

Companion Models for Basic Non-Linear and Transient Devices

Steven Herbst, Antoine Levitt

August 11, 2008

1 Introduction

1.1 Linear DC analysis

1.1.1 Nodal analysis

The simplest kind of circuit simulation deals with constant sources and resistors. There are a number of different techniques to formulate the network equations in a general way, the most convenient for computer simulation being nodal analysis (NA). This method formulates equations in the $GV = I$ matrix form, which for current sources and resistors is a straightforward application of the Kirchoff current law (KCL) at each node. V is the unknown vector of node voltages, I the vector of current sources and G the conductance matrix.

As an interesting side effect of this formulation, we can derive the equations on a per-component basis : each component “stamps” itself in the G matrix and I right-hand side, and the linear system is then solved for V . For instance, a resistor between node i and node j contributes to the current flowing to node j by $\frac{1}{R}(v_j - v_i)$ and to the current flowing to node i by the opposite of the same amount. So its “stamp” is four entries in the G matrix : $1/R$ in (i, i) and (j, j) , and $-1/R$ in (i, j) and (j, i) . The stamp for the current source is derived in the same way : a current source of known current J between node i and j adds J to entry number i of I , and $-J$ to entry number j of I .

The algorithm for nodal analysis is :

- Iterate through each component, and stamp them according to their type in the global G matrix and I vector
- Solve the linear system (by any direct or iterative method ; we use LU decomposition) for the V vector

1.1.2 Modified nodal analysis

Using NA, we can simulate circuits comprised of resistors and current sources. For the other basic building block, the voltage source, an artifice is required.

Since we can not incorporate the missing equations directly, we add another unknown for every voltage source: this is known as modified nodal analysis (MNA) and is at the core of edacious. The idea is to split the unknown vector (called x in the simulation) and the right-hand side (called z) in two parts : the first corresponds to classical nodal analysis, the second corresponds to the equations of voltage sources.

The second part of the unknowns are the currents flowing through the voltage sources. A voltage source of known voltage V between nodes i and j adds the unknown $I_{i,j}$ to the unknowns vector, contributes to the existing equations by adding $I_{i,j}$ to i_i and $-I_{i,j}$ to i_j , and adds the new equation $V_j - V_i = V$. These equations, like the previous ones, can be incorporated by stamping.

The new matrix relation becomes $Ax = z$:

- z is (z_{NA}, z_{MNA}) , z_{NA} being the stamps of current sources, and z_{MNA} the stamps of voltage sources.
- x is (x_{NA}, x_{MNA}) , x_{NA} being the voltages at the nodes, and x_{MNA} the currents flowing through the voltage sources.
- A is the block matrix $(G, C; B, D)$, G being the admittance matrix of NA, and B, C, D representing various constitutive equations pertaining to voltage sources.

1.2 Non-linear DC analysis

To simulate real circuits containing transistors and diodes, we are interested in simulating components having arbitrary i - v relations. We use the standard Newton method, described in any numerical analysis textbook. The derivation of how the multidimensional Newton method leads to our algorithm is a bit tedious, but here is an intuitive explanation : the Newton method works by linearizing the equation around the operating point X_n , solving the linearized equation to obtain X_{n+1} , and iterating until convergence (being when X_n is *near* X_{n+1} ; the precise definition of *near* involving a compromise between speed and accuracy). In circuits, we linearize the i - v characteristic around point v_n : $i = i(v_n) + (dv/di)(v - v_n)$. That allows us to form companion models describing the linearized component, which we can stamp. The equation is then solved for another operating point, and the cycle continues until a stable answer is found.

1.3 Transient analysis

Transient analysis uses pretty much the same idea as non-linear analysis. We have to take into account are energy storage components, namely capacitors and inductors. For example, the constitutive relation for the inductor is $u = L di/dt$, or equivalently $i(t + \Delta t) = i(t) + 1/L \int_t^{t+\Delta t} u(x) dx$. We then use an approximation method (we implement single-step methods, namely Forward Euler, Backward Euler and Trapezoidal Rule) to compute this integral. This

leads to a formula describing the behavior of the component (i.e. a relationship between $i(t + \Delta t)$ and $v(t + \Delta t)$), which we can then convert to a companion model and use to compute the solution, using a Newton loop. This gives us a new operating point to use for the next time step.

