“Symbolic modeling of analog circuits”

By

Elyoenai Martinez Romero

Thesis submitted in partial fulfillment of the degree of

Master of Science

Department of Electronics

National Institute of Astrophysics, Optics and Electronics

(INAOE)

Advisor:

Dr. Esteban Tlelo Cuautle

Tonantzintla, Puebla, México

Acknowledgments

I want to express my total gratitude to my advisor Prof. Dr. Esteban Tlelo Cuautle for all the support and suggestions given during this time of Thesis development.

I am also grateful with my family and friends for all the support and encouragement given along this stage.

I would like to thank the National Institute for Astrophysics, Optics and Electronics (INAOE) for giving me the opportunity to have studied into one of the best institutions of this country.

I wish to express my gratitude to Consejo Nacional de Ciencia y Tecnología (CONACYT) for the financial support given for my master studies.

My gratitude is also for the members of my defense committee for their time and suggestions. Thanks to Prof. Dr. Luis Hernández Martínez, Prof. Dr. Juan Manuel Ramírez Cortés and Prof. Dr. Georgina Flores Becerra.

Finally, I want to thank all the professors at INAOE for the knowledge I was given during this time at this institute.

Index

Abstract

Symbolic analysis has been for some years a topic of interest in electronics, because it represents a way to have a better understanding on circuit behavior. Through this time, different approaches have been developed to obtain expressions that allow representing the circuit behavior and then try to improve its performance.

These approaches have been done for example, through graphs methods, flow charts or analytical methods, like MNA or pure nodal analysis, which is the chosen method to be used in this Thesis.

Nodal analysis utilized in this Thesis could not be employed without an important element in circuit analysis; it is the nullor, which can be employed to model different devices. But it is also useful for reducing the rank of matrices of the system being analyzed, by using the properties this element have.

In this Thesis, a method for obtaining symbolic expressions is explained, and it is shown how the nullor helps creating the nodal formulation which allows getting those expressions.

There is an open research in symbolic analysis due to the length of symbolic expressions and the necessity to reduce them in order they could be easier to be understood, this way, some simplification methods have been developed and presented in this Thesis, to show their usefulness to obtain shorter expressions.

Within these reduction methods, a method of reducing order is treated, it is known as asymptotic waveform evaluation, and allows the reduction of order in expressions, to have smaller expressions.

Symbolic analysis is also useful for calculating expressions that represent parameters like noise in amplifiers; this topic is also reviewed in this Thesis.

Chapter 1. Symbolic analysis

1.1 Symbolic analysis methods

Usually, in the process of designing a circuit using metal-oxide-semiconductor field effect transistors (MOSFETs), after the first approach on sizing elements, a numerical simulation is held; in order the circuit behavior can be verified and afterwards improved, to achieve the desired specifications. However, when trying to get a better insight on what is happening on the circuit, a symbolic expression representing it could be more useful to try to understand its behavior. There are different approaches to obtain the referred symbolic expression; this chapter is aimed to summarize some of these methods.

1.1.1 Tree enumeration methods

Tree enumeration methods are based on graph theory, they have been the base for old and more recent symbolic analysis programs. There are two categories to classify them: the directed and the undirected tree enumeration. These methods face the disadvantage that they could only deal with small RLCgm circuits because a large number of symbolic terms are generated, producing expressions difficult to handle [1].

1.1.1.1 Directed tree enumeration

To use this method, a slightly modified circuit, should be implemented, adding an admittance in parallel with a current source to the original circuit, this modification contributes to the construction of an adequate determinant expression and the necessary cofactors for the augmented circuit. To see graphically how the augmented circuit can be created, let's take for instance the amplifier of figure 1.1.

The small signal circuit representation of this circuit, where the MOSFET is modeled by a voltage-controlled current source (VCCS), can be observed in figure 1.2a. After adding the mentioned controlled source and the admittance, the augmented circuit is obtained, which can be seen in figure 1.2b.

