(1 - S_{22}) \cdot (Z_1^* + Z_1 \cdot S_{11}) + Z_1 \cdot S_{12} \cdot S_{21}} \tag{19.32}$$

$$G_{12} = \frac{-2 \cdot S_{12} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(1 - S_{22}) \cdot (Z_1^* + Z_1 \cdot S_{11}) + Z_1 \cdot S_{12} \cdot S_{21}} \tag{19.33}$$

$$G_{21} = \frac{2 \cdot S_{21} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(1 - S_{22}) \cdot (Z_1^* + Z_1 \cdot S_{11}) + Z_1 \cdot S_{12} \cdot S_{21}} \tag{19.34}$$

$$G_{22} = \frac{(Z_1^* + Z_1 \cdot S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_1 \cdot Z_2 \cdot S_{12} \cdot S_{21}}{(1 - S_{22}) \cdot (Z_1^* + Z_1 \cdot S_{11}) + Z_1 \cdot S_{12} \cdot S_{21}} \tag{19.35}$$

Converting from the second type of hybrid parameters to scattering parameters results in

$$S_{11} = \frac{(1 - G_{11} \cdot Z_1^*) \cdot (G_{22} + Z_2) + Z_1^* \cdot G_{12} \cdot G_{21}}{(1 + G_{11} \cdot Z_1) \cdot (G_{22} + Z_2) - Z_1 \cdot G_{12} \cdot G_{21}} \tag{19.36}$$

$$S_{12} = \frac{-2 \cdot G_{12} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(1 + G_{11} \cdot Z_1) \cdot (G_{22} + Z_2) - Z_1 \cdot G_{12} \cdot G_{21}} \tag{19.37}$$

$$S_{21} = \frac{2 \cdot G_{21} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(1 + G_{11} \cdot Z_1) \cdot (G_{22} + Z_2) - Z_1 \cdot G_{12} \cdot G_{21}} \tag{19.38}$$

$$S_{22} = \frac{(1 + G_{11} \cdot Z_1) \cdot (G_{22} - Z_2^*) - Z_1 \cdot G_{12} \cdot G_{21}}{(1 + G_{11} \cdot Z_1) \cdot (G_{22} + Z_2) - Z_1 \cdot G_{12} \cdot G_{21}} \tag{19.39}$$

Converting from scattering parameters to admittance parameters results in

$$Y_{11} = \frac{(1 - S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_2 \cdot S_{12} \cdot S_{21}}{(Z_1^* + Z_1 \cdot S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) - Z_1 \cdot Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.40}$$

$$Y_{12} = \frac{-2 \cdot S_{12} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(Z_1^* + Z_1 \cdot S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) - Z_1 \cdot Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.41}$$

$$Y_{21} = \frac{-2 \cdot S_{21} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(Z_1^* + Z_1 \cdot S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) - Z_1 \cdot Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.42}$$

$$Y_{22} = \frac{(1 - S_{22}) \cdot (Z_1^* + Z_1 \cdot S_{11}) + Z_1 \cdot S_{12} \cdot S_{21}}{(Z_1^* + Z_1 \cdot S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_1 \cdot Z_2 \cdot S_{12} \cdot S_{21}} \tag{19.43}$$

Converting from admittance parameters to scattering parameters results in

$$S_{11} = \frac{(1 - Y_{11} \cdot Z_1^*) \cdot (1 + Y_{22} \cdot Z_2) + Y_{12} \cdot Z_1^* \cdot Y_{21} \cdot Z_2}{(1 + Y_{11} \cdot Z_1) \cdot (1 + Y_{22} \cdot Z_2) - Y_{12} \cdot Z_1 \cdot Y_{21} \cdot Z_2} \tag{19.44}$$

$$S_{12} = \frac{-2 \cdot Y_{12} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(1 + Y_{11} \cdot Z_1) \cdot (1 + Y_{22} \cdot Z_2) - Y_{12} \cdot Z_1 \cdot Y_{21} \cdot Z_2} \tag{19.45}$$

$$S_{21} = \frac{-2 \cdot Y_{21} \cdot \sqrt{Re(Z_1) \cdot Re(Z_2)}}{(1 + Y_{11} \cdot Z_1) \cdot (1 + Y_{22} \cdot Z_2) - Y_{12} \cdot Z_1 \cdot Y_{21} \cdot Z_2} \tag{19.46}$$

$$S_{22} = \frac{(1 + Y_{11} \cdot Z_1) \cdot (1 - Y_{22} \cdot Z_2^*) + Y_{12} \cdot Z_1 \cdot Y_{21} \cdot Z_2^*}{(1 + Y_{11} \cdot Z_1) \cdot (1 + Y_{22} \cdot Z_2) - Y_{12} \cdot Z_1 \cdot Y_{21} \cdot Z_2} \tag{19.47}$$

Converting from scattering parameters to impedance parameters results in

$$Z_{11} = \frac{(Z_1^* + Z_1 \cdot S_{11}) \cdot (1 - S_{22}) + Z_1 \cdot S_{12} \cdot S_{21}}{(1 - S_{11}) \cdot (1 - S_{22}) - S_{12} \cdot S_{21}} \tag{19.48}$$

$$Z_{12} = \frac{2 \cdot S_{12} \cdot \sqrt{Re\left(Z_1\right) \cdot Re\left(Z_2\right)}}{(1 - S_{11}) \cdot (1 - S_{22}) - S_{12} \cdot S_{21}} \tag{19.49}$$

$$Z_{21} = \frac{2 \cdot S_{21} \cdot \sqrt{Re\left(Z_1\right) \cdot Re\left(Z_2\right)}}{(1 - S_{11}) \cdot (1 - S_{22}) - S_{12} \cdot S_{21}} \tag{19.50}$$

$$Z_{22} = \frac{(1 - S_{11}) \cdot (Z_2^* + Z_2 \cdot S_{22}) + Z_2 \cdot S_{12} \cdot S_{21}}{(1 - S_{11}) \cdot (1 - S_{22}) - S_{12} \cdot S_{21}} \tag{19.51}$$

Converting from impedance parameters to scattering parameters results in

$$S_{11} = \frac{(Z_{11} - Z_1^*) \cdot (Z_{22} + Z_2) - Z_{12} \cdot Z_{21}}{(Z_{11} + Z_1) \cdot (Z_{22} + Z_2) - Z_{12} \cdot Z_{21}} \tag{19.52}$$

$$S_{12} = \frac{2 \cdot Z_{12} \cdot \sqrt{Re\left(Z_1\right) \cdot Re\left(Z_2\right)}}{(Z_{11} + Z_1) \cdot (Z_{22} + Z_2) - Z_{12} \cdot Z_{21}} \tag{19.53}$$

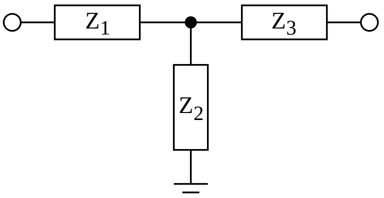
$$S_{21} = \frac{2 \cdot Z_{21} \cdot \sqrt{Re\left(Z_1\right) \cdot Re\left(Z_2\right)}}{(Z_{11} + Z_1) \cdot (Z_{22} + Z_2) - Z_{12} \cdot Z_{21}} \tag{19.54}$$

$$S_{22} = \frac{(Z_{11} + Z_1) \cdot (Z_{22} - Z_2^*) - Z_{12} \cdot Z_{21}}{(Z_{11} + Z_1) \cdot (Z_{22} + Z_2) - Z_{12} \cdot Z_{21}} \tag{19.55}$$

Two-Port parameters of passive devices

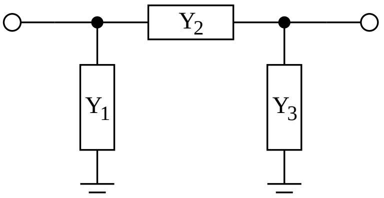
Basically the twoport parameters of passive twoports can be determined using Kirchhoff's voltage law and Kirchhoff's current law or by applying the definition equations of the twoport parameters. This has been done [92] for some example circuits.

- T-topology



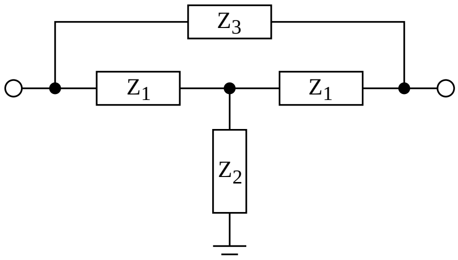
$$Z = \begin{bmatrix} Z_1 + Z_2 & Z_2 \\ Z_2 & Z_2 + Z_3 \end{bmatrix}$$

- π-topology



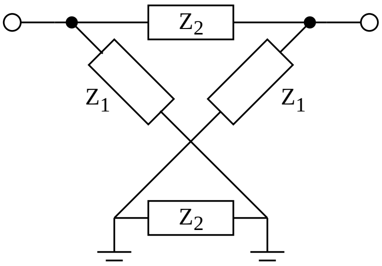
$$Y = \begin{bmatrix} Y_1 + Y_2 & -Y_2 \\ -Y_2 & Y_2 + Y_3 \end{bmatrix}$$

- symmetric T-bridge



$$Z = \begin{bmatrix} \dfrac{Z_1^2 + Z_1 \cdot Z_3}{2 \cdot Z_1 + Z_3} + Z_2 & \dfrac{Z_1^2}{2 \cdot Z_1 + Z_3} + Z_2 \\ \dfrac{Z_1^2}{2 \cdot Z_1 + Z_3} + Z_2 & \dfrac{Z_1^2 + Z_1 \cdot Z_3}{2 \cdot Z_1 + Z_3} + Z_2 \end{bmatrix}$$

- symmetric X-topology



$$Z = \frac{1}{2} \begin{bmatrix} Z_1 + Z_2 & Z_1 - Z_2 \\ Z_1 - Z_2 & Z_1 + Z_2 \end{bmatrix}$$

19.1.4 Calculating Power

A component with y-parameter matrix Y (or z-parameter matrix Z) and node voltage vector U (or node current vector I) consumes the apparent power as follows:

$$\begin{aligned} S &= \frac{1}{2} \cdot U^T \cdot I^* && (19.56) \\ &= \frac{1}{2} \cdot U^T \cdot Y^* \cdot U^* = \frac{1}{2} \sum_m \sum_n U_m \cdot Y_{mn}^* \cdot U_n^* && (19.57) \\ &= \frac{1}{2} \cdot I^T \cdot Z^T \cdot I^* = \frac{1}{2} \sum_m \sum_n I_m \cdot Z_{nm} \cdot I_n^* && (19.58) \\ & && (19.59) \end{aligned}$$

The factor $\frac{1}{2}$ is to convert peak values to effective values. The current is positive if it flows into the component. The real part of S is the power dissipated by the component. If it's negative the component is a source. In Harmonic Balance simulation all harmonics must be summed up (Parseval's theorem). E.g. this means, that if a DC current is also flowing through the component, the total power P_{tot} is the AC power plus the DC power:

$$P_{tot} = \frac{1}{2} \cdot U_{AC}^T \cdot Y^* \cdot U_{AC}^* + U_{DC}^T \cdot Re(Y) \cdot U_{DC} \tag{19.60}$$

The power dissipated by a component with s-parameter matrix S is as follows:

$$P_{diss} = a^{*T} \cdot \left(E - S^{*T} \cdot S\right) \cdot a \tag{19.61}$$

19.2 Solving linear equation systems

When dealing with non-linear networks the number of equation systems to be solved depends on the required precision of the solution and the average necessary iterations until the solution is stable. This emphasizes the meaning of the solving procedures choice for different problems.

The equation systems

$$[A] \cdot [x] = [z] \tag{19.62}$$

solution can be written as

$$[x] = [A]^{-1} \cdot [z] \tag{19.63}$$

19.2.1 Matrix inversion

The elements $\beta_{\mu\nu}$ of the inverse of the matrix A are

$$\beta_{\mu\nu} = \frac{A_{\mu\nu}}{detA} \tag{19.64}$$

whereas $A_{\mu\nu}$ is the matrix elements $a_{\mu\nu}$ cofactor. The cofactor is the sub determinant (i.e. the minor) of the element $a_{\mu\nu}$ multiplied with $(-1)^{\mu+\nu}$. The determinant of a square matrix can be recursively computed by either of the following equations.

$$detA = \sum_{\mu=1}^{n} a_{\mu\nu} \cdot A_{\mu\nu} \quad \text{using the } \nu\text{-th column} \tag{19.65}$$

$$detA = \sum_{\nu=1}^{n} a_{\mu\nu} \cdot A_{\mu\nu} \quad \text{using the } \mu\text{-th row} \tag{19.66}$$

This method is called the Laplace expansion. In order to save computing time the row or column with the most zeros in it is used for the expansion expressed in the above equations. A sub determinant $(n-1)$-th order of a matrix's element $a_{\mu\nu}$ of n-th order is the determinant which is computed by canceling the μ-th row and ν-th column. The following example demonstrates calculating the determinant of a 4th order matrix with the elements of the 3rd row.

$$\begin{vmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{vmatrix} = a_{31} \begin{vmatrix} a_{12} & a_{13} & a_{14} \\ a_{22} & a_{23} & a_{24} \\ a_{42} & a_{43} & a_{44} \end{vmatrix} - a_{32} \begin{vmatrix} a_{11} & a_{13} & a_{14} \\ a_{21} & a_{23} & a_{24} \\ a_{41} & a_{43} & a_{44} \end{vmatrix} \tag{19.67}$$

$$+ a_{33} \begin{vmatrix} a_{11} & a_{12} & a_{14} \\ a_{21} & a_{22} & a_{24} \\ a_{41} & a_{42} & a_{44} \end{vmatrix} - a_{34} \begin{vmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{41} & a_{42} & a_{43} \end{vmatrix}$$

This recursive process for computing the inverse of a matrix is most easiest to be implemented but as well the slowest algorithm. It requires approximately $n!$ operations.

19.2.2 Gaussian elimination

The Gaussian algorithm for solving a linear equation system is done in two parts: forward elimination and backward substitution. During forward elimination the matrix A is transformed into an upper triangular equivalent matrix. Elementary transformations due to an equation system having the same solutions for the unknowns as the original system.

$$A = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \ldots & a_{nn} \end{bmatrix} \rightarrow \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ 0 & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & 0 & a_{nn} \end{bmatrix} \tag{19.68}$$

The modifications applied to the matrix A in order to achieve this transformations are limited to the following set of operations.

- multiplication of a row with a scalar factor
- addition or subtraction of multiples of rows
- exchanging two rows of a matrix

Step 1: Forward elimination

The transformation of the matrix A is done in n − 1 elimination steps. The new matrix elements of the k-th step with k = 1, . . . , n − 1 are computed with the following recursive formulas.

$$a_{ij} = 0 \qquad i = k+1, \ldots, n \text{ and } j = k \tag{19.69}$$

$$a_{ij} = a_{ij} - a_{kj} \cdot a_{ik}/a_{kk} \qquad i = k+1, \ldots, n \text{ and } j = k+1, \ldots, n \tag{19.70}$$

$$z_i = z_i - z_k \cdot a_{ik}/a_{kk} \qquad i = k+1, \ldots, n \tag{19.71}$$

The triangulated matrix can be used to calculate the determinant very easily. The determinant of a triangulated matrix is the product of the diagonal elements. If the determinant $detA$ is non-zero the equation system has a solution. Otherwise the matrix A is singular.

$$detA = a_{11} \cdot a_{22} \cdot \ldots \cdot a_{nn} = \prod_{i=1}^{n} a_{ii} \tag{19.72}$$

When using row and/or column pivoting the resulting determinant may differ in its sign and must be multiplied with $(-1)^m$ whereas m is the number of row and column substitutions.

Finding an appropriate pivot element

The Gaussian elimination fails if the pivot element a_{kk} turns to be zero (division by zero). That is why row and/or column pivoting must be used before each elimination step. If a diagonal element $a_{kk} = 0$, then exchange the pivot row k with the row $m > k$ having the coefficient with the largest absolute value. The new pivot row is m and the new pivot element is going to be a_{mk}. If no such pivot row can be found the matrix is singular.

