Computer-Aided IC Design

An Open-Source Summary

Thomas Debelle

KU LEUVEN

October 3, 2025

Contents

1 Numerical simulation of analog circuits

1.1 Analog circuit simulation

> **Definition**
>
> A simulation is a numerical calculation of the response of a circuit to an input stimulus

It requires 3 basic inputs:

1. **Netlist**: how are the devices interconnected together
2. **Device models**: the actual device information used in the circuit. Usually provided by the foundry, related to a certain node process. Comes in a set of files named **PDK** (Process Development Kit)
3. **Controls**: the stimuli and environmental settings (temperature, …)

Using those 3 basic ingredients, we can simulate various things such as:

- DC sweep: find the operating point `.OP` and try various DC values
- Time domain: transient analysis to check the behavior in time
- Frequency domain: after doing transient analysis apply FFT to check frequency behavior.
- PSS: small-signal analysis around a DC point

A crucial point is to swipe various **operating conditions**. Devices are not perfect and their parameters will vary from one to another. We usually realize Monte-Carlo (**MC**) parameter variations. We can also check various temperatures.

1.1.1 Transient simulations

Usually, a circuit can be represented as a system of non linear differential equations.

$$V = RI \qquad V = L\frac{dI}{dt} \qquad I = C\frac{dV}{dt} \tag{1}$$

We know that the circuit should follow KCL, KVL and the device specific equation. On paper, this rather looks easy as we only need to ensure that the sum of current entering a node is null and the sum of voltage in a loop is also null.

But, 2 components makes the task harder. We don't have a direct $I(\alpha, \beta, ...)$ for them:

1. *Voltage source*: if assumed ideal, it can deliver any current. We must add an equation an variable to ensure that it can be appropriately solved.
2. *Inductors*: the fundamental equation 1 shows us that we know the voltage directly but not the current. So, we must transform it into an integral equation with $I = 1/L \int \Delta V \, dt$. We have this extra ΔV and we can use $\Delta V = Ldi/dt$

This brings the total amount of equations to $N - 1 + V_s + L$ with $N - 1$ being the amount of components, V_s the amount of voltage sources and L being the number of inductor.

We can safely assume that on average, this number can be simplified to $\approx N - 1$.

This technique is called the **Modified Nodal Analysis** MNA and is quite memory efficient. This is what was used in the first open-source program made at UC Berkeley, **SPICE**.

1.1.2 Numerical techniques

We can summarize the process of solving such system numerically as 3 nested loops:

- **time discretization (numerical integration) :** transforms system of nonlinear differential equations into a sequence of purely algebraic systems of nonlinear equations, to be solved at each discrete time point
  - **nonlinear equation solution (Newton-Raphson) :** iteratively solves the system of nonlinear equations at a given time point by solving a sequence of linearizations of the system
    - **linear equation solution (LU factorization) :** solves the system of linearized algebraic equations during each iteration at each discrete time point

1.1.2.1 1. Time discretization First thing is to discretize the derivative by replacing them with a finite difference approximation.

Table 1: Various integration techniques

Characteristics	Backward Euler (BE)	Trapezoidal	Gear-Shichman
Next value	$x_{t+h} = x_t + hx'_{t+h}$	$x_{t+h} = x_t + h/2(x'_{t+h} + x'_t)$	$x_{t+h} = 4/3x_t - 1/3x_{t-h} + 2h/3x'_{t+h}$
Next derivative	$x'_{t+h} = 1/h(x_{t+h} - x_t)$	$x'_{t+h} = 2/h(x_{t+h} - x_t) - x'_t$	$x'_{t+h} = -2/hx_t + 1/2hx_{t-h} - 2/3hx'_{t+h}$
LTE	/	$-h^3/12x'''_{t+h} + \mathcal{O}(h^4) = \mathcal{O}(h^3)$	$2h^3/9x'''_{t+h} + \mathcal{O}(h^4) = \mathcal{O}(h^3)$
A-stable	**V**	**V**	**V** if $h_n/h_{n-1} \leqslant 1$

If the Local Truncation Error (LTE) is too large, we can implement some simple feedback pattern that will adjust the timestep to obtain an error below a certain threshold.

Stability is also crucial and we use a classic *test equation* $x' = \lambda x$. We call a system A-stable if it is stable any physically stable circuit (pole laying ini the left-hand side of the plane).