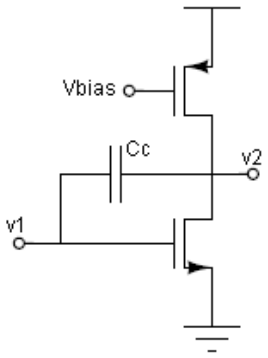
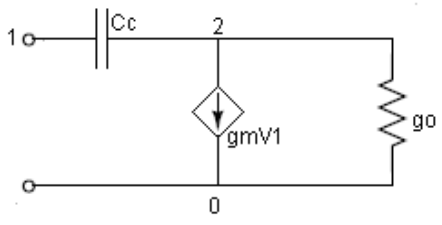
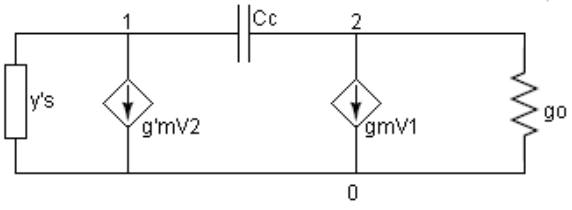


Figure 1.1 Example of circuit for applying tree enumeration



(a)



(b)

Figure 1.2 (a) Small signal representation of amplifier of figure 1.1 (b) Augmented circuit.

Once the augmented circuit is ready, a directed graph representing it can be formed by using the stamps corresponding to each of the elements of the circuit. In this case, after employing the adequate stamps, the graph that represents the circuit of the example can be seen in figure 1.3. In this graph, all directed trees should be enumerated, because the admittance products of these trees are used to find the nodal admittance matrix determinant and cofactors to produce the required symbolic transfer functions. However, the number of branches in the graph may grow exponentially with the increase of the circuit-elements.

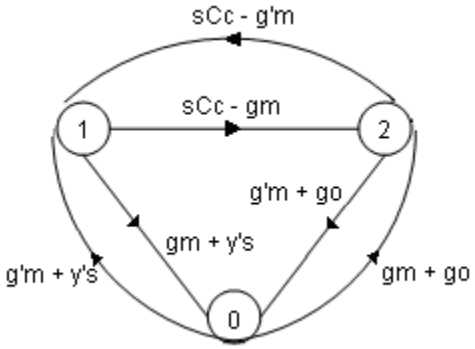
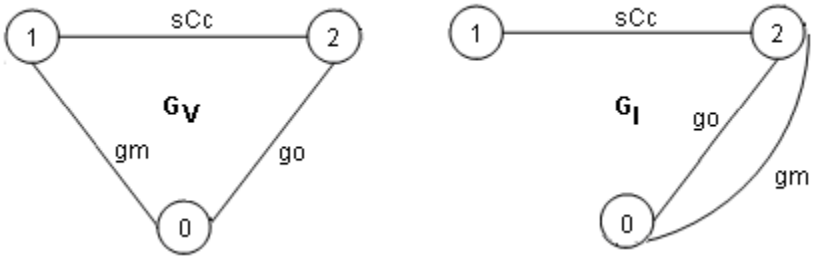


Figure 1.3 Directed graph

1.1.1.2 Undirected tree enumeration

This method is based on the construction of two graphs, one for currents and the other for voltages, for this reason, this method is also known as “two-graph tree enumeration method”. The constructed graphs are equal for RLC circuits, but they are slightly different for RLCgm circuits, because the gm admittance corresponding to the controlled sources is positioned in a different place, depending on the current or voltage graph. The construction of these graphs is useful for generating the cofactors of nodal admittance matrix Y_n that generates the symbolic transfer functions.

As an example for graph construction, let’s use the circuit of previous example, in specific, the small signal representation of it, that is, the one of figure 1.2a. Using this circuit, two graphs can be constructed; one for voltages and one for currents, this is illustrated in figure 1.4.



(a) (b)

Figure 1.4 (a) Voltage graph (b) Current graph

After applying the adequate operations with these graphs, the needed transfer function can be obtained. However, as for the directed tree enumeration, the number of branches may grow exponentially with the increase in the number of circuit-elements.