Total pivoting looks for the element with the largest absolute value within the matrix and exchanges rows and columns. When exchanging columns in equation systems the unknowns get reordered as well. For the numerical solution of equation systems with Gaussian elimination column pivoting is sufficient.

In order to improve numerical stability pivoting should also be applied if $a_{kk} \neq 0$ because division by small diagonal elements propagates numerical (rounding) errors. This appears especially with poorly conditioned (the two dimensional case: two lines with nearly the same slope) equation systems.

Step 2: Backward substitution

The solutions in the vector x are obtained by backward substituting into the triangulated matrix. The elements of the solution vector x are computed by the following recursive equations.

$$x_n = \frac{z_n}{a_{nn}} \tag{19.73}$$

$$x_i = \frac{z_i}{a_{ii}} - \sum_{k=i+1}^{n} x_k \cdot \frac{a_{ik}}{a_{ii}} \qquad i = n-1, \ldots, 1 \tag{19.74}$$

The forward elimination in the Gaussian algorithm requires approximately $n^3/3$, the backward substitution $n^2/2$ operations.

19.2.3 Gauss-Jordan method

The Gauss-Jordan method is a modification of the Gaussian elimination. In each k-th elimination step the elements of the k-th column get zero except the diagonal element which gets 1. When the right hand side vector z is included in each step it contains the solution vector x afterwards.

The following recursive formulas must be applied to get the new matrix elements for the k-th elimination step. The k-th row must be computed first.

$$a_{kj} = a_{kj}/a_{kk} \qquad j = 1 \dots n \tag{19.75}$$

$$z_k = z_k/a_{kk} \tag{19.76}$$

Then the other rows can be calculated with the following formulas.

$$a_{ij} = a_{ij} - a_{ik} \cdot a_{kj} \qquad j = 1, \dots, n \text{ and } i = 1, \dots, n \text{ with } i \neq k \tag{19.77}$$

$$z_i = z_i - a_{ik} \cdot z_k \qquad i = 1, \dots, n \text{ with } i \neq k \tag{19.78}$$

Column pivoting may be necessary in order to avoid division by zero. The solution vector x is not harmed by row substitutions. When the Gauss-Jordan algorithm has been finished the original matrix has been transformed into the identity matrix. If each operation during this process is applied to an identity matrix the resulting matrix is the inverse matrix of the original matrix. This means that the Gauss-Jordan method can be used to compute the inverse of a matrix.

Though this elimination method is easy to implement the number of required operations is larger than within the Gaussian elimination. The Gauss-Jordan method requires approximately $N^3/2 + N^2/2$ operations.

19.2.4 LU decomposition

LU decomposition (decomposition into a lower and upper triangular matrix) is recommended when dealing with equation systems where the matrix A does not alter but the right hand side (the vector z) does. Both the Gaussian elimination and the Gauss-Jordan method involve both the right hand side and the matrix in their algorithm. Consecutive solutions of an equation system with an altering right hand side can be computed faster with LU decomposition.

The LU decomposition splits a matrix A into a product of a lower triangular matrix L with an upper triangular matrix U.

$$A = LU \quad \text{with} \quad L = \begin{bmatrix} l_{11} & 0 & \dots & 0 \\ l_{21} & l_{22} & \ddots & \vdots \\ \vdots & & \ddots & 0 \\ l_{n1} & \dots & \dots & l_{nn} \end{bmatrix} \quad \text{and} \quad U = \begin{bmatrix} u_{11} & u_{12} & \dots & u_{1n} \\ 0 & u_{22} & & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & u_{nn} \end{bmatrix} \tag{19.79}$$

The algorithm for solving the linear equation system $Ax = z$ involves three steps:

- LU decomposition of the coefficient matrix A
  $\rightarrow Ax = LUx = z$
- introduction of an (unknown) arbitrary vector y and solving the equation system $Ly = z$ by forward substitution
  $\rightarrow y = Ux = L^{-1}z$

- solving the equation system $Ux = y$ by backward substitution
  $\rightarrow x = U^{-1}y$

The decomposition of the matrix A into a lower and upper triangular matrix is not unique. The most important decompositions, based on Gaussian elimination, are the Doolittle, the Crout and the Cholesky decomposition.

If pivoting is necessary during these algorithms they do not decompose the matrix A but the product with an arbitrary matrix PA (a permutation of the matrix A). When exchanging rows and columns the order of the unknowns as represented by the vector z changes as well and must be saved during this process for the forward substitution in the algorithms second step.

Step 1: LU decomposition

Using the decomposition according to Crout the coefficients of the L and U matrices can be stored in place the original matrix A. The upper triangular matrix U has the form

$$U = \begin{bmatrix} 1 & u_{12} & \dots & u_{1n} \\ 0 & 1 & & \vdots \\ \vdots & \ddots & \ddots & u_{n-1,n} \\ 0 & \dots & 0 & 1 \end{bmatrix} \tag{19.80}$$

The diagonal elements u_{jj} are ones and thus the determinant $detU$ is one as well. The elements of the new coefficient matrix LU for the k-th elimination step with $k = 1, \dots, n$ compute as follows:

$$u_{jk} = \frac{1}{l_{jj}} \left(a_{jk} - \sum_{r=1}^{j-1} l_{jr} u_{rk} \right) \qquad j = 1, \dots, k-1 \tag{19.81}$$

$$l_{jk} = a_{jk} - \sum_{r=1}^{k-1} l_{jr} u_{rk} \qquad j = k, \dots, n \tag{19.82}$$

Pivoting may be necessary as you are going to divide by the diagonal element l_{jj}.

Step 2: Forward substitution

The solutions in the arbitrary vector y are obtained by forward substituting into the triangulated L matrix. At this stage you need to remember the order of unknowns in the vector z as changed by pivoting. The elements of the solution vector y are computed by the following recursive equation.

$$y_i = \frac{z_i}{l_{ii}} - \sum_{k=1}^{i-1} y_k \cdot \frac{l_{ik}}{l_{ii}} \qquad i = 1, \dots, n \tag{19.83}$$

Step 3: Backward substitution

The solutions in the vector x are obtained by backward substituting into the triangulated U matrix. The elements of the solution vector x are computed by the following recursive equation.

$$x_i = y_i - \sum_{k=i+1}^{n} x_k \cdot u_{ik} \qquad i = n, \dots, 1 \tag{19.84}$$

The division by the diagonal elements of the matrix U is not necessary because of Crouts definition in eq. (19.80) with $u_{ii} = 1$.

The LU decomposition requires approximately $n^3/3 + n^2 - n/3$ operations for solving a linear equation system. For M consecutive solutions the method requires $n^3/3 + Mn^2 - n/3$ operations.

19.2.5 QR decomposition

Singular matrices actually having a solution are over- or under-determined. These types of matrices can be handled by three different types of decompositions: Householder, Jacobi (Givens rotation) and singular value decomposition. Householder decomposition factors a matrix A into the product of an orthonormal matrix Q and an upper triangular matrix R, such that:

$$A = Q \cdot R \tag{19.85}$$

The Householder decomposition is based on the fact that for any two different vectors, v and w, with $\|v\| = \|w\|$, i.e. different vectors of equal length, a reflection matrix H exists such that:

$$H \cdot v = w \tag{19.86}$$

To obtain the matrix H, the vector u is defined by:

$$u = \frac{v - w}{\|v - w\|} \tag{19.87}$$

The matrix H defined by

$$H = I - 2 \cdot u \cdot u^T \tag{19.88}$$

is then the required reflection matrix.

The equation system

$$A \cdot x = z \quad \text{is transformed into} \quad QR \cdot x = z \tag{19.89}$$

With $Q^T \cdot Q = I$ this yields

$$Q^T QR \cdot x = Q^T z \quad \rightarrow \quad R \cdot x = Q^T z \tag{19.90}$$

Since R is triangular the equation system is solved by a simple matrix-vector multiplication on the right hand side and backward substitution.

Step 1: QR decomposition

Starting with $A_1 = A$, let $v_1 =$ the first column of A_1, and $w_1^T = (\pm\|v_1\|, 0, \ldots 0)$, i.e. a column vector whose first component is the norm of v_1 with the remaining components equal to 0. The Householder transformation $H_1 = I - 2 \cdot u_1 \cdot u_1^T$ with $u_1 = v_1 - w_1 / \|v_1 - w_1\|$ will turn the first column of A_1 into w_1 as with $H_1 \cdot A_1 = A_2$. At each stage k, $v_k =$ the kth column of A_k on and below the diagonal with all other components equal to 0, and w_k's kth component equals the norm of v_k with all other components equal to 0. Letting $H_k \cdot A_k = A_{k+1}$, the components of the kth column of A_{k+1} below the diagonal are each 0. These calculations are listed below for each stage for the matrix A.

$$\begin{gathered}
v_1 = \begin{bmatrix} a_{11} \\ a_{21} \\ \vdots \\ a_{n1} \end{bmatrix} \quad w_1 = \begin{bmatrix} \pm\|v_1\| \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad u_1 = \frac{v_1 - w_1}{\|v_1 - w_1\|} = \begin{bmatrix} u_{11} \\ u_{21} \\ \vdots \\ u_{n1} \end{bmatrix} \\
H_1 = I - 2 \cdot u_1 \cdot u_1^T \quad \rightarrow \quad H_1 \cdot A_1 = A_2 = \begin{bmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ 0 & a_{22} & \ldots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & a_{n2} & \ldots & a_{nn} \end{bmatrix}
\end{gathered} \tag{19.91}$$

With this first step the upper left diagonal element of the R matrix, $a_{11} = \pm\|v_1\|$, has been generated. The elements below are zeroed out. Since H_1 can be generated from u_1 stored in place of the first column of A_1 the multiplication $H_1 \cdot A_1$ can be performed without actually generating H_1.

$$v_2 = \begin{bmatrix} 0 \\ a_{22} \\ \vdots \\ a_{n2} \end{bmatrix} \quad w_1 = \begin{bmatrix} 0 \\ \pm\|v_2\| \\ \vdots \\ 0 \end{bmatrix} \quad u_2 = \frac{v_2 - w_2}{\|v_2 - w_2\|} = \begin{bmatrix} 0 \\ u_{22} \\ \vdots \\ u_{n2} \end{bmatrix}$$
$$H_2 = I - 2 \cdot u_2 \cdot u_2^T \quad \rightarrow \quad H_2 \cdot A_2 = A_3 = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ 0 & a_{22} & \dots & a_{2n} \\ \vdots & 0 & \ddots & \vdots \\ 0 & 0 & & a_{nn} \end{bmatrix} \tag{19.92}$$

These elimination steps generate the R matrix because Q is orthonormal, i.e.

$$\begin{aligned} A = Q \cdot R \quad \rightarrow \quad Q^T A = Q^T Q \cdot R \quad &\rightarrow \quad Q^T A = R \\ &\rightarrow \quad H_n \cdot \ldots \cdot H_2 \cdot H_1 \cdot A = R \end{aligned} \tag{19.93}$$

After n elimination steps the original matrix A contains the upper triangular matrix R, except for the diagonal elements which can be stored in some vector. The lower triangular matrix contains the Householder vectors $u_1 \dots u_n$.

$$A = \begin{bmatrix} u_{11} & r_{12} & \dots & r_{1n} \\ u_{21} & u_{22} & & r_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ u_{n1} & u_{n2} & \dots & u_{nn} \end{bmatrix} \qquad R_{diag} = \begin{bmatrix} r_{11} \\ r_{22} \\ \vdots \\ r_{nn} \end{bmatrix} \tag{19.94}$$

With $Q^T = H_1 \cdot H_2 \cdot \ldots \cdot H_n$ this representation contains both the Q and R matrix, in a packed form, of course: Q as a composition of Householder vectors and R in the upper triangular part and its diagonal vector R_{diag}.

Step 2: Forming the new right hand side

In order to form the right hand side $Q^T z$ let remember eq. (19.88) denoting the reflection matrices used to compute Q^T.

$$H_n \cdot \ldots \cdot H_2 \cdot H_1 = Q^T \tag{19.95}$$

Thus it is possible to replace the original right hand side vector z by

$$H_n \cdot \ldots \cdot H_2 \cdot H_1 \cdot z = Q^T \cdot z \tag{19.96}$$

which yields for each $k = 1 \dots n$ the following expression:

$$H_k \cdot z = \left(I - 2 \cdot u_k \cdot u_k^T\right) \cdot z = z - 2 \cdot u_k \cdot u_k^T \cdot z \tag{19.97}$$

The latter $u_k^T \cdot z$ is a simple scalar product of two vectors. Performing eq. (19.97) for each Householder vector finally results in the new right hand side vector $Q^T z$.

Step 3: Backward substitution

The solutions in the vector x are obtained by backward substituting into the triangulated R matrix. The elements of the solution vector x are computed by the following recursive equation.

$$x_i = \frac{z_i}{r_{ii}} - \sum_{k=i+1}^{n} x_k \cdot \frac{r_{ik}}{r_{ii}} \qquad\qquad i = n, \dots, 1 \tag{19.98}$$

Motivation

Though the QR decomposition has an operation count of $2n^3 + 3n^2$ (which is about six times more than the LU decomposition) it has its advantages. The QR factorization method is said to be unconditional stable and more accurate. Also it can be used to obtain the minimum-norm (or least square) solution of under-determined equation systems.

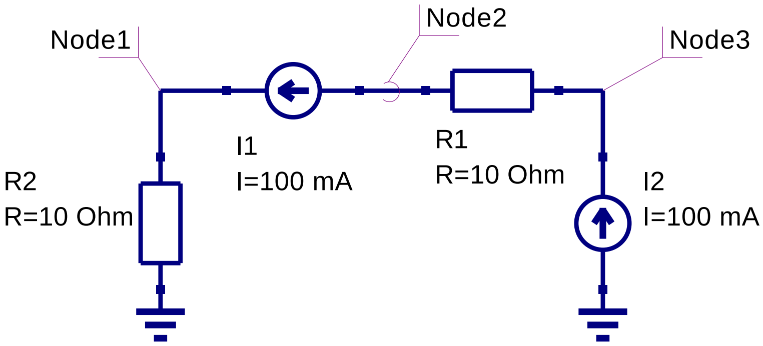


Figure 19.3: circuit with singular modified nodal analysis matrix

The circuit in fig. 19.3 has the following MNA representation:

$$Ax = \begin{bmatrix} \frac{1}{R_2} & 0 & 0 \\ 0 & \frac{1}{R_1} & -\frac{1}{R_1} \\ 0 & -\frac{1}{R_1} & \frac{1}{R_1} \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & -0.1 \\ 0 & -0.1 & 0.1 \end{bmatrix} \cdot \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} I_1 \\ -I_1 \\ I_2 \end{bmatrix} = \begin{bmatrix} 0.1 \\ -0.1 \\ 0.1 \end{bmatrix} = z \tag{19.99}$$

The second and third row of the matrix A are linear dependent and the matrix is singular because its determinant is zero. Depending on the right hand side z, the equation system has none or unlimited solutions. This is called an under-determined system. The discussed QR decomposition easily computes a valid solution without reducing accuracy. The LU decomposition would probably fail because of the singularity.

QR decomposition with column pivoting

Least norm problem

With some more effort it is possible to obtain the minimum-norm solution of this problem. The algorithm as described here would probably yield the following solution:

$$x = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix} \tag{19.100}$$

This is one out of unlimited solutions. The following short description shows how it is possible to obtain the minimum-norm solution. When decomposing the transposed problem

$$A^T = Q \cdot R \tag{19.101}$$

the minimum-norm solution $\hat{x}$ is obtained by forward substitution of

$$R^T \cdot x = z \tag{19.102}$$

and multiplying the result with Q.

$$\hat{x} = Q \cdot x \tag{19.103}$$

In the example above this algorithm results in a solution vector with the least vector norm possible:

$$\hat{x} = \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} = \begin{bmatrix} 1 \\ -0.5 \\ 0.5 \end{bmatrix} \tag{19.104}$$

This algorithm outline is also sometimes called LQ decomposition because of R^T being a lower triangular matrix used by the forward substitution.