1.4 Summary

The structure of the final algorithm is as follow :

```

Transient loop :
while the simulation is not over
  Formulate companion models for energy storage
  components, using current operating point
  Newton loop :
  while the convergence is not achieved
    Formulate companion models for non-linear components,
    using current operating point
    Solve for new operating point
  end while
end while

```

2 Non-linear components

2.1 PN Diode

As a two-terminal device with only one distinct operating region, the PN diode was a natural starting point for non-linear simulation. The current model does not include parasitic capacitances, but this is planned for the future.

The i - v relation describing a diode is the following:

$$i(v) = I_S(e^{v/V_t} - 1) \quad (1)$$

where I_S is the reverse saturation current and V_t is the thermal voltage (kT/q). From this, we derive the small-signal conductance:

$$g = \frac{di}{dv} = (I_S/V_t)e^{v/V_t} \approx i(v)/V_t \quad (2)$$

The error committed in the approximation is I_S/V_t , which for common diodes is around $10^{-12}S$. This is negligible for ordinary applications and allows us to avoid computing the exp function twice.

In simulation, we use the Newton-Raphson method to solve circuits with non-linear elements. The method works by linearizing the constitutive equations, solving for a new operating point, and iterating until convergence. The linearized curve around operating point v_n is :

$$i_{lin}(v) = i(v_n) + (v - v_n)\frac{di}{dv}(v_n) \quad (3)$$

By rearranging this equation, we find the i - v relation $i_{lin} = (i(v_n) - gv_n) + gv$. Hence the appropriate companion model for a diode is a conductance g in parallel with a current source $i(v_n) - gv_n$, not simply $i(v_n)$, as one might suspect. The result is shown in Figure 1.

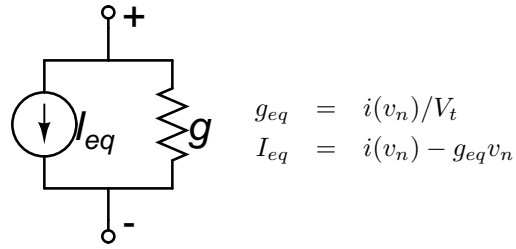


Figure 1: Linear companion model for diode

2.2 Metal-Oxide-Silicon Field-Effect Transistor

The next simplest non-linear devices are n- and p-channel MOSFETs, as no current flows into the gate, and the non-linear source-to-drain port depends on only two parameters, v_{GS} and v_{DS} . The model is complicated slightly, however, by the fact that MOSFETs have three distinct operating regions. For brevity, we derive only the results for n-channel MOSFETs. Similar results for p-channel MOSFETs are presented at the end of this subsection.

2.2.1 Cutoff: $v_{GS} - V_T < 0$

Cutoff is a trivial operating region, as all of the companion model parameters (depicted in Fig. 2) zero. No current flows into any of the nodes.

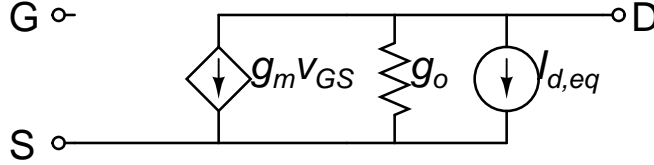


Figure 2: Linear companion model for n-channel MOSFET

2.2.2 Saturation: $0 \leq v_{GS} - V_T < v_{DS}$

In saturation, the following large-signal model holds:

$$I_D = \frac{K}{2}(v_{GS} - V_T)^2(1 + (v_{DS} - v_{DS,sat})/V_a) \quad (4)$$

where V_a is the Early voltage and $v_{DS,sat} = v_{GS} - V_T$. Since I_D is a function of two variables, we now use partial derivatives to find the small-signal conductances across the DS port.

$$g_m = \frac{\partial I_D}{\partial v_{GS}} = K(v_{GS} - V_T)(1 + (v_{DS} - v_{DS,sat})/V_a) = \sqrt{2KI_D} \quad (5)$$

$$g_o = \frac{\partial I_D}{\partial v_{DS}} = \frac{K}{2}(v_{GS} - V_T)^2/V_a = I_D/V_a \quad (6)$$

We seek to define a bias current from drain to source that will cause the node voltages to evolve by Newton-Raphson iteration, as we did for the diode in the previous subsection. We now use the multivariable form of the iteration equation:

$$I_{D,n+1} = I_D + g_m(v_{GS,n+1} - v_{GS,n}) + g_o(v_{DS,n+1} - v_{DS,n}) \quad (7)$$