1.1.2 Signal flowgraph (Topological methods)

There are two kinds of flowgraphs aimed to generate symbolic equations of circuits, the Mason signal flowgraph and the Coates graph, the first one is the popular known method used in other applications, such as control, but in this case, it is also useful for symbolic analysis. The second method was created ad-hoc for generating symbolic expressions. These flowgraph methods have an advantage over tree enumeration methods, because they can deal with circuits containing all types of controlled sources. However, these methods have also the limitation of circuit size, because they generate very large expressions that cannot be managed easily due to their complexity.

Mason flowgraph method for finding transfer functions among two nodes of a circuit (x_j/x_i) is based in the application of Mason's formula:

$$\frac{x_j}{x_i} = \frac{1}{\Delta}\sum_k P_k \Delta_k \qquad (1.1)$$

Δ = 1-(sum of all Li's), - where Li's are the loops of the flowchart
+ (sum of all second-order loop weights)
- (sum of all third-order loop weights)
+ …

For applying this method a tree and cotree should be constructed from the original circuit; in this case, voltage sources should be located in tree, and current sources in the cotree. Then KCL, branch admittances, and tree branches voltages are used to find an expression for every cotree link current. KVL, branch impedances, and cotree link currents are used to find an expression for every tree branch voltage. The signal flowchart should be created by drawing a node for each current and voltage source, tree branch voltage, and cotree link current. The branches between nodes of the graph represent the equations corresponding to the analyzed circuit. Once the flowchart is constructed, Mason formula can be applied to find the transfer function between the chosen nodes.

A figure that can be used to illustrate the formation of a signal flowgraph can be seen next (figure 1.5)

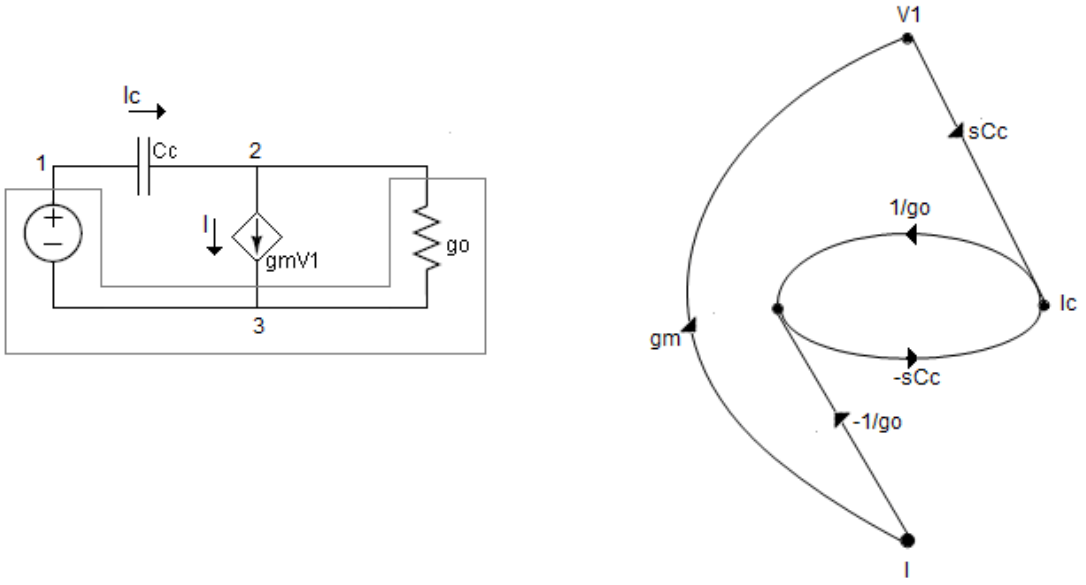


Figure 1.5 Illustration on graph creation for Mason method

1.1.3 Parameter extraction method

This is a method best suited when dealing with a semi-symbolic expression. Because of that, it can handle larger circuits. However, if the number of the symbolic terms is large,

it has the same problem of exponential growing of terms, making difficult to manage the resulting expressions.