19.2.6 Singular value decomposition

Very bad conditioned (ratio between largest and smallest eigenvalue) matrices, i.e. nearly singular, or even singular matrices (over- or under-determined equation systems) can be handled by the singular value decomposition (SVD). This type of decomposition is defined by

$$A = U \cdot \Sigma \cdot V^H \tag{19.105}$$

where the U matrix consists of the orthonormalized eigenvectors associated with the eigenvalues of $A \cdot A^H$, V consists of the orthonormalized eigenvectors of $A^H \cdot A$ and Σ is a matrix with the singular values of A (non-negative square roots of the eigenvalues of $A^H \cdot A$) on its diagonal and zeros otherwise.

$$\Sigma = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_n \end{bmatrix} \tag{19.106}$$

The singular value decomposition can be used to solve linear equation systems by simple substitutions

$$A \cdot x = z \tag{19.107}$$

$$U \cdot \Sigma \cdot V^H \cdot x = z \tag{19.108}$$

$$\Sigma \cdot V^H \cdot x = U^H \cdot z \tag{19.109}$$

since

$$U^H \cdot U = V^H \cdot V = V \cdot V^H = I \tag{19.110}$$

To obtain the decomposition stated in eq. (19.105) Householder vectors are computed and their transformations are applied from the left-hand side and right-hand side to obtain an upper bidiagonal matrix B which has the same singular values as the original A matrix because all of the transformations introduced are orthogonal.

$$U_B^{H\,(n)} \cdot \ldots \cdot U_B^{H\,(1)} \cdot A \cdot V_B^{(1)} \cdot \ldots \cdot V_B^{(n-2)} = U_B^H \cdot A \cdot V_B = B^{(0)} \tag{19.111}$$

Specifically, $U_B^{H\,(i)}$ annihilates the subdiagonal elements in column i and $V_B^{(j)}$ zeros out the appropriate elements in row j.

$$B^{(0)} = \begin{bmatrix} \beta_1 & \delta_2 & 0 & \cdots & 0 \\ 0 & \beta_2 & \delta_3 & 0 & 0 \\ \vdots & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \beta_{n-1} & \delta_n \\ 0 & 0 & 0 & 0 & \beta_n \end{bmatrix} \tag{19.112}$$

Afterwards an iterative process (which turns out to be a QR iteration) is used to transform the bidiagonal matrix B into a diagonal form by applying successive Givens transformations (therefore orthogonal as well) to the bidiagonal matrix. This iteration is said to have cubic convergence and yields the final singular values of the matrix A.

$$B^{(0)} \rightarrow B^{(1)} \rightarrow \ldots \rightarrow \Sigma \tag{19.113}$$

$$B^{(k+1)} = \left(S^{(k)}\right)^H \cdot B^{(k)} \cdot T^{(k)} \tag{19.114}$$

Each of the transformations applied to the bidiagonal matrix is also applied to the matrices U_B and V_B^H which finally yield the U and V^H matrices after convergence.

So far for the algorithm outline. Without the very details the following sections briefly describe each part of the singular value decomposition.

Notation

Beforehand some notation marks are going to be defined.

- Conjugate transposed (or adjoint):

  $$A \rightarrow \left(A^T\right)^* = \left(A^*\right)^T = A^H$$

- Euclidian norm:

  $$\|x\| = \sqrt{\sum_{i=1}^{n} x_i \cdot x_1^*} = \sqrt{\sum_{i=1}^{n} |x_i|^2} = \sqrt{|x_1|^2 + \cdots + |x_n|^2} = \sqrt{x \cdot x^H}$$

- Hermitian (or self adjoint):

  $$A = A^H$$

  whereas A^H denotes the conjugate transposed matrix of A. In the real case the matrix A is then said to be "symmetric".

- Unitary:

  $$A \cdot A^H = A^H \cdot A = I$$

  Real matrices A with this property are called "orthogonal".

Householder reflector

A Householder matrix is an elementary unitary matrix that is Hermitian. Its fundamental use is their ability to transform a vector x to a multiple of $\vec{e}_1$, the first column of the identity matrix. The elementary Hermitian (i.e. the Householder matrix) is defined as

$$H = I - 2 \cdot u \cdot u^H \quad \text{where} \quad u^H \cdot u = 1 \tag{19.115}$$

Beside excellent numerical properties, their application demonstrates their efficiency. If A is a matrix, then

$$\begin{aligned} H \cdot A &= A - 2 \cdot u \cdot u^H \cdot A \\ &= A - 2 \cdot u \cdot \left(A^H \cdot u\right)^H \end{aligned} \tag{19.116}$$

and hence explicit formation and storage of H is not required. Also colums (or rows) can be transformed individually exploiting the fact that $u^H \cdot A$ yields a scalar product for single columns or rows.

Specific case In order to reduce a 4×4 matrix A to upper triangular form successive Householder reflectors must be applied.

$$A = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix} \tag{19.117}$$

In the first step the diagonal element a_{11} gets replaced and its below elements get annihilated by the multiplication with an appropriate Householder vector, also the remaining right-hand columns get modified.

$$u_1 = \begin{bmatrix} u_{11} \\ u_{21} \\ u_{31} \\ u_{41} \end{bmatrix} \qquad H_1 = I - 2 \cdot u_1 \cdot u_1^H \qquad \rightarrow A_1 = H_1 \cdot A = \begin{bmatrix} \beta_1 & a_{12}^{(1)} & a_{13}^{(1)} & a_{14}^{(1)} \\ 0 & a_{22}^{(1)} & a_{23}^{(1)} & a_{24}^{(1)} \\ 0 & a_{32}^{(1)} & a_{33}^{(1)} & a_{34}^{(1)} \\ 0 & a_{42}^{(1)} & a_{43}^{(1)} & a_{44}^{(1)} \end{bmatrix} \tag{19.118}$$

This process must be repeated

$$u_2 = \begin{bmatrix} 0 \\ u_{22} \\ u_{32} \\ u_{42} \end{bmatrix} \qquad H_2 = I - 2 \cdot u_2 \cdot u_2^H \qquad \rightarrow A_2 = H_2 \cdot A_1 = \begin{bmatrix} \beta_1 & a_{12}^{(2)} & a_{13}^{(2)} & a_{14}^{(2)} \\ 0 & \beta_2 & a_{23}^{(2)} & a_{24}^{(2)} \\ 0 & 0 & a_{33}^{(2)} & a_{34}^{(2)} \\ 0 & 0 & a_{43}^{(2)} & a_{44}^{(2)} \end{bmatrix} \tag{19.119}$$

$$u_3 = \begin{bmatrix} 0 \\ 0 \\ u_{33} \\ u_{43} \end{bmatrix} \qquad H_3 = I - 2 \cdot u_3 \cdot u_3^H \qquad \rightarrow A_3 = H_3 \cdot A_2 = \begin{bmatrix} \beta_1 & a_{12}^{(3)} & a_{13}^{(3)} & a_{14}^{(3)} \\ 0 & \beta_2 & a_{23}^{(3)} & a_{24}^{(3)} \\ 0 & 0 & \beta_3 & a_{34}^{(3)} \\ 0 & 0 & 0 & a_{44}^{(3)} \end{bmatrix} \tag{19.120}$$

$$u_4 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ u_{44} \end{bmatrix} \qquad H_4 = I - 2 \cdot u_4 \cdot u_4^H \qquad \rightarrow A_4 = H_4 \cdot A_3 = \begin{bmatrix} \beta_1 & a_{12}^{(4)} & a_{13}^{(4)} & a_{14}^{(4)} \\ 0 & \beta_2 & a_{23}^{(4)} & a_{24}^{(4)} \\ 0 & 0 & \beta_3 & a_{34}^{(4)} \\ 0 & 0 & 0 & \beta_4 \end{bmatrix} \tag{19.121}$$

until the matrix A contains an upper triangular matrix R. The matrix Q can be expressed as the the product of the Householder vectors. The performed operations deliver

$$H_4^H \cdot H_3^H \cdot H_2^H \cdot H_1^H \cdot A = Q^H \cdot A = R \quad \rightarrow \quad A = Q \cdot R \tag{19.122}$$

since Q is unitary. The matrix Q itself can be expressed in terms of H_i using the following transformation.

$$Q^H = H_4^H \cdot H_3^H \cdot H_2^H \cdot H_1^H \tag{19.123}$$

$$\left(Q^H\right)^H = \left(H_4^H \cdot H_3^H \cdot H_2^H \cdot H_1^H\right)^H \tag{19.124}$$

$$Q = H_1 \cdot H_2 \cdot H_3 \cdot H_4 \tag{19.125}$$

The eqn. (19.123)-(19.125) are necessary to be mentioned only in case Q is not Hermitian, but still unitary. Otherwise there is no difference computing Q or Q^H using the Householder vectors. No care must be taken in choosing forward or backward accumulation.

General case In the general case it is necessary to find an elementary unitary matrix

$$H = I - \tau \cdot u \cdot u^H \tag{19.126}$$

that satisfies the following three conditions.

$$|\tau|^2 \cdot u^H \cdot u = \tau + \tau^* = 2 \cdot Re\{\tau\} \quad , \quad H^H \cdot x = \gamma \cdot \|x\| \cdot \vec{e}_1 \quad , \quad |\gamma| = 1 \tag{19.127}$$

When choosing the elements $u_{ii} = 1$ it is possible the store the Householder vectors as well as the upper triangular matrix R in the same storage of the matrix A. The Householder matrices H_i can be completely restored from the Householder vectors.

$$A = \begin{bmatrix} \beta_1 & a_{12} & a_{13} & a_{14} \\ u_{21} & \beta_2 & a_{23} & a_{24} \\ u_{31} & u_{32} & \beta_3 & a_{34} \\ u_{41} & u_{42} & u_{43} & \beta_4 \end{bmatrix} \tag{19.128}$$

There exist several approaches to meet the conditions expressed in eq. (19.127). For fewer computational effort it may be convenient to choose γ to be real valued. With the notation

$$H^H \cdot x = H^H \cdot \begin{bmatrix} \alpha \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} \beta \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{19.129}$$

one possibility is to define the following calculation rules.

$$\nu = sign\left(Re\{\alpha\}\right) \cdot \|x\| \tag{19.130}$$

$$\tau = \frac{\alpha + \nu}{\nu} \tag{19.131}$$

$$\gamma = -1 \tag{19.132}$$

$$\beta = \gamma \cdot \|x\| = -\|x\| \quad \rightarrow \quad \text{real valued} \tag{19.133}$$

$$u = \frac{x + \nu \cdot \vec{e}_1}{\alpha + \nu} \quad \rightarrow \quad u_{ii} = 1 \tag{19.134}$$

These definitions yield a complex τ, thus H is no more Hermitian but still unitary.

$$H = I - \tau \cdot u \cdot u^H \quad \rightarrow \quad H^H = I - \tau^* \cdot u \cdot u^H \tag{19.135}$$

Givens rotation

A Givens rotation is a plane rotation matrix. Such a plane rotation matrix is an orthogonal matrix that is different from the identity matrix only in four elements.

$$M = \begin{bmatrix} 1 & 0 & \cdots & & & & & & & \cdots & 0 \\ 0 & \ddots & & & & & & & & & \vdots \\ \vdots & & 1 & & & & & & & & \\ & & & +c & 0 & \cdots & 0 & +s & & & \\ & & & 0 & 1 & & & 0 & & & \\ & & & \vdots & & \ddots & & \vdots & & & \\ & & & 0 & & & 1 & 0 & & & \\ & & & -s & 0 & \cdots & 0 & +c & & & \\ & & & & & & & & 1 & & \vdots \\ \vdots & & & & & & & & & \ddots & 0 \\ 0 & \cdots & & & & & & & \cdots & 0 & 1 \end{bmatrix} \tag{19.136}$$

The elements are usually choosen so that

$$R = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \qquad c = \cos\theta,\ s = \sin\theta \quad \rightarrow \quad |c|^2 + |s|^2 = 1 \tag{19.137}$$

The most common use of such a plane rotation is to choose c and s such that for a given a and b

$$R = \begin{bmatrix} c & s \\ -s & c \end{bmatrix} \cdot \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} d \\ 0 \end{bmatrix} \tag{19.138}$$

multiplication annihilates the lower vector entry. In such an application the matrix R is often termed “Givens rotation” matrix. The following equations satisfy eq. (19.138) for a given a and b exploiting the conditions given in eq. (19.137).

$$c = \frac{\pm a}{\sqrt{|a|^2 + |b|^2}} \quad \text{and} \quad s = \frac{\pm b}{\sqrt{|a|^2 + |b|^2}} \tag{19.139}$$

$$d = \sqrt{|a|^2 + |b|^2} \tag{19.140}$$

Eigenvalues of a 2-by-2 matrix

The eigenvalues of a 2-by-2 matrix

$$A = \begin{bmatrix} a & b \\ c & d \end{bmatrix} \tag{19.141}$$

can be obtained directly from the quadratic formula. The characteristic polynomial is

$$\begin{aligned} det\,(A - \mu I) = det \begin{bmatrix} a-\mu & b \\ c & d-\mu \end{bmatrix} &= (a-\mu)\cdot(d-\mu) - bc \\ 0 &= \mu^2 - (a+d)\cdot\mu + (ad - bc) \end{aligned} \tag{19.142}$$

The polymonial yields the two eigenvalues.

$$\mu_{1,2} = \frac{a+d}{2} \pm \sqrt{\left(\frac{a+d}{2}\right)^2 + bc - ad} \tag{19.143}$$

For a symmetric matrix A (i.e. $b = c$) eq.(19.143) can be rewritten to:

$$\mu_{1,2} = e + d \pm \sqrt{e^2 + b^2} \qquad \text{with} \qquad e = \frac{a-d}{2} \tag{19.144}$$

Step 1: Bidiagonalization

In the first step the original matrix A is bidiagonalized by the application of Householder reflections from the left and right hand side. The matrices U_B^H and V_B can each be determined as a product of Householder matrices.

$$\underbrace{U_B^{H\,(n)} \cdot \ldots \cdot U_B^{H\,(1)}}_{U_B^H} \cdot A \cdot \underbrace{V_B^{(1)} \cdot \ldots \cdot V_B^{(n-2)}}_{V_B} = U_B^H \cdot A \cdot V_B = B^{(0)} \tag{19.145}$$

Each of the required Householder vectors are created and applied as previously defined. Suppose a $n \times n$ matrix, then applying the first Householder vector from the left hand side eliminates the first column and yields

$$U_B^{H\,(1)} \cdot A = \begin{bmatrix} \boldsymbol{\beta_1} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1n}^{(1)} \\ u_{21} & a_{22}^{(1)} & a_{23}^{(1)} & & a_{2n}^{(1)} \\ u_{31} & a_{32}^{(1)} & a_{33}^{(1)} & & a_{3n}^{(1)} \\ \vdots & & & \ddots & \vdots \\ u_{n1} & a_{n2}^{(1)} & a_{n3}^{(1)} & \cdots & a_{nn}^{(1)} \end{bmatrix} \tag{19.146}$$

Next, a Householder vector is applied from the right hand side to annihilate the first row.

$$U_B^{H\,(1)} \cdot A \cdot V_B^{(1)} = \begin{bmatrix} \beta_1 & \boldsymbol{\delta_2} & v_{13} & \cdots & v_{1n} \\ u_{21} & a_{22}^{(2)} & a_{23}^{(2)} & & a_{2n}^{(2)} \\ u_{31} & a_{32}^{(2)} & a_{33}^{(2)} & & a_{3n}^{(2)} \\ \vdots & & & \ddots & \vdots \\ u_{n1} & a_{n2}^{(2)} & a_{n3}^{(2)} & \cdots & a_{nn}^{(2)} \end{bmatrix} \tag{19.147}$$

Again, a Householder vector is applied from the left hand side to annihilate the second column.

$$U_B^{H\,(2)} \cdot U_B^{H\,(1)} \cdot A \cdot V_B^{(1)} = \begin{bmatrix} \beta_1 & \delta_2 & v_{13} & \cdots & v_{1n} \\ u_{21} & \boldsymbol{\beta_2} & a_{23}^{(3)} & & a_{2n}^{(3)} \\ u_{31} & u_{32} & a_{33}^{(3)} & & a_{3n}^{(3)} \\ \vdots & \vdots & & \ddots & \vdots \\ u_{n1} & u_{n2} & a_{n3}^{(3)} & \cdots & a_{nn}^{(3)} \end{bmatrix} \tag{19.148}$$

This process is continued until

$$U_B^{H} \cdot A \cdot V_B = \begin{bmatrix} \beta_1 & \delta_2 & v_{13} & \cdots & v_{1n} \\ u_{21} & \beta_2 & \delta_3 & & v_{2n} \\ u_{31} & u_{32} & \ddots & \ddots & \\ \vdots & \vdots & & \beta_{n-1} & \delta_n \\ u_{n1} & u_{n2} & u_{n3} & & \boldsymbol{\beta_n} \end{bmatrix} \tag{19.149}$$

For each of the Householder transformations from the left and right hand side the appropiate τ values must be stored in separate vectors.