For G-S if we must increase the timestep, we can switch to BE temporarily and then use G-S which is more flexible.

1.1.2.2 2. Non-linear equation solutions

We now solve the system of nonlinear equations using an iterative algorithm. It will linearize locally the equations using an initial guess that we refine.

$$F(x) = 0 \Rightarrow x_{(k+1)} = x_{(k)} - \frac{F(x_{(k)})}{F'(x_{(k)})} \qquad \text{Newton-Raphson algorithm}$$

It will converge in a certain neighborhood, if it doesn't we could also reduce the timestep and do again the first step then N-R. We can also have the multi-dimensional algorithm based on the jacobian:

$$J_F(x_{(k)}(t+h))(x_{(k+1)}(t+h) - x_{(k)}(t+h)) = -F(x_{(k)}(t+h)) \tag{2}$$

$$J_F(x_{(k)}(t+h)) = \frac{\partial q(x_{(k)}(t+h))}{\partial x} + \frac{h}{2}\frac{\partial f(x_{(k)}(t+h))}{\partial x} \tag{3}$$

The convergence is quadratic and we can base ourselves on previous results. But for the first iteration, we have no guess besides the `.OP` value. The circuit is at rest, caps are open circuit and inductance short circuit. This leaves a lot of node floating which can be tricky for `.OP` simulation. One solution is **conductance ramping** (GMIN). The idea is to put a conductance between the floating node and possible well-defined nets (GND, Vss, …). We then reduce slowly the conductance to reach the open circuit condition.

1.1.2.3 3. Linear equation solution

The system looks like $Ax = B$ and since it will be sparse we can use some LU-factorization algorithm in sparse Gaussian elimination $\mathcal{O}(n^3/3)$. Followed by forward L and backward U substitution.

If the magnitude of the pivot is too small, near-singularity of the coefficient matrix results in non-convergence.

1.2 RF simulation techniques

Sadly, we cannot use transient analysis for RF circuit. This would take too much time due to the small timestep needed until it reaches the steady-state. But, we often are just interested in the steady-state of the circuit. If we use classic `.TRAN` analysis and we want to check the frequency response, we must use the proper FFT.

$$N_{FFT} = 2^n \geqslant \frac{2f_{max}}{\Delta f} \qquad h = \frac{n_{periods}}{N_{FFT} \cdot F_{osc}}$$

So to perform relevant analysis we must know in advance some characteristics on the circuit. Which is not always possible if they get more and more complex.

So instead of simulating the transient behavior, let's focus on the steady-state behavior.

1.2.1 Direct steady-state solution

> **Definition**
>
> solution that is asymptotically approached when the effect of the initial condition dies out.
>
> may not exist or there may be several; can be periodic - quasi-periodic

Table 2: Comparing the two main methods in RF simulation

Method	Type	Description	Use case
Shooting methods	Temporal	From the initial condition, tries to find T where $x(T) = X(0)$	good for non-linear circuits, uses N-R
Harmonic balance	Frequency	Based on the fact that any signal can be seen as a **truncated Fourier series** like $x(t) = \sum_{k=-K}^{k=K} X[k]e^{j2\pi f_k t}$ So for each node, we will have $2K + 1$ coefficients to balance.	Good for (quasi-)periodic and linear circuits. There are pre-conditioner for such sparse matrix problem.

1.2.2 Periodic small-signal analysis (PAC)

We **linearize** around the periodic operating point and compute the PAC response. It will have different tone for one excitation. It is a **time varying linear** analysis. This doesn't reflect harmonic or non-linear effects.

1.2.2.1 PNOISE From this type of analysis, we can do some noise analysis where the noise sources get modulated by the operating point. The transfer function (TF) is also time-varying.

This is the reason we have pink, $1/f$ noise.

1.2.3 Envelope calculation

Sometimes, we are

1.2.4 Recap

Table 3: Recap table of RF analysis techniques

Method	Use case
TRAN	to see the startup behavior and exact signal, slow.
Steady-State	Steady-state analysis with shooting or harmonic balance methods according to the circuit
PAC	Analysis of multiple time varying operating point

1.3 Multi-level analog modeling

1.4 Analog behavioral description languages

2 License

2.1 Modification

To contribute to this work or any other one from this project please find more information at the Github repository.