Rearranging this equation, we find that $I_{D,n+1} = (I_D - g_m v_{GS,n} - g_o v_{DS,n}) + g_m v_{GS,n+1} + g_o v_{DS,n+1}$. Hence the bias current is $I_{D,eq} = I_D - g_m v_{GS,n} - g_o v_{DS,n}$. Although we will not derive it here, it should now be apparent that the following general result holds when determining bias currents for Newton-Raphson iteration:

$$I_{eq} = I(\mathcal{V}_i) - \sum_{v \in \mathcal{V}} \frac{\partial I}{\partial v} v_i \quad (8)$$

where \mathcal{V} is the set of device port voltages and \mathcal{V}_i is the set of calculated node voltages from the previous iteration.

2.2.3 Triode: $0 \leq v_{DS} \leq v_{GS} - V_T$

The drain current in this operating region is related to the v_{GS} and v_{DS} by Equation 9. Note that this model does not include the Early effect.

$$I_S = K((v_{GS} - V_T) - v_{DS}/2)v_{DS} \quad (9)$$

Applying partial derivatives, we find the small-signal conductances:

$$g_m = \frac{\partial I_D}{\partial v_{GS}} = K v_{DS} \quad (10)$$

$$g_o = \frac{\partial I_D}{\partial v_{DS}} = K((v_{GS} - V_T) - v_{DS}) \quad (11)$$

Again, $I_{D,eq} = I_D - g_m v_{GS} - g_o v_{DS}$.

2.2.4 p-channel Results

The linear companion model for a p-channel MOSFET is shown in Figure 3, and the values of g_m , g_o , and $I_{S,eq}$ for the various operating regions are summarized in Table 1. For all operating regions, $I_{S,eq}$ is related to I_S , g_m , and g_o by the following equation:

$$I_{S,eq} = I_S - g_m v_{SG} - g_o v_{SD} \quad (12)$$

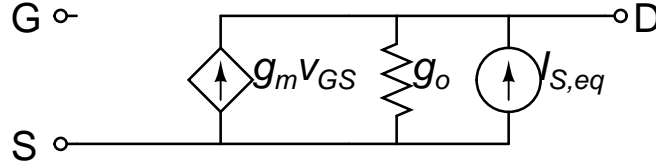


Figure 3: Linear companion model for p-channel MOSFET

Table 1: PMOS companion model parameters

Operating Region	Parameters
Cutoff: $v_{SG} - V_T < 0$	$I_S = 0$ $g_m = 0$ $g_o = 0$
Saturation: $0 \leq v_{GS} - V_T < v_{DS}$	$I_S = \frac{K}{2}(v_{SG} - V_T)^2(1 + (v_{SD} - v_{SD,sat})/V_a)$ $g_m = \sqrt{2KI_S}$ $g_o = I_S/V_a$
Triode: $0 \leq v_{DS} \leq v_{GS} - V_T$	$I_S = K((v_{SG} - V_T) - v_{SD}/2)v_{SD}$ $g_m = Kv_{SD}$ $g_o = K((v_{SG} - V_T) - v_{SD})$

2.3 Bipolar Junction Transistor

Companion models for BJTs are the most complex circuits presented here because current flows between all nodes. The model is not quite as unwieldy as it looks, however, because there is only one operating region.

We start with a simplified, mid-band Gummel-Poon NPN model, as shown in Figure 4. The base currents are defined as follows:

$$i_{bf} = (I_{fs}/\beta_f)(e^{v_{BE}/V_t} - 1) \quad (13)$$

$$i_{br} = (I_{rs}/\beta_r)(e^{v_{BC}/V_t} - 1) \quad (14)$$

Then, defining the small-signal conductances as in Figure 5, we have

$$g_{\pi,f} = \frac{\partial i_{bf}}{\partial v_{BE}} = (I_{fs}/\beta_f)e^{v_{BE}/V_t}/V_t \approx I_{bf}/V_t \quad (15)$$

$$g_{\pi,r} = \frac{\partial i_{br}}{\partial v_{BC}} = (I_{rs}/\beta_r)e^{v_{BC}/V_t}/V_t \approx I_{br}/V_t \quad (16)$$

$$g_{m,f} = \frac{\partial i_C}{\partial v_{BE}} = I_{fs}e^{v_{BE}/V_t}/V_t = \beta_f g_{\pi,f} \quad (17)$$

$$g_{m,r} = \frac{\partial i_C}{\partial v_{BC}} = I_{rs} e^{v_{BC}/V_t} / V_t = \beta_r g_{\pi,r} \quad (18)$$

We add the Early effect by placing an output resistance across the CE port, with conductance I_C/V_a .