Step 2: Matrix reconstructions

Using the Householder vectors stored in place of the original A matrix and the appropiate τ value vectors it is now necessary to unpack the U_B and V_B^H matrices. The diagonal vector β and the super-diagonal vector δ can be saved in separate vectors previously. Thus the U_B matrix can be unpacked in place of the A matrix and the V_B^H matrix is unpacked in a separate matrix.

There are two possible algorithms for computing the Householder product matrices, i.e. forward accumulation and backward accumulation. Both start with the identity matrix which is successively multiplied by the Householder matrices either from the left or right.

$$U_B^H = H_{Un}^H \cdot \ldots \cdot H_{U2}^H \cdot H_{U1}^H \cdot I \tag{19.150}$$

$$\rightarrow \quad U_B = I \cdot H_{Un} \cdot \ldots \cdot H_{U2} \cdot H_{U1} \tag{19.151}$$

Recall that the leading portion of each Householder matrix is the identity except the first. Thus, at the beginning of backward accumulation, U_B is "mostly the identity" and it gradually becomes full as the iteration progresses. This pattern can be exploited to reduce the number of required flops. In contrast, U_B^H is full in forward accumulation after the first step. For this reason, backward accumulation is cheaper and the strategy of choice. When unpacking the U_B matrix in place of the original A matrix it is necessary to choose backward accumulation anyway.

$$V_B = I \cdot H_{V1}^H \cdot H_{V2}^H \cdot \ldots \cdot H_{Vn}^H \tag{19.152}$$

$$\rightarrow \quad V_B^H = I \cdot H_{Vn} \cdot \ldots \cdot H_{V2} \cdot H_{V1} \tag{19.153}$$

Unpacking the V_B^H matrix is done in a similar way also performing successive Housholder matrix multiplications using backward accumulation.

Step 3: Diagonalization – shifted QR iteration

At this stage the matrices U_B and V_B^H exist in unfactored form. Also there are the diagonal vector β and the super-diagonal vector δ. Both vectors are real valued. Thus the following algorithm can be applied even though solving a complex equation system.

$$B^{(0)} = \begin{bmatrix} \beta_1 & \delta_2 & 0 & \cdots & 0 \\ 0 & \beta_2 & \delta_3 & 0 & 0 \\ \vdots & 0 & \ddots & \ddots & 0 \\ 0 & 0 & 0 & \beta_{n-1} & \delta_n \\ 0 & 0 & 0 & 0 & \beta_n \end{bmatrix} \tag{19.154}$$

The remaining problem is thus to compute the SVD of the matrix B. This is done applying an implicit-shift QR step to the tridiagonal matrix $T = B^T B$ which is a symmetric. The matrix T is not explicitely formed that is why a QR iteration with implicit shifts is applied.

After bidiagonalization we have a bidiagonal matrix $B^{(0)}$:

$$B^{(0)} = U_B^H \cdot A \cdot V_B \tag{19.155}$$

The presented method turns $B^{(k)}$ into a matrix $B^{(k+1)}$ by applying a set of orthogonal transforms

$$B^{(k+1)} = S^H \cdot B^{(k)} \cdot T \tag{19.156}$$

The orthogonal matrices S and T are chosen so that $B^{(k+1)}$ is also a bidiagonal matrix, but with the super-diagonal elements smaller than those of $B^{(k)}$. The eq.(19.156) is repeated until the non-diagonal elements of $B^{(k+1)}$ become smaller than ε and can be disregarded.

The matrices S and T are constructed as

$$S = S_1 \cdot S_2 \cdot S_3 \cdot \ldots \cdot S_n \tag{19.157}$$

and similarly T where T_i and S_i are matrices of simple rotations as given in eq.(19.136). Both T and S are products of Givens rotations and thus perform orthogonal transforms.

The left multiplication of $B^{(k)}$ by S_i^H replaces two rows of $B^{(k)}$ by their linear combinations. The rest of $B^{(k)}$ is unaffected. Right multiplication of $B^{(k)}$ by T_i similarly changes only two columns of $B^{(k)}$.

A matrix T_2 is chosen the way that

$$B^{(k+1)} = T_2^H \cdot \left(B^{(k)}\right)^H \cdot B^{(k)} \cdot T_2 \tag{19.158}$$

is a QR transform with a shift. Note that multiplying $B^{(k)}$ by T_2 gives rise to a non-zero element which is below the main diagonal. A new rotation angle is then chosen so that multiplication by S_2^H gets rid of that elemnent. But this will create a non-zero element which is right beside the super-diagonal. T_3 is made to make it disappear, but this leads to another non-zero element below the diagonal, etc.

In the end, S_n the matrix $S^H B T$ becomes bidiagonal again. However, because of a special choice of T_2 (QR algorithm), its non-diagonal elements are smaller than those of B.

For a single QR step the computation of the eigenvalue μ of the 2-by-2 submatrix of $T_n = B_n^T \cdot B_n$ that is closer to the t_{22} matrix element is required.

$$T_n = \begin{bmatrix} t_{11} & t_{12} \\ t_{21} & t_{22} \end{bmatrix} = B_n^T \cdot B_n = \begin{bmatrix} \beta_{n-1} & 0 \\ \delta_n & \beta_n \end{bmatrix} \cdot \begin{bmatrix} \beta_{n-1} & \delta_n \\ 0 & \beta_n \end{bmatrix} \tag{19.159}$$

$$= \begin{bmatrix} \beta_{n-1}^2 & \delta_n \cdot \beta_{n-1} \\ \delta_n \cdot \beta_{n-1} & \beta_n^2 + \delta_n^2 \end{bmatrix} \tag{19.160}$$

The required eigenvalue is called Wilkinson shift, see eq.(19.144) for details. The sign for the eigenvalue is chosen such that it is closer to t_{22}.

$$\mu = t_{22} + d - sign(d) \cdot \sqrt{d^2 + t_{12}^2} \tag{19.161}$$

$$= t_{22} + d - t_{12}^2 \cdot sign\left(\frac{d}{t_{12}}\right) \cdot \sqrt{\left(\frac{d}{t_{12}}\right)^2 + 1} \tag{19.162}$$

whereas

$$d = \frac{t_{11} - t_{22}}{2} \tag{19.163}$$

Step 4: Solving the equation system

It is straight-forward to solve a given equation system once having the singular value decomposition computed.

$$A \cdot x = z \tag{19.164}$$

$$U\Sigma V^H \cdot x = z \tag{19.165}$$

$$\Sigma V^H \cdot x = U^H \cdot z \tag{19.166}$$

$$V^H \cdot x = \Sigma^{-1} U^H \cdot z \tag{19.167}$$

$$x = V\Sigma^{-1} U^H \cdot z \tag{19.168}$$

The inverse of the diagonal matrix Σ yields

$$\Sigma^{-1} = \begin{bmatrix} 1/\sigma_1 & 0 & \cdots & 0 \\ 0 & 1/\sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/\sigma_n \end{bmatrix} \tag{19.169}$$

With v_i being the i-th row of the matrix V, u_i the i-th column of the matrix U and σ_i the i-th singular value eq. (19.168) can be rewritten to

$$x = \sum_{i=1}^{n} \frac{u_i^H \cdot z}{\sigma_i} \cdot v_i \tag{19.170}$$

It must be mentioned that very small singular values σ_i corrupt the complete result. Such values indicate (nearly) singular (ill-conditioned) matrices A. In such cases, the solution vector x obtained by zeroing the small σ_i's and then using equation (19.168) is better than direct-method solutions (such as LU decomposition or Gaussian elimination) and the SVD solution where the small σ_i's are left non-zero. It may seem paradoxical that this can be so, since zeroing a singular value corresponds to throwing away one linear combination of the set of equations that is going to be solved. The resolution of the paradox is that a combination of equations that is so corrupted by roundoff error is thrown away precisely as to be at best useless; usually it is worse than useless since it "pulls" the solution vector way off towards infinity along some direction that is almost a nullspace vector.

19.2.7 Jacobi method

This method quite simply involves rearranging each equation to make each variable a function of the other variables. Then make an initial guess for each solution and iterate. For this method it is necessary to ensure that all the diagonal matrix elements a_{ii} are non-zero. This is given for the nodal analysis and almostly given for the modified nodal analysis. If the linear equation system is solvable this can always be achieved by rows substitutions.

The algorithm for performing the iteration step $k+1$ writes as follows.

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(z_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right) \qquad \text{for } i = 1, \ldots, n \tag{19.171}$$

This has to repeated until the new solution vectors $x^{(k+1)}$ deviation from the previous one $x^{(k)}$ is sufficiently small.

The initial guess has no effect on whether the iterative method converges or not, but with a good initial guess (as possibly given in consecutive Newton-Raphson iterations) it converges faster (if it converges). To ensure convergence the condition

$$\sum_{j=1, j \neq i}^{n} |a_{ij}| \leq |a_{ii}| \qquad \text{for } i = 1, \ldots, n \tag{19.172}$$

and at least one case

$$\sum_{i=1, i \neq j}^{n} |a_{ij}| \leq |a_{ii}| \tag{19.173}$$

must apply. If these conditions are not met, the iterative equations may still converge. If these conditions are met the iterative equations will definitely converge.

Another simple approach to a convergence criteria for iterative algorithms is the Schmidt and v. Mises criteria.

$$\sqrt{\sum_{i=1}^{n} \sum_{j=1, j \neq i}^{n} \left| \frac{a_{ij}}{a_{ii}} \right|^2} < 1 \tag{19.174}$$

19.2.8 Gauss-Seidel method

The Gauss-Seidel algorithm is a modification of the Jacobi method. It uses the previously computed values in the solution vector of the same iteration step. That is why this iterative method is expected to converge faster than the Jacobi method.

The slightly modified algorithm for performing the $k+1$ iteration step writes as follows.

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(z_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^{n} a_{ij} x_j^{(k)} \right) \qquad \text{for } i = 1, \ldots, n \tag{19.175}$$

The remarks about the initial guess $x^{(0)}$ as well as the convergence criteria noted in the section about the Jacobi method apply to the Gauss-Seidel algorithm as well.

19.2.9 A comparison

Of course, there are many different algorithms for solving linear equation systems. The table below shows a comparison of the most famous ones. In circuit simulations gaussian elimination or LU decomposition are usually used. All the others are less fast and have no practical advantages. The listings below show functions for these two algorithms. Gaussian elimination is a little bit faster, whereas LU decomposition can solve very fast several matrices when the right-hand side differs only.

method	precision	application	programming effort	computing complexity	notes
Laplace expansion	numerical errors	general	straight forward	$n!$	very time consuming
Gaussian elimination	numerical errors	general	intermediate	$n^3/3 + n^2/2$	
Gauss-Jordan	numerical errors	general	intermediate	$n^3/3 + n^2 - n/3$	computes the inverse besides
LU decomposition	numerical errors	general	intermediate	$n^3/3 + n^2 - n/3$	useful for consecutive solutions
QR decomposition	good	general	high	$2n^3 + 3n^3$	
Singular value decomposition	good	general	very high	$2n^3 + 4n^3$	ill-conditioned matrices can be handled
Jacobi	very good	diagonally dominant systems	easy	n^2 in each iteration step	possibly no convergence
Gauss-Seidel	very good	diagonally dominant systems	easy	n^2 in each iteration step	possibly no convergence

Listing 19.1: gaussian elimination algorithm in C

```
// ******************************************************************
// Solves the linear equation system "mat * rVec = iVec"
// by using Gaussian elimination with column pivoting. Thus,
// the row numbers in result vector "rVec" stay unchanged.
// Returns false if no solution exists. Otherwise returns true and the
// solution is in "rVec". "rVec" needs not to be filled with values before.
// ******************************************************************

bool solveRealEqnSys(double *mat, double *iVec, double *rVec, int Size)
{
  double d, tmp, *p1, *p2;
  int i, j, k, pivot = -1;

  // forward elimination
  for(i=0; i<Size; i++) {

    // search pivot element
    d = 0.0;
    p1 = mat + i;
    for(j=i; j<Size; j++)
      if(fabs(p1[j*Size]) > d) {
        d = fabs(p1[j*Size]);
        pivot = j;
      }

    if(d < 1e-150)
      return false;    // matrix singular

    if(pivot != i) {
      // exchange rows in vector
      d = iVec[i];
      iVec[i] = iVec[pivot];
      iVec[pivot] = d;

      // exchange rows in matrix
      p1 = mat + i*Size;
      p2 = mat + pivot*Size;
      for(j=i; j<Size; j++) {
        d = p1[j];
        p1[j] = p2[j];
        p2[j] = d;
      }
    }

    // perform elimination step
    p1 = mat + i*Size;
    tmp = p1[i];
    for(j=i+1; j<Size; j++) {
      p2 = mat + j*Size;
      d = p2[i] / tmp;
      for(k=i+1; k<Size; k++)
        p2[k] -= d * p1[k];
      iVec[j] -= d * iVec[i];
    }
  }

  // backward substitution
  for(i=Size-1; i >= 0; i--) {
    d = iVec[i];
    p1 = mat + i*Size;
    for(j=i+1; j<Size; j++)
      d -= p1[j] * rVec[j];
    rVec[i] = d / p1[i];
  }

  return true;
}
```

Listing 19.2: LU decomposition algorithm in C

```
// ********************************************************************
// LU decomposition with column pivoting (from Golub and Van Loan).
// Solves the linear equation system "mat * RESULT = vec". The result is
// in "vec". The row numbers in result vector "rVec" stay unchanged.
// Returns false if no solution exists. Otherwise returns true
// ********************************************************************

bool solveComplexEqnSys(complex *mat, complex *vec, int Size)
{
  double d;
  complex c, tmp, *p1, *p2;
  int i, j, k, pivot = -1;

  for(i=0; i<Size; i++) {  // going through every row

    d = 0.0;
    p1 = mat + i;
    for(j=i; j<Size; j++)  // search pivot element
      if(d < norm(p1[j*Size])) {
        d =  norm(p1[j*Size]);
        pivot = j;
      }

    if(d < 1e-150)
      return false;    // matrix singular

    if(pivot != i) {
      p1 = mat + i*Size;
      p2 = mat + pivot*Size;
      for(j=Size-1; j>=0; j--) { // exchange rows in matrix
        c = p1[j];
        p1[j] = p2[j];
        p2[j] = c;
      }

      if(vec) {
        c = vec[i];
        vec[i] = vec[pivot];   // exchange RHS elements
        vec[pivot] = c;
      }
    }

    p1 = mat + i*Size;
    tmp = p1[i];
    for(j=i+1; j<Size; j++) {   // perform elimination step
      p2 = mat + j*Size;
      p2[i] /= tmp;
      c = p2[i];

      for(k=i+1; k<Size; k++)
        p2[k] -= c * p1[k];
    }
  }

  // forward substitution of LU matrix
  for(i=0; i<Size; i++) {
    c = vec[i];
    p1 = mat + i*Size;
    for(j=0; j<i; j++)
      c -= p1[j] * vec[j];
    vec[i] = c;
  }

  // backward substitution of LU matrix
  for(i=Size-1; i>=0; i--) {
    c = vec[i];
    p1 = mat + i*Size;
    for(j=i+1; j<Size; j++)
      c -= p1[j] * vec[j];
    vec[i] = c / p1[i];
  }

  return true;
}
```

As can be seen from the table above the time for solving an equation system with direct solvers like Gaussian elimination or LU decomposition rises cubically with the number of rows. Hence, large circuits soon create very time consuming simulations. Fortunately, the MNA matrix is sparse, i.e. most of its elements are zero. Sparse solver algorithms take advantage from this and store non-zero elements only. As this saves much memory, it also speeds up the algorithm, because memory through-put and cache size dominate the computation time. But the biggest speed-up can be reached by using algorithms that order the rows and columns of the MNA matrix in a way that minimizes the fill-in (elements that change from zero to non-zero during the LU decomposition). For circuit simulations the most useful open-source packages are Sparse1.4 [93] and SuiteSparse [94]. But even with these programs, the computation time for solving the MNA matrix dominates the simulation speed. Except for small circuits, the component models takes 10% of the overall time only.
Figure 19.4 gives an impression about the simulation time with different equation solvers using transient analysis with 5000 time steps on a 1.4GHz Pentium® processor.