To apply this method for symbolic analysis, an appropriate pattern should be found in the matrix being dealt with. For example, if indefinite admittance matrix is being used, the analysis and rules depend on the appearance of symbolic parameters in four locations in the matrix: (i,i),(i,j),(j,i) and (j,j). Other methods are based on a different pattern found in the matrix.

To use a parameter extraction method, a symbolic variable α, which appears in the matrix to solve, needs to be "extracted" using matrix operations, like adding rows or columns, or subtracting them, making possible to take the variable out of the matrix.

1.1.4 Interpolation method

This method is based on finding coefficients of the polynomial resulting from a determinant, to do this; different values of *s* are substituted to evaluate the function, being a better approach to use complex values for *s*, not only real ones.

When substituting values for *s*, a set of linear equations is formed. This way, to find the coefficients, the set of linear equations should be solved. The main disadvantage of this method relies on that it only generates rational expressions with the unique symbol described by *s*.

1.1.5 Nodal analysis and MNA

These methods lie on the idea of obtaining the fully symbolic equations of the circuit directly from its description, and then putting them into a linear matrix form: A*x*=b.
Where A is a symbolic matrix of dimension *n* x *n*, *x* is a vector of circuit variables of length n, b is a symbolic vector of constants. The analysis consists in solving for *x* of the system of equations.

To generate the already mentioned matrix, there are different techniques, such as nodal analysis and its modifications. This is the method adopted in this Thesis, and it is highlighted in the following chapters.

1.1.5.1 Nodal Analysis

This is one of the most known and popular methods for circuit analysis, and it can also be employed as a method for symbolic analysis. It is based on Kirchhoff's laws of current and voltage.

With nodal analysis (NA), only conductances and current sources can be dealt with; however there are modifications that allow managing other circuit elements.

Nodal Analysis is formulated as:

$$Y_n V_n = I_n \tag{1.2}$$

Y_n is known as Indefinite Admittance Matrix or Nodal Admittance Matrix, and has a (n x n) order and I_n is the independent vector or stimulus vector (n order), containing the independent current sources present in the circuit. V_n is the vector of variables.

Admittance matrix is the base for formulation methods; it can be set up through Kirchhoff's current law in nodes of the treated circuit, that is: [2]

$$\sum_{k=1}^{b} i_k = 0 \tag{1.3}$$

With b, the number of branches connected to that node and i_k the current in branch k. If this approach is applied to every node of the circuit, a system of equations can be constructed, which contains Y_n matrix.

There are two types of admittance matrix, definite and indefinite. An indefinite matrix become definite when node k is grounded (taken as the reference node) thus deleting the corresponding row and column from the matrix.

1.1.5.2 Modified Nodal Analysis

This method overcomes the problem of NA, because it can handle more circuit elements which allow performing a more complete analysis. Formulating the Y matrix of the circuit (nodal admittance matrix, as it was mentioned before), is the beginning of this method. However, when the circuit elements are modeled by using nullors, then the circuit analysis can be performed by just applying the NA method, instead of the MNA, as it is shown in the following chapters. It is worthy to mention that the NA method applied to nullor equivalent circuits does not increase its order compared to the MNA method, which is described below.

1.1.5.2.1 Stamps approach

One way to construct nodal admittance matrix is using element stamps, which allows applying automated methods. The manner this technique is applied, consist on analyzing every branch of the circuit and adding to nodal admittance matrix the contribution given for the elements present in the circuit. In the next figures, taken from [1], the stamps of some elements are shown; these are the conductance, current source and VCVS.