Using the general bias current formula (Eqn. 8), we find

$$I_{bf,eq} = I_b f(v_{BE}) - g_{\pi,f} v_{BE} \quad (19)$$

$$I_{br,eq} = I_b r(v_{BC}) - g_{\pi,r} v_{BC} \quad (20)$$

$$I_{C,eq} = I_C(v_{BE}, v_{BC}) - g_{m,f} v_{BE} + g_{m,r} v_{BC} - g_o v_{CE} \quad (21)$$

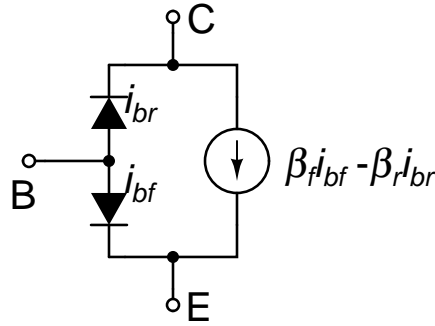


Figure 4: Mid-band Gummel-Poon BJT Model

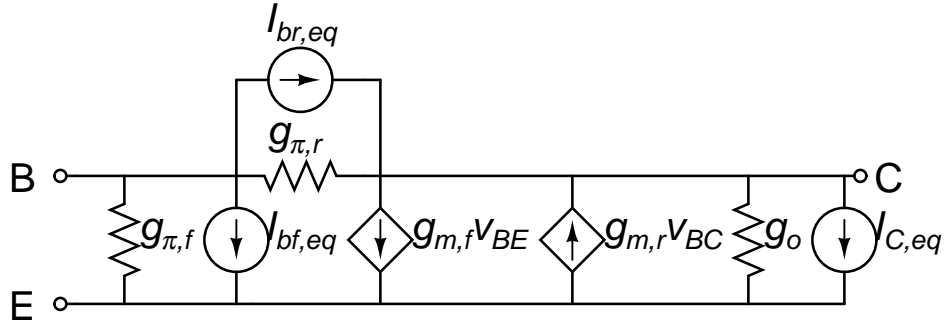


Figure 5: Linear companion model for NPN BJT

The analogous model for a PNP BJT is shown in Figure 6, and the corresponding parameters are listed below:

$$g_{\pi,f} = \frac{\partial i_{bf}}{\partial v_{EB}} = (I_{fs}/\beta_f) e^{v_{EB}/V_t} / V_t \approx I_{bf}/V_t \quad (22)$$

$$g_{\pi,r} = \frac{\partial i_{br}}{\partial v_{CB}} = (I_{rs}/\beta_r)e^{v_{CB}/V_t}/V_t \approx I_{br}/V_t \quad (23)$$

$$g_{m,f} = \frac{\partial i_E}{\partial v_{EB}} = I_{fs}e^{v_{EB}/V_t}/V_t = \beta_f g_{\pi,f} \quad (24)$$

$$g_{m,r} = \frac{\partial i_E}{\partial v_{CB}} = I_{rs}e^{v_{CB}/V_t}/V_t = \beta_r g_{\pi,r} \quad (25)$$

$$g_o = I_E/V_a \quad (26)$$

$$I_{bf,eq} = I_b f(v_{EB}) - g_{\pi,f} v_{EB} \quad (27)$$

$$I_{br,eq} = I_b r(v_{CB}) - g_{\pi,r} v_{CB} \quad (28)$$

$$I_{C,eq} = I_C(v_{EB}, v_{CB}) - g_{m,f} v_{EB} + g_{m,r} v_{CB} - g_o v_{EC} \quad (29)$$