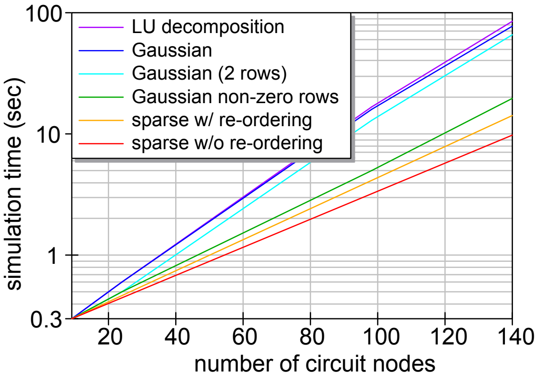


Figure 19.4: speed comparison of different equation solvers

Iterative matrix solvers like BiCGStab or GMRES are able to reach even more speed advantages than sparse solvers. The disadvantages are that they find an approximate solution only and that they may encounter severe convergence problems.

19.3 Frequency-Time Domain Transformation

Any signal can completely be described in time or in frequency domain. As both representations are equivalent, it is possible to transform them into each other. This is done by the so-called Fourier Transformation (FT) and the inverse Fourier Transformation (IFT), respectively:

$$\text{Fourier Transformation:} \qquad \underline{U}(f) = \int_{-\infty}^{\infty} u(t) \cdot e^{-j\cdot 2\pi f \cdot t}\, dt \tag{19.176}$$

$$\text{inverse Fourier Transformation:} \qquad u(t) = \int_{-\infty}^{\infty} \underline{U}(f) \cdot e^{j\cdot 2\pi f \cdot t}\, df \tag{19.177}$$

In digital systems the data $u(t)$ or $\underline{U}(f)$, respectively, consists of a finite number N of samples u_k and $\underline{U}_n$. This leads to the discrete Fourier Transformation (DFT) and its inverse operation

(IDFT):

$$\text{DFT:} \qquad \underline{U}_n = \sum_{k=0}^{N-1} u_k \cdot \exp\left(-j \cdot n \frac{2\pi \cdot k}{N}\right) \tag{19.178}$$

$$\text{IDFT:} \qquad u_k = \frac{1}{N} \cdot \sum_{n=0}^{N} \underline{U}_n \cdot \exp\left(j \cdot k \frac{2\pi \cdot n}{N}\right) \tag{19.179}$$

The absolute time and frequency values do not appear anymore in the DFT. They depend on the sampling frequency f_s and the number of samples N.

$$\Delta f = \frac{1}{T_s} = \frac{1}{N \cdot \Delta t} = \frac{f_s}{N} = \frac{2 \cdot f_{max}}{N} \tag{19.180}$$

$$\Delta t = \frac{1}{f_s} = \frac{1}{2 \cdot f_{max}} = \frac{1}{N \cdot \Delta f} = \frac{T_s}{N} \tag{19.181}$$

Where Δf is the distance between frequency samples, Δt is the distance between time samples and T_s is the complete sampling time interval. Because the DFT creates positive and negative frequencies, the maximum frequency f_{max} (also called Nyquist frequency) is half of the sampling frequency.

During the FT, the quantity is multiplied by time. So the unit for a voltage becomes "Vs", i.e. it's now a voltage per frequency. This isn't the case for the DFT. Here the unit of the operand stays unchanged, because of its normalization to the sampling frequency. To get a comparable quantity in the discrete domain, the DFT needs to be multiplied by the sampling time step:

$$\underline{U}_{T,n} = \Delta t \cdot \underline{U}_n \tag{19.182}$$

Indeed, the calculation of the signal power P consumed by the resistance R doesn't need the time information. But the calculation of the signal energy W does.

$$P \cdot R = \frac{W}{T_s} \cdot R = \frac{1}{T_s} \cdot \int_0^{T_s} u^2 \, dt \approx \frac{1}{T_s} \cdot \sum^{N} u_k^2 \cdot \Delta t = \frac{1}{N} \cdot \sum^{N} u_k^2 \tag{19.183}$$

The same calculation can be performed in frequency domain. The results equal each other exactly.

$$P \cdot R = \frac{W}{T_s} \cdot R = \frac{1}{T_s} \cdot \int_{-f_{max}}^{f_{max}} |\underline{U}_T|^2 \, df \approx \frac{1}{T_s} \cdot \sum^{N} |\underline{U}_{T,n}|^2 \cdot \Delta f = \frac{1}{N^2} \cdot \sum^{N} |\underline{U}_n|^2 \tag{19.184}$$

I.e. the power spectral density in continuous, discrete and normalized discrete domain is, respectively:

$$\text{PSD} = \lim_{T \to \infty} \frac{|\underline{U}_T|^2}{T} \qquad \text{PSD} = \frac{\Delta t \cdot |\underline{U}_n|^2}{N} \qquad \text{PSD} = \frac{|\underline{U}_n|^2}{N} \tag{19.185}$$

With DFT the N time samples are transformed into N frequency samples. This also holds if the time data are real numbers, as is always the case in "real life": The complex frequency samples are conjugate complex symmetrical and so equalizing the score:

$$\underline{U}_{N-n} = \underline{U}_n^* \tag{19.186}$$

That is, knowing the input data has no imaginary part, only half of the Fourier data must be computed.

19.3.1 Fast Fourier Transformation

As can be seen in equation 19.178 the computing time of the DFT rises with N^2. This is really huge, so it is very important to reduce the time consumption. Using a strongly optimized algorithm, the so-called Fast Fourier Transformation (FFT), the DFT is reduced to an $N \cdot \log_2 N$ time rise. The following information stems from [95], where the theoretical background is explained comprehensively.

The fundamental trick of the FFT is to cut the DFT into two parts, one with data having even indices and the other with odd indices:

$$\underline{U}_n = \sum_{k=0}^{N-1} u_k \cdot \exp\left(-j \cdot n \frac{2\pi \cdot k}{N}\right) \tag{19.187}$$

$$= \sum_{k=0}^{N/2-1} u_{2k} \cdot \exp\left(-j \cdot n \frac{2\pi \cdot 2k}{N}\right) + \sum_{k=0}^{N/2-1} u_{2k+1} \cdot \exp\left(-j \cdot n \frac{2\pi \cdot (2k+1)}{N}\right) \tag{19.188}$$

$$= \underbrace{\sum_{k=0}^{N/2-1} u_{2k} \cdot \exp\left(-j \cdot n \frac{2\pi \cdot k}{N/2}\right)}_{F_{even}} + W_{n,N} \cdot \underbrace{\sum_{k=0}^{N/2-1} u_{2k+1} \cdot \exp\left(-j \cdot n \frac{2\pi \cdot k}{N/2}\right)}_{F_{odd}} \tag{19.189}$$

$$\text{with} \qquad W_{n,N} = \exp\left(2\pi \cdot j \cdot \frac{n}{N}\right) \qquad \text{(so-called 'twiddle factor')} \tag{19.190}$$

The new formula shows no speed advantages. The important thing is that the even as well as the odd part each is again a Fourier series. Thus the same procedure can be repeated again and again until the equation consists of N terms. Then, each term contains only one data u_k with factor $e^0 = 1$. This works if the number of data is a power of two (2, 4, 8, 16, 32, ...). So finally, the FFT method performs $\log_2 N$ times the operation

$$u_{k1,even} + W_{n,x} \cdot u_{k2,odd} \tag{19.191}$$

to get one data of $\underline{U}_n$. This is called the Danielson-Lanzcos algorithm. The question now arises which data values of u_k needs to be combined according to equation (19.191). The answer is quite easy. The data array must be reordered by the bit-reversal method. This means the value at index k_1 is swapped with the value at index k_2 where k_2 is obtained by mirroring the binary number k_1, i.e. the most significant bit becomes the least significant one and so on. Example for $N = 8$:

000	↔	000	011	↔	110	110	↔	011
001	↔	100	100	↔	001	111	↔	111
010	↔	010	101	↔	101			

Having this new indexing, the values to combine according to equation 19.191 are the adjacent values. So, performing the Danielson-Lanzcos algorithm has now become very easy.

Figure 19.5 illustrates the whole FFT algorithm starting with the input data u_k and ending with one value of the output data $\underline{U}_n$.

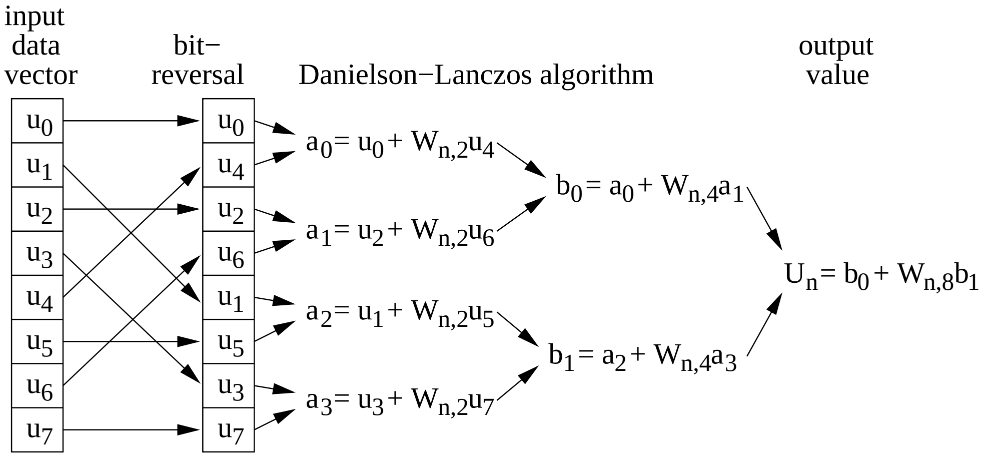


Figure 19.5: principle of a FFT with data length 8

This scheme alone gives no advantage. But it can compute all output values within, i.e. no temporary memory is needed and the periocity of $W_{n,N}$ is best exploited. To understand this, let's have a look on the first Danielson-Lanczos step in figure 19.5. The four multiplications and additions have to be performed for each output value (here 8 times!). But indeed this is not true, because $W_{n,2}$ is 2-periodical in n and furthermore $W_{n,2} = -W_{n+1,2}$. So now, $u_0 + W_{0,2} \cdot u_4$ replaces the old u_0 value and $u_0 - W_{0,2} \cdot u_4$ replaces the old u_4 value. Doing this for all values, four multiplications and eight additions were performed in order to calculate the first Danielson-Lanczos step for all (!!!) output values. This goes on, as $W_{n,4}$ is 4-periodical in n and $W_{n,4} = -W_{n+2,4}$. So this time, two loop iterations (for $W_{n,4}$ and for $W_{n+1,4}$) are necessary to compute the current Danielson-Lanczos step for all output values. This concept continues until the last step.

Finally, a complete FFT source code in C should be presented. The original version was taken from [95]. It is a radix-2 algorithem, known as the Cooley-Tukey Algorithm. Here, several changes were made that gain about 10% speed improvement.

Listing 19.3: 1D-FFT algorithm in C

```
// ***********************************************************
// Parameters:
// num   - number of complex samples
// *data - array containing the data samples, real and imaginary
//         part in alternating order (length: 2*num)
//         order of frequencies: 0, 1, 2, ... , N/2, ... , -2, -1
// isign - is 1 to calculate FFT and -1 to calculate inverse FFT
// ***********************************************************

#define SWAP(a,b) { wr = a; a = b; b = wr; }

void fft_radix2(int num, double *data, int isign)
{
  double wt, theta, wr, wi, wpr, wpi, tempi, tempr;
  int i, j, m, n;
  n = 2*num;
  j = 0;

  // bit reversal method
  // 1) index 0 need not to be swapped
  //     -> start at i=2
  // 2) swap scheme is symmetrical
  //     -> swap first and second half in one iteration
  for (i=2; i<num; i+=2) {
    m = num;
    while (j >= m) {  // calculate swap index
      j -= m;
      m >>= 1;
    }
    j += m;

    if(j > i) {    // was index already swapped ?
      SWAP(data[j], data[i]);       // swap real part
      SWAP(data[j+1], data[i+1]); // swap imaginary part

      if(j < num) {  // swap second half ?
        SWAP (data[n-j-2], data[n-i-2]); // swap real part
        SWAP (data[n-j-1], data[n-i-1]); // swap imaginary part
      }
    }
  }

  // Danielson-Lanzcos algorithm
  int mmax, istep;
 mmax = 2;
  while (n > mmax) { // each Danielson-Lanzcos step
    istep = mmax << 1;
    theta = isign * (2.0 * PI / mmax);
    wpr = cos(theta);
    wpi = sin(theta);
    wr = 1.0;
    wi = 0.0;
    for (m=1; m<mmax; m+=2) {
      for (i=m; i<=n; i+=istep) {
        j = i+mmax;
        tempr = wr*data[j-1] + wi*data[j];
        tempi = wr*data[j] - wi*data[j-1];
        data[j-1] = data[i-1] - tempr;
        data[j] = data[i] - tempi;
        data[i-1] += tempr;
        data[i] += tempi;
      }
      wt = wr;
      wr = wt*wpr - wi*wpi;
      wi = wi*wpr + wt*wpi;
    }
    mmax = istep;
  }

  if (isign == -1)          // perform inverse FFT ?
    for (i=0; i<n; i++)
      data[i] /= num;       // normalize result
}
```

There are many other FFT algorithms mainly aiming at higher speed (radix-4 FFT, radix-8 FFT, split-radix FFT, Winograd FFT). These algorithms are much more complex, but on modern processors with numerical co-processors they gain only small speed advantages, because the reduced

FLOPS are equaled by the far more complex indexing.
The above-mentioned methods only work if the data size is a power of two. For other data sizes the Bluestein algorithm can be used. Even though it runs about four times slower, it still has $N \cdot \log(N)$ complexity and thus, is much faster than the standard DFT.

19.3.2 Real-Valued FFT

All physical systems are real-valued in time domain. As already mentioned above, this fact leads to a symmetry in frequency domain, which can be exploited to save 50% memory usage and about 30% computation time. Rewriting the C listing from above to a real-valued FFT routine creates the following function. As this scheme is not symmetric anymore, an extra procedure for the inverse transformation is needed. It is also depicted below.