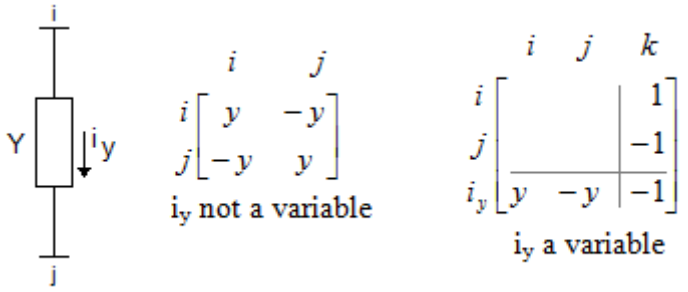


Figure 1.6 Stamp corresponding to conductance branch

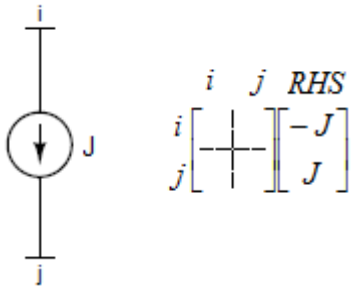


Figure 1.7 Stamp corresponding to current source

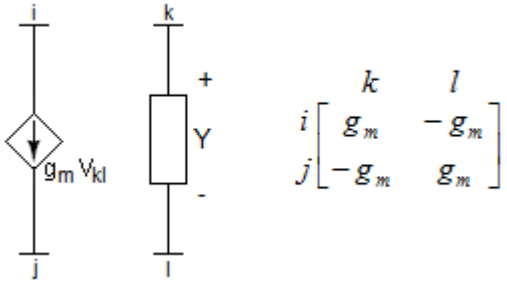


Figure 1.8 Stamp corresponding to VCCS

MNA method allows including those elements that cannot be handled with simple nodal analysis, such VCVS, CCCS, CCVS, this is done by introducing some branch currents as extra variables into the system of equations. Each new variable introduced would require an extra equation to solve for it; these extra equations are obtained from the branch relationship given by the added branch currents, corresponding to the element that wants to be taken in consideration.

The matrix resulting after the addition of new variables is known as MNA matrix, and has the form:

$$\begin{bmatrix} Y_n & B \\ C & D \end{bmatrix} \begin{bmatrix} V \\ I \end{bmatrix} = \begin{bmatrix} J \\ E \end{bmatrix} \tag{1.4}$$

Where I is a vector of size n_i, and contains the extra branch current variables introduced. E has the independent voltage sources values. C and D have the branch relationship equations whose currents are in vector I.

1.1.5.3 Compacted nodal analysis using nullors

1.1.5.3.1 Nullor concept

In 1954, Tellegen showed that an ideal amplifier could be used as a general block for implementing linear or non-linear circuits. In 1964, Carlin proposed the Nullor for modeling the ideal amplifier as a two port element, with four associated variables (figure 1.9). This element is composed by a Nullator in the input and a Norator in the output. [3], [4].

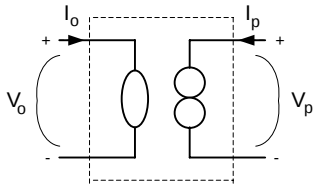


Figure 1.9 Nullor

Nullator has the property that v_o voltage and i_o current are always zero. Conversely, norator has the property that their voltage v_p and current i_p are arbitrarily assigned. From these properties, an equation can be derived, which represents Nullor behavior and is formed through the next null transfer matrix.

$$\begin{bmatrix} v_o \\ i_o \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} v_p \\ i_p \end{bmatrix} \tag{1.5}$$

Nullor allows modeling active elements such as opamps, otas, cfoas or cc's [5].

1.1.5.3.2 Compacted nodal analysis

The way to obtain a compacted system of equations (CSE) of an analog circuit is through obtaining an equivalent circuit with nullors substituting each active device and non NA compatible element with its nullor model. Then applying nullator and norator properties, that is: [3]

1. If a nullator is grounded, as it can be seen in figure 1.10(a), applying its voltage property, *i* node will be virtually grounded.

2. If a norator is grounded, the case is the same, because when applying its current property, *i* node will be virtually connected to ground.

3. For a floating nullator, as that shown in figure 1.10b, *i* node will be virtually connected to *j* node.

4. For a floating norator, like the one showed in figure 1.10d, *i* and *j* node will be virtually connected.

After the application of these considerations, a CSE is obtained, having an order of m = n – N, where n is the number of nodes and N the number of nullors. More on this topic will be presented in another chapter.