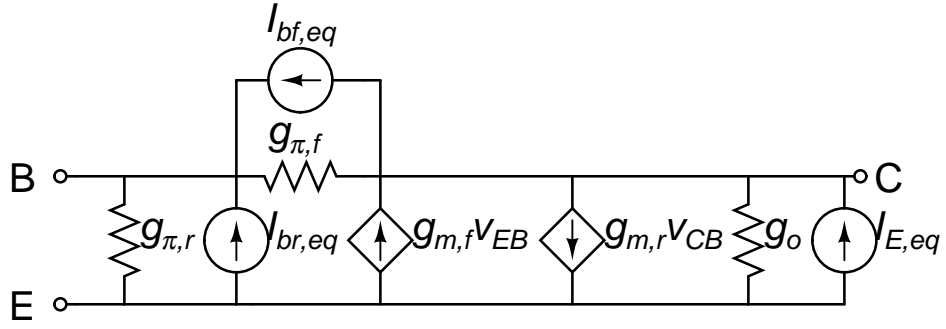


Figure 6: Linear companion model for PNP BJT

3 Energy storage components

As explained in introduction, transient simulation of energy storage require companion models to be formed. We present here the results for one-step integration methods : Backward Euler (BE), Forward Euler (FE) and Trapezoidal Rule (TR)

3.1 Capacitor

In deriving the model parameters, we start with the differential equation relating current to voltage in a capacitor:

$$I = C(dv/dt) \quad (30)$$

We then transform this differential equation to an integral equation

$$v(t + \Delta t) = v(t) + \int_t^{t+\Delta t} i(x)dx \quad (31)$$

We then use an integration method to get an expression for v_{n+1} , which can be reformulated as

$$v_{n+1} \approx v_n + (\Delta t/C)I \quad (32)$$

How we define I determines the integration method implemented by the companion model. Geometrically, it corresponds to the point at which we evaluate the derivative : FE computes the derivative at the current point, BE at the next point, and TR is an average of the two. Table 2 summarizes three common integration methods and the corresponding model parameters.

We recognise either a Thevenin model or a voltage source, depending on whether I depends on I_{n+1} (implicit method) or just I_n (explicit method). Note that every implicit model uses a Thevenin model, which can be converted to an equivalent Norton model. We chose Thevenin model for the capacitor since it is more accurate : computers have trouble dividing by small numbers such as Δt , and this is precisely what Norton models do. This can be explained by the fact that voltage is the state variable of capacitors : $v(t)$ is derivable, which means its change in Δt is proportional to Δt .

Table 2: Capacitor companion model parameters for integration methods

Method	Parameters
Forward Euler: $I = I_{n+1}$	$V_{eq} = v_n + (\Delta t/C)i_n$ $R = 0$
Backward Euler: $I = I_n$	$V_{eq} = v_n$ $R = \Delta t/C$
Trapezoidal: $I = (I_n + I_{n+1})/2$	$V_{eq} = v_n + (\Delta t/2C)i_n$ $R = \Delta t/2C$

3.2 Inductor

The inductor is the dual of the capacitor, so similar results apply. We end up with the formula

$$i_{n+1} \approx i_n + (\Delta t/L)V \quad (33)$$

Where V is determined by the integration method. The results are displayed in Table 3. Again, we have the choice between Thevenin and Norton model for implicit methods, and for the same reasons, we choose Norton model (current is the state variable of the inductor).



Figure 7: Linear companion model for capacitor

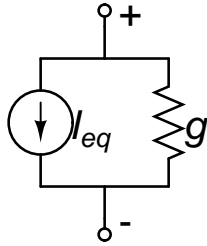


Figure 8: Linear companion model for inductor

References

- [1] S. Jahn, M. Margraf, V. Habchi, and R. Jacob. Qucs technical papers. <http://qucs.sourceforge.net/tech/>, 2003.
- [2] T. L. Pillage, R. A. Rohrer, and C. Visweswariah. *Electronic Circuit & System Simulation Methods*. McGraw-Hill, Inc., 1994.
- [3] J. Vlach and K. Singhal. *Computer Methods for Circuit Analysis and Design*. Van Nostrand Reinhold, 1983.

Table 3: Induction companion model parameters for integration methods

Method	Parameters
Forward Euler: $V = V_{n+1}$	$I_{eq} = i_n + (\Delta t/L)v_n$ $g = 0$
Backward Euler: $V = V_n$	$I_{eq} = i_n$ $R = \Delta t/L$
Trapezoidal: $V = (V_n + V_{n+1})/2$	$I_{eq} = i_n + (\Delta t/2L)v_n$ $R = \Delta t/2L$