Listing 19.4: real-valued FFT algorithm in C

```c
// ************************************************************
// Parameters:
// num   - number of real-valued samples (must be power of 2)
// *data - array containing the time samples (length: num)
//
// Output:
// *data - r(0), r(1), i(1), .... , r(N/2-1), i(N/2-1), r(N/2)
// ************************************************************

#define SWAP(a,b) { wr = a; a = b; b = wr; }

void real_fft_radix2(int num, double *data)
{
  int i, j, k, l, n1 = num >> 1, n2 = 1;
  double t1, t2, t3, wr, wi, wpr, wpi;

  // bit reversal method
  // 1) index 0 need not to be swapped
  //    -> start at i=1
  // 2) swap scheme is symmetrical
  //    -> swap first and second half in one iteration
  j = 0;
  for (i=1; i<n1; i++) {
    k = n1;
    while (j >= k) {  // calculate swap index
      j -= k;
      k >>= 1;
    }
    j += k;

    if(j > i) {   // was index already swapped ?
      SWAP(data[j], data[i]);

      if(j < n1)  // swap second half ?
        SWAP (data[num-j-1], data[num-i-1]);
    }
  }

  // length two butterflies
  for (i=0; i<num; i+=2) {
    t1 = data[i+1];
    data[i+1] = data[i] - t1;
    data[i]  += t1;
  }

  while (n1 < num) {
    n2 <<= 1;        // half a butterfly
    n1 = n2 << 1;    // length of a butterfly

    for (i=0; i<num; i+=n1) {
      t1 = data[i+n2];
      data[i+n2] = -data[i+n1-1];
      data[i+n1-1] = data[i] - t1;
      data[i] += t1;
    }

    t1 = 2.0*M_PI / ((double)n1);
    wpr = cos(t1);  // real part of twiddle factor
    wpi = sin(t1);  // imaginary part of twiddle factor
    wr = 1.0;    // start of twiddle factor
    wi = 0.0;

    for (j=3; j<n2; j+=2) { // all complex lines of a butterfly
      t1 = wr;
      wr = t1*wpr - wi*wpi; // calculate next twiddle factor
      wi = wi*wpr + t1*wpi;

      for (i=0; i<num; i+=n1) {  // through all butterflies
        k = i + j - 2;
        l = i + n1 - j;
        t1 = data[l]*wr + data[l+1]*wi;
        t3 = data[k+1];
        t2 = data[l+1]*wr - data[l]*wi;
        data[l] = data[k];

        if((i & n1) != 0) {  // index swap ?
          t1 = -t1;
          t3 = -t3;
        }

        data[k]   += t1;
        data[k+1] = t2 + t3;
        data[l]   -= t1;
        data[l+1] = t2 - t3;
      }
    }
  }
}
```

Listing 19.5: real-valued inverse FFT algorithm in C

```
// *********************************************************
// Parameters:
// num   - count of numbers in data (must be power of 2)
// *data - r(0), r(1), i(1), .... , r(N/2-1), i(N/2-1), r(N/2)
//
// Output:
// *data - array containing the time samples (length: num)
// *********************************************************

#define SWAP(a,b) { wr = a; a = b; b = wr; }
```

```
void real_ifft_radix2(int num, double *data)
{
  int i, j, k, l, n1, n2 = num;
  double t1, t2, t3, wr, wi, wpr, wpi;

  while (n2 > 2) {
    n1 = n2;   // length of a butterfly
    n2 >>= 1;  // half a butterfly

    for (i=0; i<num; i+=n1) {  // through all butterflies
      t1 = data[i+n1-1];
      data[i+n1-1] = -2.0 * data[i+n2];
      data[i+n2-1] *= 2.0;
      data[i+n2] = data[i] - t1;
      data[i] += t1;
    }

    t1 = 2.0*M_PI / ((double)n1);
    wpr = cos(t1);  // real part of twiddle factor
    wpi = sin(t1);  // imaginary part of twiddle factor
    wr = 1.0;   // start of twiddle factor
    wi = 0.0;

    for (j=3; j<n2; j+=2) { // all complex lines of a butterfly
      t1 = wr;
      wr = t1*wpr + wi*wpi; // calculate next twiddle factor
      wi = wi*wpr - t1*wpi;

      for (i=0; i<num; i+=n1) {  // through all butterflies
        k = i + j - 2;
        l = i + n1 - j;
        t1 = data[l] - data[k];
        t2 = data[l+1] + data[k+1];
        t3 = data[k+1] - data[l+1];
        data[k] += data[l];

        if((i & n1) != 0) {
          t1 = -t1;
          t3 = -t3;
        }

        data[k+1] = t3;
        data[l]   = t2*wi - t1*wr;
        data[l+1] = t2*wr + t1*wi;
      }
    }
  }

  // length two butterflies
  for (i=0; i<num; i+=2) {
    t1 = data[i+1];
    data[i+1] = (data[i] - t1) / num;
    data[i]   = (data[i] + t1) / num;
  }

  // bit reversal method
  // 1) index 0 need not to be swapped
  //     -> start at i=1
  // 2) swap scheme is symmetrical
  //     -> swap first and second half in one iteration
  j = 0;
  n1 = num >> 1;
  for (i=1; i<n1; i++) {
    k = n1;
    while (j >= k) {  // calculate swap index
      j -= k;
      k >>= 1;
    }
    j += k;

    if(j > i) {    // was index already swapped ?
      SWAP(data[j], data[i]);

      if(j < n1)  // swap second half ?
        SWAP (data[num-j -1], data[num-i -1]);
    }
  }
}
```

19.3.3 More-Dimensional FFT

A standard Fourier transformation is not useful in harmonic balance methods, because with multi-tone excitation many mixing products appear. The best way to cope with this problem is to use multi-dimensional Fourier transformation and its inverse opreation, respectively.

$$\underline{U}(j\omega_1, j\omega_1, \ldots, j\omega_n) = \tag{19.192}$$

$$\int\limits_{-\infty}^{\infty}\int\limits_{-\infty}^{\infty}\ldots\int\limits_{-\infty}^{\infty} u(t_1, t_2, \ldots, t_n)\cdot e^{-j\omega_1\cdot t_1}\cdot e^{-j\omega_2\cdot t_2}\ldots\cdot e^{-j\omega_n\cdot t_n}\; dt_1 dt_2\ldots dt_n \tag{19.193}$$

$$u(t_1, t_2, \ldots, t_n) = \left(\frac{1}{2\pi}\right)^n\cdot \tag{19.194}$$

$$\int\limits_{-\infty}^{\infty}\int\limits_{-\infty}^{\infty}\ldots\int\limits_{-\infty}^{\infty} \underline{U}(j\omega_1, j\omega_1, \ldots, j\omega_n)\cdot e^{j\omega_1\cdot t_1}\cdot e^{j\omega_2\cdot t_2}\ldots\cdot e^{j\omega_n\cdot t_n}\; d\omega_1 d\omega_2\ldots d\omega_n \tag{19.195}$$

As can be seen, the n-dimensional Fourier transformation handles n different arbitrary time and frequency scales.

Fourier Transformations in more than one dimension soon become very time consuming. Using FFT mechanisms is therefore mandatory. A more-dimensional Fourier Transformation consists of many one-dimensional Fourier Transformations (1D-FFT). First, 1D-FFTs are performed for the data of the first dimension at every index of the second dimension. The results are used as input data for the second dimension that is performed the same way with respect to the third dimension. This procedure is continued until all dimensions are calculated. The following equations shows this for two dimensions.

$$
\begin{aligned}
\underline{U}_{n1,n2} &= \sum_{k_2=0}^{N_2-1}\sum_{k_1=0}^{N_1-1} u_{k_1,k_2}\cdot\exp\left(-j\cdot n_1\frac{2\pi\cdot k_1}{N_1}\right)\cdot\exp\left(-j\cdot n_2\frac{2\pi\cdot k_2}{N_2}\right) && (19.196)\\
&= \sum_{k_2=0}^{N_2-1}\exp\left(-j\cdot n_2\frac{2\pi\cdot k_2}{N_2}\right)\cdot\underbrace{\sum_{k_1=0}^{N_1-1} u_{k_1,k_2}\cdot\exp\left(-j\cdot n_1\frac{2\pi\cdot k_1}{N_1}\right)}_{\text{1D-FFT}} && (19.197)
\end{aligned}
$$

Finally, a complete n-dimensional FFT source code should be presented. It was taken from [95] and somewhat speed improved. The data is stored in "C order" (row-major format), that is

the array notation `data[n1][n2][n3]`

points to the index `data + n1 * N2*N3 + n2 * N3 + n3`

Listing 19.6: multidimensional FFT algorithm in C

```c
// ********************************************************************
// ndim   - number of dimensions
// num[]  - array containing number of complex samples for every dimension
// data[] - array containing the data samples, real and imaginary part in
//          alternating order (length: 2*sum of num[]),
//          through the array the first dimension changes least rapidly!
// isign  - is 1 to calculate FFT and -1 to calculate inverse FFT
// ********************************************************************

#define SWAP(a,b) { wr = a; a = b; b = wr; }

void mfft_radix2(int ndim, int num[], double data[], int isign)
{
  int idim, i1, m, i3rev, ip1, ip2, ip3, mmax, ifp2;
  int i, j, n, nprev, ntot;
  double tempi, tempr, wt, theta, wr, wi, wpi, wpr;

  ntot = 1;
  for(idim=0; idim<ndim; idim++) // compute total number of complex values
    ntot *= num[idim];

  nprev = 1;
  for(idim=ndim-1; idim>=0; idim--) { // main loop over the dimensions
    n = num[idim];
    ip1 = 2*nprev;
    ip2 = ip1*n;
    ip3 = 2*ntot;

    j = 0;
    for(i=ip1; i<ip2; i+=ip1) { // bit-reversal method
      m = ip2 >> 1;
      while(j >= m) {
        j -= m;
        m >>= 1;
      }
      j += m;

      if(j > i)    // was index already swapped ?
        for(i1=i; i1<=i+ip1-2; i1+=2)
          for(ifp2=i1; ifp2<=ip3; ifp2+=ip2) {
            i3rev = j+ifp2-i;
            SWAP(data[ifp2], data[i3rev]);       // swap real part
            SWAP(data[ifp2+1], data[i3rev+1]);   // swap imaginary part
          }
    }

    mmax = ip1;
    while (ip2 > mmax) { // Danielson-Lanczos algorithm
      ifp2 = mmax << 1;
      theta = isign*M_PI/(mmax/ip1);
      wpr = cos(theta);
      wpi = sin(theta);
      wr = 1.0;
      wi = 0.0;
      for(m=1; m<=mmax; m+=ip1) {
        for(i1=m; i1<=m+ip1-2; i1+=2)
          for(i=i1; i<=ip3; i+=ifp2) {
            j = i + mmax;
            tempr = wr*data[j-1] + wi*data[j];
            tempi = wr*data[j] - wi*data[j-1];
            data[j-1] = data[i-1] - tempr;
            data[j] = data[i] - tempi;
            data[i-1] += tempr;
            data[i] += tempi;
          }

        wt = wr;
        wr = wt*wpr - wi*wpi;
        wi = wi*wpr + wt*wpi;
      }
      mmax = ifp2;
    }
    nprev *= n;
  }

  if (isign == -1)          // perform inverse FFT ?
    for (i=2*ntot-1; i>=0; i--)
      data[i] /= ntot;     // normalize result
}
```

The next listings contain FFT functions for real-valued time-domain data. The principle is simple: Perform a real-valued one-dimensional FFT for the last dimension and a standard multidimensional FFT for all other dimension. As the positions of the Nyquist frequencies are not well structured, the memory size for the data is somewhat larger than half of the one for a complex FFT:

$$\text{size} = N_1 \cdot N_2 \cdot N_3 \cdot \ldots \cdot (0.5 \cdot N_n + 1) \qquad (19.198)$$

Listing 19.7: multidimensional real-valued FFT algorithm in C

```
// *****************************************************************
// ndim   - number of dimensions
// num[]  - array containing number of complex samples for every dimension
// data[] - array containing the data samples (real-valued)
//          Before and after each data block, there must be an additional
//          (empty) date -> length: num[0] + num[1] + ... + (num[ndim-1]+2)
//          through the array the first dimension changes least rapidly!
// *****************************************************************
```

```

#define SWAP(a,b) { wr = a; a = b; b = wr; }

void real_mfft_radix2(int ndim, int num[], double data[])
{
  int idim, i1, i, m, i2rev, i3rev, ip1, ip2, ip3, mmax, ifp2;
  int ibit, j, n, nprev, nrem, ntot;
  double tempi, tempr, wt, theta, wr, wi, wpi, wpr;

  ntot = 1;
  for(idim=0; idim<ndim; idim++) // compute total number of complex values
    ntot *= num[idim];

  m = 1;
  n = num[ndim-1];
  // First, perform real-valued FFT for the last dimension.
  for(i=0; i<ntot; i+=n) {
    real_fft_radix2(n, data+m);
    data[m-1] = data[m];
    data[m] = 0.0;
    data[m+n] = 0.0;
    m += n + 2;
  }

  nprev = n/2 + 1;
  // Now, perform complex-valued FFT for all other dimensions.
  for(idim=ndim-2; idim>=0; idim--) { // main loop over the dimensions
    n = num[idim];
    nrem = ntot/(n*nprev);
    ip1 = nprev << 1;
    ip2 = ip1*n;
    ip3 = ip2*nrem;
    i2rev = 1;

    for(i=1; i<=ip2; i+=ip1) { // bit-reversal method

      if(i < i2rev) {
        for(i1=i; i1<=i+ip1-2; i1+=2) {
          for(m=i1; m<=ip3; m+=ip2) {
            i3rev = i2rev+m-i;
            SWAP(data[m-1], data[i3rev-1]); // swap real part
            SWAP(data[m], data[i3rev]);      // swap imaginary part
          }
        }
      }
      ibit=ip2 >> 1;
      while (ibit >= ip1 && i2rev > ibit) {
        i2rev -= ibit;
        ibit >>= 1;
      }
      i2rev += ibit;
    }

    mmax = ip1;
    while (ip2 > mmax) { // Danielson-Lanzcos algorithm
      ifp2 = mmax << 1;
      theta = 2.0*M_PI/(ifp2/ip1);
      wpr = cos(theta);
      wpi = sin(theta);
      wr = 1.0;
      wi = 0.0;
      for(m=1; m<=mmax; m+=ip1) {
        for(i1=m; i1<=m+ip1-2; i1+=2)
          for(i=i1; i<=ip3; i+=ifp2) {
            j = i + mmax;
            tempr = wr*data[j-1] + wi*data[j];
            tempi = wr*data[j] - wi*data[j-1];
            data[j-1] = data[i-1] - tempr;
            data[j] = data[i] - tempi;
            data[i-1] += tempr;
            data[i] += tempi;
          }

        wt = wr;
        wr = wt*wpr - wi*wpi;
        wi = wi*wpr + wt*wpi;
      }
      mmax = ifp2;
    }
    nprev *= n;
  }
}
```

Listing 19.8: multidimensional real-valued inverse FFT algorithm in C

```
// ******************************************************************
// ndim   - number of dimensions
// num[]  - array containing number of complex samples for every dimension
// data[] - array with data samples, real and imaginary part in alternating
//          order -> length: num[0] + num[1] + ... + (num[ndim-1]+2)
//          through the array the first dimension changes least rapidly!
// ******************************************************************

#define SWAP(a,b) { wr = a; a = b; b = wr; }

void real_mifft_radix2(int ndim, int num[], double data[])
{
  int idim, i1, i, m, i2rev, i3rev, ip1, ip2, ip3, mmax, ifp2;
  int ibit, j, n, nprev, nrem, ntot;
  double tempi, tempr, wt, theta, wr, wi, wpi, wpr;

  ntot = 1;
  for(idim=0; idim<ndim; idim++) // compute total number of complex values
    ntot *= num[idim];

  nprev = num[ndim-1]/2 + 1;
  // Perform complex-valued FFT for all dimensions except the last one.
  for(idim=ndim-2; idim>=0; idim--) { // main loop over the dimensions
    n = num[idim];
    nrem = ntot/(n*nprev);
    ip1 = nprev << 1;
    ip2 = ip1*n;
    ip3 = ip2*nrem;
    i2rev = 1;

    for(i=1; i<=ip2; i+=ip1) { // bit-reversal method
```

```

      if(i < i2rev) {
        for(i1=i; i1<=i+ip1-2; i1+=2) {
          for(m=i1; m<=ip3; m+=ip2) {
            i3rev = i2rev+m-i;
            SWAP(data[m-1], data[i3rev-1]); // swap real part
            SWAP(data[m], data[i3rev]);     // swap imaginary part
          }
        }
      }
      ibit=ip2 >> 1;
      while (ibit >= ip1 && i2rev > ibit) {
        i2rev -= ibit;
        ibit >>= 1;
      }
      i2rev += ibit;
    }

    mmax = ip1;
    while (ip2 > mmax) { // Danielson-Lanzcos algorithm
      ifp2 = mmax << 1;
      theta = -2.0*M_PI/(ifp2/ip1);
      wpr = cos(theta);
      wpi = sin(theta);
      wr = 1.0;
      wi = 0.0;
      for(m=1; m<=mmax; m+=ip1) {
        for(i1=m; i1<=m+ip1-2; i1+=2)
          for(i=i1; i<=ip3; i+=ifp2) {
            j = i + mmax;
            tempr = wr*data[j-1] + wi*data[j];
            tempi = wr*data[j] - wi*data[j-1];
            data[j-1] = data[i-1] - tempr;
            data[j] = data[i] - tempi;
            data[i-1] += tempr;
            data[i] += tempi;
          }

        wt = wr;
        wr = wt*wpr - wi*wpi;
        wi = wi*wpr + wt*wpi;
      }
      mmax = ifp2;
    }
    nprev *= n;
  }

  m = 1;
  n = num[ndim-1];
  nrem = ntot / n;
  // Finally, perform real-valued FFT for the last dimension.
  for(i=0; i<ntot; i+=n) {
    data[m] = data[m-1];
    data[m-1] = 0.0;
    data[m+n] = 0.0;
    real_ifft_radix2(n, data+m);
    for(j=0; j<n; j++)
      data[m+j] /= nrem;  // normalize result
    m += n + 2;
  }
}
```

19.4 Newton's divided differences interpolation

The divided difference $f[x_0, x_1, \ldots, x_n]$ of a function $f(x)$ on the points $x_0, x_1, \ldots, x_n$ is defined by:

$$\begin{aligned} f[x_0] &= f(x_0) \\ f[x_0, x_1] &= \frac{f(x_0) - f(x_1)}{x_0 - x_1} \\ f[x_0, x_1, x_2] &= \frac{f[x_0, x_1] - f[x_1, x_2]}{x_0 - x_2} \\ &\ldots \\ f[x_0, x_1, \ldots, x_n] &= \frac{f[x_0, \ldots, x_{n-1}] - f[x_1, \ldots, x_n]}{x_0 - x_n} \end{aligned} \tag{19.199}$$

Two important properties should be mentioned shortly:

- linearity

  $$(a \cdot f + b \cdot g)[x_0, \ldots, x_n] = a \cdot f[x_0, \ldots, x_n] + b \cdot g[x_0, \ldots, x_n] \tag{19.200}$$

  with a, b being real numbers and $f(x), g(x)$ being functions.