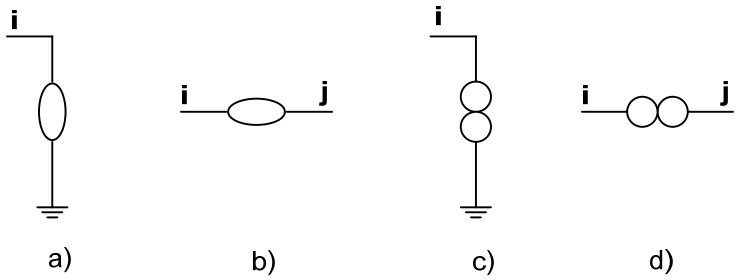


Figure 1.10 Nullator and norator properties

1.1.5.3.3 Transistor modeling using nullors

To take advantage on computer aided analysis, physical devices should be modeled in order simulations or symbolic analysis can be performed automatically. Let's take, for instance small signal transistor model, which allows considering this element as linear, in this case there are different models that can be used to represent transistors according to the dealt application, for example, there exist a basic and a high frequency MOS transistor model [2], basic model, using the VCCS, is shown in figure 1.11.

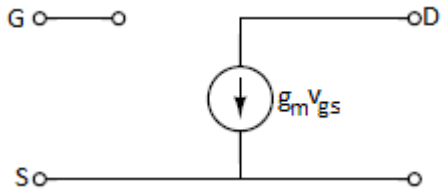


Figure 1.11 MOS Small signal model

To start modeling a transistor through nullors, the basic small signal model of figure 1.11 can be used, that is because its representation using a nullor is like it can be viewed in figure 1.12.

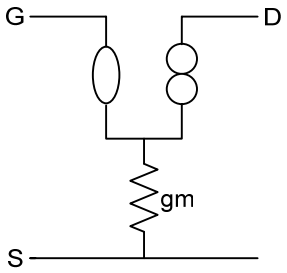


Figure 1.12 Transistor modeled through nullor element

Parting from figure 1.12, a more complex model can be developed, that is, parasitic elements can be added in order a more complex can be used, however, a model including more elements results in greater symbolic expressions, which creates a problem when trying to interpret the result. A solution to this problem is presented in the following chapters through the generation of symbolic behavioral models of ~~low~~ voltage amplifiers.

Using nullors to represent transistors gives the advantage in nodal formulation of reducing the rank, that can be showed from the fact that the two columns corresponding to the input nodes of a nullor can be added since the two input node voltages are equal, and the two rows corresponding to the output nodes are added to eliminate the output current [6], because of this, each nullor reduces the rank of the matrix by one.

1.2 Non dominant elements approach

Due to the exponential growth of the number of terms with the circuit size, the symbolic expressions for analog integrated circuits rapidly become too lengthy and complicated to use or interpret, rendering them virtually useless. A solution that can help solving this problem is considering the magnitude of elements, which varies in semiconductor devices. This way, in a transistor its transconductance is usually larger than its output conductance, taking this into account, leads to prone the majority of the terms in a fully symbolic expression, because just some of the terms are necessary to represent the circuit behavior. [7] This topic will be later explained in another chapter.

1.3 Moments and moment matching method

When analyzing an integrated analog circuit, the order of the system of equations might be much higher than it is required to understand the global behavior. In this manner, to reduce the order of the equations, the asymptotic waveform evaluation (AWE) method can be applied. Basically, one needs to compute moments and then match the moments by applying Padé approximation.

1.3.1 Concept of moments

In the *s* domain, the transfer function of a linear network *H*(*s*) is defined as the ratio of the output to the input under zero initial conditions [8]:

$$H(s) = \frac{Y(s)}{X(s)} \tag{1.8}$$

If the input is the impulse function δ(t), its Laplace transformation is 1. So the transfer function is also the impulse response at the port. If H(s) is expanded around s=0 by the Taylor series expansion, we have:

$$H(s) = \sum_{k=0}^{\infty} m_k s^k \tag{1.9}$$

where

$$m_k = \frac{1}{k!} \times \left. \frac{d^k H(s)}{ds^k} \right|_{s=0} \tag{1.10}$$

Where the *k*th coefficient of *H(s)*, m_k, is called the *k*th moment.