- symmetry
  If $y_0, \ldots, y_n$ is a permutation of $x_0, \ldots, x_n$, then it holds

  $$f[x_0, \ldots, x_n] = f[y_0, \ldots, y_n] \tag{19.201}$$

An application for these formulae is the numerical derivation. The generalization of the mean value theorem may be written as follows:

$$f[x_0, x_1, \ldots, x_n] = \frac{f^{(n)}(u)}{n!} \qquad \text{for some} \qquad u \in [x_0, x_n] \tag{19.202}$$

This also implies that the divided difference can be used as an approximation for the numerical derivation:

$$\frac{\partial^n f}{\partial x^n} \approx n! \cdot f[x_0, x_1, \ldots, x_n] \tag{19.203}$$

Another application for these formulae is the interpolation (or extrapolation) of function values with k-order accuracy:

$$f(x) \approx \sum_{i=0}^{k} f[x_0, x_1, \ldots, x_i] \cdot \prod_{j=0}^{i-1} (x - x_j) \tag{19.204}$$

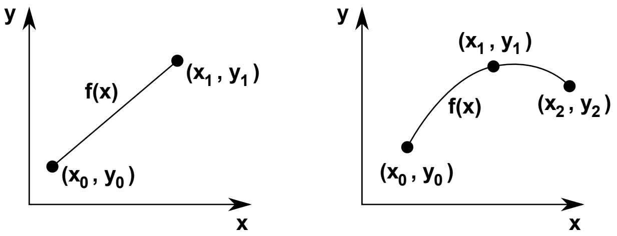


Figure 19.6: linear (left) and quadratic (right) interpolation

For a linear interpolation (order 1) this yields:

$$f_1(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0} \cdot (x - x_0) = f(x_1) + \frac{f(x_1) - f(x_0)}{x_1 - x_0} \cdot (x - x_1) \tag{19.205}$$

For a quadratic interpolation (order 2) it yields:

$$f_2(x) = f(x_0) + \frac{f(x_1) - f(x_0)}{x_1 - x_0} \cdot (x - x_0) + \frac{\frac{f(x_2)-f(x_1)}{x_2-x_1} - \frac{f(x_1)-f(x_0)}{x_1-x_0}}{x_2 - x_0} \cdot (x - x_0) \cdot (x - x_1) \tag{19.206}$$

Setting the derivative of $f_2(x)$ to zero yields the coordinate x_{lm} of the local minimum or maximum:

$$\begin{aligned} x_{lm} &= \frac{x_1 + x_0}{2} - \frac{1}{2} \cdot \frac{x_2 - x_0}{m - 1} \\ \text{with} \quad m &= \frac{x_1 - x_0}{x_2 - x_1} \cdot \frac{f(x_2) - f(x_1)}{f(x_1) - f(x_0)} \end{aligned} \tag{19.207}$$

Appendix A

Bibliography

[1] A. Blum, *Skriptum zur Vorlesung Elektronik IV über Schaltungssimulation.* Universität des Saarlandes: Universitas Saraviensis, 2001.

[2] K. Kurokawa, “Power Waves and the Scattering Matrix,” *IEEE Transactions on Microwave Theory and Techniques*, pp. 194–202, Mar. 1965.

[3] R. C. Compton and D. B. Rutledge, “Perspectives in Microwave Circuit Analysis,” in *Proceedings of the 32nd Midwest Symposium on Circuits and Systems*, vol. 2. IEEE, Aug. 1969, pp. 716–718.

[4] D. Bockelman and W. Eisenstadt, “Combined Differential and Common-Mode Scattering Parameters: Theory and Simulation,” *IEEE Transactions on Microwave Theory and Techniques*, p. 1530, July 1995.

[5] M. L. Edwards and J. H. Sinsky, “A New Criterion for Linear 2-Port Stability Using a Single Geometrically Derived Parameter,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 12, pp. 2303–2311, Dec. 1992.

[6] H.-J. Michel, *Zweitoranalyse mit Leitungswellen.* Teubner, 1981.

[7] S. W. Wedge and D. B. Rutledge, “Wave Techniques for Noise Modeling and Measurement,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 11, pp. 2004–2012, Nov. 1992.

[8] H. Hillbrand and P. H. Russer, “An Efficient Method for Computer Aided Noise Analysis of Linear Amplifier Networks,” *IEEE Transactions on Circuits and Systems*, vol. 23, no. 4, pp. 235–238, Apr. 1976.

[9] S. Voigtmann, *General Linear Methods for Integrated Circuit Design.* PhD thesis, Humboldt University Berlin, 2006.

[10] Y. Zhou, *Stable High Order Methods for Circuit Simulation.* PhD thesis, University of Ottawa, 2011.

[11] C. W. Gear, “Simultaneous Numerical Solution of Differential-Algebraic Equations,” *IEEE Transactions on Circuit Theory*, vol. 18, no. 1, pp. 89–95, Jan. 1971.

[12] S. Skelboe, “The Control of Order and Steplength for Backward Differential Methods,” *BIT*, vol. 17, pp. 91–107, 1977.

[13] J. Carroll, “A Composite Integration Scheme for the Numerical Solution of Systems of Ordinary Differential Equations,” *Journal of Computational and Applied Mathematics*, vol. 25, pp. 1–13, 1989.

[14] H. Z. Quanrong Wanga, "On different numerical inverse Laplace methods for solute transport problems," *Advances in Water Resources, Elsevier Ltd.*, 2014.

[15] K. M. S.K. Park, "Random Number Generators: Good Ones Are Hard to Find," *Communications of the ACM*, vol. 31, no. 10, Oct. 1988.

[16] D. G. Carta, "Two Fast Implementations of the 'Minimal Standard' Random Number Generator," *Comm. ACM*, vol. 33, no. 1, pp. 87–88, Jan. 1990.

[17] L. M. et al., "Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection," *Biomed Eng Online*, vol. 6, no. 23, 2007.

[18] S. Maas, *Nonlinear Microwave and RF Circuits*, second edition ed. www.artechhouse.com, 2003.

[19] S. Wang, *Efficient Harmonic Balance Analysis of Microwave Circuits Using Iterative Matrix Solvers and Newton Updating.* North Carolina State University: Master Thesis, 1999.

[20] C.-R. Chang, M. B. Steer, S. Martin, and E. Reese, "Computer-Aided Analysis of Free-Running Microwave Oscillators," *IEEE Transactions on Microwave Theory and Techniques*, vol. 39, no. 10, pp. 1735–1744, Oct. 1991.

[21] J. Roychowdhury, D. Long, and P. Feldmann, "Cyclostationary Noise Analysis of Large RF Circuits with Multitone Excitations," *IEEE Journal of Solid-State Circuits*, pp. 324–336, Mar. 1998.

[22] V. Rizzoli and A. Neri, "State of the Art and Present Trends in Nonlinear Microwave CAD Techniques," *IEEE Transactions on Microwave Theory and Techniques*, pp. 343–365, Feb. 1988.

[23] J. T. Ivan Krivy, "The Controlled Random Search Algorithm in Optimizing Regression Models," *The Statistical Software Newsletter*, pp. 229–234, Feb. 1995.

[24] P. Storn, "Differential Evolution - a simple and efficient adaptive scheme for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341–359, 1997.

[25] S. Das, A. Abraham, U. K. Chakraborty, and A. Konar, "Differential Evolution Using a Neighborhood-based Mutation Operator," *IEEE Transactions on Evolutionary Computation*, vol. 13, pp. 526–553, 2009.

[26] T. Edwards, *Foundations for Microstrip Circuit Design*, 2nd ed. John Wiley & Sons, 1991.

[27] T. O. Grosch and L. A. Carpenter, "Two-Port to Three-Port Noise-Wave Transformation for CAD Applications," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, no. 9, pp. 1543–1548, Sept. 1993.

[28] W. M. Leach, "Loudspeaker Voice-Coil Inductance Losses: Circuit Models, Parameter Estimation, and Effect on Frequency Response," *Journal of the Audio Engineering Society*, vol. 50, no. 6, pp. 442–449, June 2002.

[29] P. B. Weil and L. P. McNamee, "Simulation of Excess Phase in Bipolar Transistors," *IEEE Transactions on Circuits and Systems*, vol. 25, no. 2, pp. 114–116, Feb. 1978.

[30] H. Shichman and D. A. Hodges, "Modeling and Simulation of Insulated-Gate Field-Effect Transistor Switching Circuits," *IEEE Journal of Solid-State Circuits*, vol. 3, no. 3, pp. 285–289, Sept. 1968.

[31] J. E. Meyer, "MOS Models and Circuit Simulation," *RCA Review*, vol. 32, pp. 42–63, Mar. 1971.

[32] A. Matiss, *Entwurf und Realisierung neuartiger Schaltungskonzepte mit Resonanztunneldioden.* Göttingen: Cuvillier Verlag Göttingen, 2008.

[33] S. Dindo, D. Kennedy, and J. Wareberg, "Software Simplifies Large-Signal Transistor Modeling," *Microwaves & RF*, Oct. 1999.

[34] H. S. et al., "GaAs FET device and circuit simulation in SPICE," *IEEE Transactions on Electronic Devices*, vol. ED34, pp. 160–169, Feb. 1987.

[35] H. A. Wheeler, "Transmission-Line Properties of Parallel Strips Separated by a Dielectric Sheet," *IEEE Transactions on Microwave Theory and Techniques*, vol. 13, no. 2, pp. 172–185, Mar. 1965.

[36] M. V. Schneider, "Microstrip Lines for Microwave Integrated Circuits," *The Bell System Technical Journal*, vol. 48, pp. 1421–1444, May 1969.

[37] E. Hammerstad and Ø. Jensen, "Accurate Models for Microstrip Computer-Aided Design," *Symposium on Microwave Theory and Techniques*, pp. 407–409, June 1980.

[38] H. A. Wheeler, "Transmission-Line Properties of a Strip on a Dielectric Sheet on a Plane," *IEEE Transactions on Microwave Theory and Techniques*, vol. 25, no. 8, pp. 631–647, Aug. 1977.

[39] M. Kirschning and R. H. Jansen, "Accurate Model for Effective Dielectric Constant of Microstrip with Validity up to Millimeter-Wave Frequencies," *Electronics Letters*, vol. 8, no. 6, pp. 272–273, Mar. 1982.

[40] R. H. Jansen and M. Kirschning, "Arguments and an accurate Model for the Power-Current Formulation of Microstrip Characteristic Impedance," *Archiv für Elektronik und Übertragungstechnik (AEÜ)*, vol. 37, pp. 108–112, 1983.

[41] E. Yamashita, K. Atsuki, and T. Ueda, "An Approximate Dispersion Formula of Microstrip Lines for Computer Aided Design of Microwave Integrated Circuits," *IEEE Transactions on Microwave Theory and Techniques*, vol. 27, pp. 1036–1038, Dec. 1979.

[42] M. Kobayashi, "A Dispersion Formula Satisfying Recent Requirements in Microstrip CAD," *IEEE Transactions on Microwave Theory and Techniques*, vol. 36, no. 8, pp. 1246–1250, Aug. 1988.

[43] W. J. Getsinger, "Microstrip Dispersion Model," *IEEE Transactions on Microwave Theory and Techniques*, vol. 21, no. 1, pp. 34–39, Jan. 1973.

[44] ——, "Microstrip Characteristic Impedance," *IEEE Transactions on Microwave Theory and Techniques*, vol. 27, no. 5, p. 293, Apr. 1979.

[45] ——, "Measurement and Modeling of the Apparent Characteristic Impedance of Microstrip," *IEEE Transactions on Microwave Theory and Techniques*, vol. 31, no. 8, pp. 624–632, Aug. 1983.

[46] T. C. Edwards and R. P. Owens, "2-18-GHz Dispersion Measurements on 10-100-Ω Microstrip Lines on Sapphire," *IEEE Transactions on Microwave Theory and Techniques*, vol. 24, no. 8, pp. 506–513, Aug. 1976.

[47] P. Pramanick and P. Bhartia, "An Accurate Description of Dispersion in Microstrip," *Microwave Journal*, pp. 89–96, Dec. 1983.

[48] M. V. Schneider, "Microstrip Dispersion," in *Proceedings of the IEEE, Letters*, vol. 60, Jan. 1972, pp. 144–146.

[49] M. D. Abouzahra and L. Lewis, “Radiation from Microstrip Discontinuities,” *IEEE Transactions on Microwave Theory and Techniques*, pp. 722–723, Aug. 1979.

[50] M. Kirschning and R. H. Jansen, “Accurate Wide-Range Design Equations for the Frequency-Dependent Characteristic of Parallel Coupled Microstrip Lines,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 32, no. 1, pp. 83–90, Jan. 1984.

[51] ——, “Corrections to ”Accurate Wide-Range Design Equations for the Frequency-Dependent Characteristic of Parallel Coupled Microstrip Lines”,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 33, no. 3, p. 288, Mar. 1985.

[52] W. J. Getsinger, “Dispersion of Parallel-Coupled Microstrip,” *IEEE Transactions on Microwave Theory and Techniques*, pp. 144–145, Mar. 1973, Short Papers.

[53] R. H. Jansen, “High-Speed Computation of Single and Coupled Microstrip Parameters Including Dispersion, High-Order Modes, Loss and Finite Strip,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 26, no. 2, pp. 75–82, Feb. 1978.

[54] M. Kirschning, R. H. Jansen, and N. H. L. Koster, “Accurate Model for open end effect of microstrip lines,” *Electronics Letters*, vol. 17, no. 3, pp. 123–125, Feb. 1981.

[55] E. Hammerstad, “Computer-Aided Design of Microstrip Couplers with Accurate Discontinuity Models,” *Symposium on Microwave Theory and Techniques*, pp. 54–56, June 1981.

[56] M. Kirschning, *Entwicklung von Näherungsmodellen für den rechnergestützten Entwurf von hybriden und monolithischen Schaltungen in Mikrostreifenleitungstechnik.* Dissertation, Universität Duisburg, 1984.

[57] M. Kirschning, R. H. Jansen, and N. H. L. Koster, “Measurement and Computer-Aided Modeling of Microstrip Discontinuities by an Improved Resonator Method,” *IEEE MTT-S International Microwave Symposium Digest*, pp. 495–497, May 1983.

[58] R. J. P. Douville and D. S. James, “Experimental Study of Symmetric Microstrip Bends and Their Compensation,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 26, pp. 175–181, Mar. 1978.

[59] K. C. Gupta, R. Garg, and I. J. Bahl, *Microstrip Lines and Slotlines.* Artech House, Inc., 1979.

[60] R. K. Hoffmann, *Integrierte Mikrowellenschaltungen*, Elektrische Grundlagen, Dimensionierung, technische Ausführung, Technologien ed. Berlin Heidelberg: Springer Verlag, 1983.

[61] K. C. Gupta, R. Garg, I. J. Bahl, and P. Bhartia, *Microstrip Lines and Slotlines*, 2nd ed. Artech House, Inc., 1996.

[62] R. Garg and I. J. Bahl, “Microstrip discontinuities,” *International Journal of Electronics*, vol. 45, no. 1, pp. 81–87, 1978.

[63] R. J. Akello, B. Easter, and I. M. Stephenson, “Equivalent circuit of the asymmetric crossover junction,” *Electronics Letters*, vol. 13, no. 4, pp. 117–118, Feb. 1977.

[64] P. Silvester and P. Benedek, “Microstrip Discontinuity Capacitances for Right-Angle Bends, T Junctions, and Crossings,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 21, no. 5, pp. 341–346, May 1973.

[65] A. Gopinath, A. F. Thomson, and I. M. Stephenson, “Equivalent Circuit Parameters of Microstrip Step Change in Width and Cross Junctions,” *IEEE Transactions on Microwave Theory and Techniques*, pp. 142–144, Mar. 1976, Short Papers.

[66] B. Easter, "The Equivalent Circuit of Some Microstrip Discontinuities," *IEEE Transactions on Microwave Theory and Techniques*, vol. 23, no. 8, pp. 655–660, Aug. 1975.

[67] S. L. March, "Analyzing Lossy Radial-Line Stubs," *IEEE Transactions on Microwave Theory and Techniques*, vol. 33, no. 3, pp. 269–271, Mar. 1985.

[68] N. Dib, J. Ababneh, and A. Omar, "CAD Modeling of Coplanar Waveguide Interdigital Capacitor," *Wiley Periodicals, Inc.*, 2005.

[69] M. E. Goldfarb and R. A. Pucel, "Modeling Via Hole Grounds in Microstrip," *IEEE Microwave and Guided Wave Letters*, vol. 1, no. 6, pp. 135–137, June 1991.

[70] I. Bahl and P. Barthia, *Microwave Solid State Circuit Design*, 2nd ed. Wiley Interscience, 2003.

[71] Wikibook. (2006, Nov.) Electronics/Inductors. [Online]. Available: http://en.wikibooks.org/w/index.php?title=Electronics/Inductors

[72] R. E. Collin, *Foundations for Microwave Engineering*, 2nd ed. New York: Mc Graw-Hill, 1992.

[73] M. Lavrentiev and B. Chabat, *Méthodes de la théorie des fonctions d'une variable complexe*, Moscow, 1972.

[74] M. Y. Frankel, S. Gupta, J. A. Valdmanis, and G. A. Mourou, "Terahertz Attenuation and Dispersion Characteristics of Coplanar Transmission Lines," *IEEE Transactions on Microwave Theory and Techniques*, vol. 39, no. 6, pp. 910–916, June 1991.

[75] S. Gevorgian, T. Martinsson, A. Deleniv, E. Kollberg, and I. Vendik, "Simple and accurate dispersion expression for the effective dielectric constant of coplanar waveguides," in *Proceedings of Microwaves, Antennas and Propagation*, vol. 144, no. 2. IEE, Apr. 1997, pp. 145–148.

[76] G. H. Owyang and T. T. Wu, "The Approximate Parameters of Slot Lines and Their Complement," *IRE Transactions on Antennas and Propagation*, pp. 49–55, Jan. 1958.

[77] G. Ghione, "A CAD-Oriented Analytical Model for the Losses of General Asymmetric Coplanar Lines in Hybrid and Monolithic MICs," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, no. 9, pp. 1499–1510, Sept. 1993.

[78] K. Beilenhoff, H. Klingbeil, W. Heinrich, and H. L. Hartnagel, "Open and Short Circuits in Coplanar MMIC's," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, no. 9, pp. 1534–1537, Sept. 1993.

[79] W. J. Getsinger, "End-Effects in Quasi-TEM Transmission Lines," *IEEE Transactions on Microwave Theory and Techniques*, vol. 41, no. 4, pp. 666–671, Apr. 1993.

[80] ——, "Circuit Duals on Planar Transmission Media," *IEEE MTT-S International Microwave Symposium Digest*, pp. 154–156, 1983.

[81] C. Sinclair and S. J. Nightgale, "An Equivalent Circuit Model for the Coplanar Waveguide Step Discontinuity," *IEEE MTT-S International Microwave Symposium Digest*, pp. 1461–1464, Sept. 1992.

[82] F. D. Paolo, *Networks and Devices Using Planar Transmission Lines.* CRC Press LLC, 2000.

[83] B. C. Wadell, *Transmission Line Design Handbook.* Boston London: Artech House, Inc., 1991.

[84] M. V. Schneider, “Dielectric Loss in Integrated Microwave Circuits,” 1969.

[85] Prof. Dr.-Ing. Klaus Petermann, *Arbeitsbltter zur Vorlesung Hochfrequenztechnik I.* Berlin: Technische Universität Berlin, 1997.

[86] P. Lefferson, “Twisted Magnet Wire Transmission Line,” *IEEE Transactions on Parts, Hybrids, and Packaging*, vol. PHP-7, no. 4, pp. 148–154, Dec. 1971.

[87] M. J. L. Jia-Sheng Hong, *Microstrip filters for RF/microwave applications.* New York: John Wiley & Sons, Inc., 2001.

[88] T. Liebig, A. Rennings, S. Held, and D. Erni, “openEMS - A free and open source equivalent-circuit (EC) FDTD simulation platform supporting cylindrical coordinates suitable for the analysis of traveling wave MRI applications,” *International Journal of Numerical Modelling: Electronic Networks, Devices and Fields*, Dec. 2012.

[89] J. L. Volakis, A. Chatterjee, and L. C. Kempel, *Finite Element Method for Electromagnetics.* New York: John Wiley & Sons, Inc., 1998.

[90] R. M. Jin-Fa Lee, “A Note on the Application of Edge-Elements for Modeling Three-Dimensional Inhomogeneously-Filled Cavities,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 9, pp. 1767–1773, Sept. 1992.

[91] J. A. Dobrowolski, *Introduction to Computer Methods for Microwave Circuit Analysis and Design.* Warsaw University of Technology: Artech House, 1991.

[92] W. Weißgerber, *Elektrotechnik für Ingenieure*, Ausgleichsvorgänge, Fourieranalyse, Vierpoltheorie ed. Vieweg Verlag, 1999, vol. 3.

[93] Kenneth Kundert. (2012, Nov.) Sparse1.4 package. [Online]. Available: http://sparse.sourceforge.net/description.html

[94] Timothy A. Davis, Eka Palamadai. (2012, Nov.) KLU package. [Online]. Available: http://faculty.cse.tamu.edu/davis/

[95] W. H. Press, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, 1988-1992.

[96] A. I. Zverev, *Handbook of Filter Synthesis.* New York: John Wiley & Sons, 1967.

[97] L. Thede, *Analog and Digital Filter Design using C.* New Jersey: Prentice Hall, 1996.

[98] A. S. Sedra and P. O. Brackett, *Filter Theory and Design: Active and Passive*, ser. Matrix Series in Circuits and Systems, A. P. Sage, Ed. Portland, Oregon: Matrix Publishers, Inc., 1978.

[99] M. Abramowitz and I. A. Segun, *Handbook of Mathematical Functions with Formulas, Graphs, and Mathematical Tables.* New York: Dover Publications, Inc., May 1968.

[100] H. Stöcker, *Taschenbuch mathematischer Formeln und moderner Verfahren*, 3rd ed. Frankfurt am Main: Verlag Harri Deutsch, 1995.

[101] J. A. Mackenzie, *The Numerical Solution of Ordinary Differential Equations.* University of Strathclyde: Department of Mathematics, 2001.

[102] A. Sangiovanni-Vincentelli and A. Nardi, *Computer-Aided Verification of Electronic Circuits and Systems.* University of Berkeley: Department of Electrical Engineering and Computer Sciences, 2002.

[103] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, and H. V. der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods.* Philadelphia, PA: SIAM, 1994.

[104] J. Berkner, *Kompaktmodelle für Bipolartransistoren*, Praxis der Modellierung, Messung und Parameterbestimmung - SGP, VBIC, HICUM und MEXTRAM ed. expert Verlag, 2002.

[105] E. E. E. Hoefer and H. Nielinger, *SPICE - Analyseprogramm für elektronische Schaltungen.* Berlin Heidelberg: Springer Verlag, 1985.

[106] S. M. Sze, *Semiconductor Devices*, Physics and Technology ed. New York: John Wiley & Sons, 1985.

[107] G. Engeln-Müllges and F. Uhlig, *Numerical Algorithms with C.* Berlin: Springer Verlag, Feb. 1996.

[108] G. H. Golub and C. F. van Loan, *Matrix Computations*, 3rd ed. London: The Johns Hopkins University Press, 1996.

[109] G. Massobrio and P. Antognetti, *Semiconductor Device Modeling with SPICE*, 2nd ed. New York: Mc Graw-Hill, 1993.

[110] J. F. Hart and et al., *Computer Approximations*, ser. The SIAM Series in Applied Mathematics. New York: John Wiley & Sons, Inc., 1968.

[111] National Institute of Standards and Technology. (2003, Dec.) Fundamental Physical Constants from NIST. [Online]. Available: http://physics.nist.gov/cuu/Constants/index.html

[112] A. C. M. de Queiroz, "Compact Nodal Analysis With Controlled Sources Modeled by Operational Amplifiers," in *Proceedings of the 38th Midwest Symposium on Circuits and Systems*, vol. 2. IEEE, Aug. 1995, pp. 1205–1208.

[113] L. M. Wedepohl and L. Jackson, "Modified nodal analysis: an essential addition to electrical circuit theory and analysis," *Engineering Science and Education Journal*, vol. 11, pp. 84–92, June 1992.

[114] V. A. Monaco and P. Tiberio, "Computer-Aided Analysis of Microwave Circuits," *IEEE Transactions on Microwave Theory and Techniques*, vol. 22, no. 3, pp. 249–263, Mar. 1974.

[115] E. Yamashita, K. Atsuki, and T. Hirahata, "Microstrip Dispersion in a Wide-Frequency Range," *IEEE Transactions on Microwave Theory and Techniques*, vol. 29, no. 6, pp. 610–611, June 1981.

[116] M. Kirschning, R. H. Jansen, and N. H. L. Koster, "Measurement and Computer-Aided Modeling of Microstrip Discontinuities by an Improved Resonator Method," *Symposium on Microwave Theory and Techniques*, pp. 495–497, May 1983.

[117] M. Kobayashi, "Important Role of Inflection Frequency in the Dispersive Property of Microstrip Lines," *IEEE Transactions on Microwave Theory and Techniques*, vol. 30, no. 11, pp. 2057–2059, Nov. 1982.

[118] P. Bhartia and P. Pramanick, "A New Microstrip Dispersion Model," *IEEE Transactions on Microwave Theory and Techniques*, vol. 32, no. 10, pp. 1379–1384, Oct. 1984.

[119] M. Kobayashi, "Frequency Dependent Characteristics of Microstrips on Anisotropic Substrates," *IEEE Transactions on Microwave Theory and Techniques*, vol. 30, no. 11, pp. 2054–2057, Nov. 1982.

[120] H. A. Atwater, "Tests of Microstrip Dispersion Formulas," *IEEE Transactions on Microwave Theory and Techniques*, vol. 36, no. 3, pp. 619–621, Mar. 1988.

[121] E. J. Denlinger, “Losses of Microstrip Lines,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 28, no. 6, pp. 513–522, June 1980.

[122] H. A. Wheeler, “Transmission-Line Properties of Parallel Wide Strips by a Conformal-Mapping Approximation,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 12, no. 3, pp. 280–289, May 1964.

[123] R. P. Owens, J. E. Aitken, and T. C. Edwards, “Quasi-Static Characteristics of Microstrip on an Anisotropic Sapphire Substrate,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 24, no. 8, pp. 499–505, Aug. 1976.

[124] M. Kirschning, R. H. Jansen, and N. H. L. Koster, “Coupled Mircrostrip Parallel-Gap Model for Improved Filter and Coupler Design,” *Electronics Letters*, vol. 19, no. 10, pp. 377–379, May 1983.

[125] G. Ghione and C. U. Naldi, “Coplanar Waveguides for MMIC Applications: Effect of Upper Shielding, Conductor Backing, Finite-Extent Ground Planes, and Line-to-Line Coupling,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 35, no. 3, pp. 260–267, Mar. 1987.

[126] ——, “Analytical Formulas for Coplanar Lines and Monolithic MICs,” *Electronics Letters*, vol. 20, no. 4, pp. 179–181, Feb. 1984.

[127] S. S. Bedair and I. Wolff, “Fast, Accurate and Simple Approximate Analytic Formulas for Calculating the Parameters of Supported Coplanar Waveguides for (M)MIC’s,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 1, pp. 41–48, Jan. 1992.

[128] J. L. B. Walker, “A Survey of European Activity on Coplanar Waveguide,” *IEEE MTT-S International Microwave Symposium Digest*, pp. 693–696, 1993.

[129] C. Veyres and V. F. Hanna, “Extension of the application of conformal mapping techniques to coplanar lines with finite dimensions,” *International Journal of Electronics*, vol. 48, no. 1, pp. 47–56, 1980.

[130] P. Russer and S. Müller, “Noise Analysis of Microwave Circuits with General Topology,” *Symposium on Microwave Theory and Techniques*, pp. 1481–1484, 1992.

[131] ——, “Noise Analysis of Microwave Circuits with General Topology and Arbitrary Representation,” *Asia-Pacific Microwave Conference*, pp. 819–822, 1992.

[132] H. Hillbrand and P. H. Russer, “Correction to ”An Efficient Method for Computer Aided Noise Analysis of Linear Amplifier Networks”,” *Letters to the Editor*, p. 691, July 1976.

[133] R. A. York and R. C. Compton, “Experimental Evaluation of Existing CAD Models for Microstrip Dispersion,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 38, no. 3, pp. 327–328, Sept. 1990.

[134] M. Kobayashi and N. Sawada, “Analysis and Synthesis of Tapered Microstrip Transmission Lines,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 40, no. 8, pp. 1642–1646, Aug. 1992.

[135] H. J. Carlin, “A Simplified Circuit Model for Microstrip,” *IEEE Transactions on Microwave Theory and Techniques*, pp. 589–591, Apr. 1973, Short Papers.

[136] H. J. Carlin and P. P. Civalleri, “A Coupled-Line Model for Dispersion in Parallel-Coupled Microstrips,” *IEEE Transactions on Microwave Theory and Techniques*, pp. 444–446, May 1975.

[137] R. Garg and I. J. Bahl, “Characteristics of Coupled Microstriplines,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 27, no. 7, pp. 700–705, July 1979.

[138] H. Cory, “Dispersion Characteristics of Microstrip Lines,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 29, no. 1, pp. 59–61, Jan. 1981.

[139] N. G. Alexopoulos and S.-C. Wu, “Frequency-Independent Equivalent Circuit Model for Microstrip Open-End and Gap Discontinuities,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 42, no. 7, pp. 1268–1272, July 1994, Short Papers.

[140] P. Silvester and P. Benedek, “Equivalent Capacitances of Microstrip Open Circuits,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 20, no. 8, pp. 511–516, Aug. 1972.

[141] W. J. R. Hoefer, “Equivalent Series Inductivity of a Narrow Transverse Slit in Microstrip,” *IEEE Transactions on Microwave Theory and Techniques*, vol. 25, no. 10, pp. 822–824, Oct. 1977.

[142] C. Descharles, C. Algani, B. Mercier, and G. Alquie, “Physical and Electrical Modelling of Bonding Wires up to 110 GHz,” *33rd European Microwave Conference*, vol. 2, pp. 639–624, 2003.