Assuming that h(t) is the corresponding time-domain impulse response, we have

$$H(s) = \int_0^\infty e^{-st} h(t) dt \tag{1.11}$$

The moments defined in (1.11) in terms of H(t) by using the Taylor expansion of e^{-st} in the Laplace transform H(s) can be written as:

$$\begin{aligned} H(s) &= \int_0^\infty e^{-st} h(t) dt \\ &= \int_0^\infty h(t) \left(1 - st + s^2 \frac{t^2}{2} + \cdots + s^k \frac{(-1)^k}{k!} t^k + \cdots \right) dt \\ &= \sum_{k=0}^{\infty} s^k \frac{(-1)^k}{k!} \int_0^\infty t^k h(t) dt \end{aligned} \tag{1.12}$$

Comparing (1.12) with the definition of (1.9), moments can be rewritten as:

$$m_k = \frac{(-1)^k}{k!} \int_0^\infty t^k h(t) dt \tag{1.13}$$

1.3.2 Padé Approximation

Padé approximation is a method that generates a family of rational functions whose moments agree with those of the impulse response. The rational functions are further decomposed into partial fractions, whose inverse Laplace transforms are used to constitute the approximated response of waveforms.

Padé approximation can be explained as: given two integers p and q, (p,q) Padé approximation of the transfer function H(s) is a rational function.

$$H_{p,q}(s) = \frac{P(s)}{Q(s)} = \frac{a_0 + a_1 s + a_2 s^2 + \cdots + a_p s^p}{1 + b_1 s + b_2 s^2 + \cdots + b_q s^q} \tag{1.14}$$

The Maclaurin expansion of $H_{p,q}$(s) agrees with that of *H*(s) in the first $p+q+1$ terms, i.e.,

$$H(s) = H_{p,q}(s) + O(s^{p'+q+1}) \tag{1.15}$$

As there are p+q+1 unknowns in (1.14), it is necessary to establish p+q+1 independent equations to solve for them. Assuming that $H_{p,q}(s)$ is a proper transfer function i.e. p<q, we can get coefficients in denominator Q(s) of (1.14) by solving the following equations:

$$\begin{bmatrix} m_0 & m_1 & \cdots & m_{q-1} \\ m_1 & m_2 & \cdots & m_q \\ \vdots & \vdots & \ddots & \vdots \\ m_{q-1} & m_q & \cdots & m_{2q-2} \end{bmatrix} \begin{bmatrix} b_q \\ b_{q-1} \\ \vdots \\ b_1 \end{bmatrix} = -\begin{bmatrix} m_q \\ m_{q+1} \\ \vdots \\ m_{2q-1} \end{bmatrix} \tag{1.16}$$

And the coefficients a_k of numerator P(s) satisfy the equation:

$$\begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{q-1} \end{bmatrix} = \begin{bmatrix} m_0 & 0 & 0 & \cdots & 0 \\ m_1 & m_0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ m_{q-1} & m_{q-2} & m_{q-3} & \cdots & m_0 \end{bmatrix} \begin{bmatrix} 1 \\ b_1 \\ \vdots \\ b_{q-1} \end{bmatrix} \tag{1.17}$$

Equations (1.16) and (1.17) can be verified by honoring the fact that the first p+q+1 moments of $H_{p,q}(s)$ match those of $H(s)$, i.e.,

$$\frac{P(s)}{Q(s)} = \frac{a_0 + a_1 s + a_2 s^2 + \cdots + a_p s^p}{1 + b_1 s + b_2 s^2 + \cdots b_q s^q} = m_0 + m_1 s + \cdots + m_{p+q} s^{p+q} + r(s) s^{p+q+1} \tag{1.18}$$

where r(s) is a polynomial function of s. Multiplying both sides with denominator Q(s), we have

$$a_0 + a_1 s + a_2 s^2 + \cdots + a_p s^p = \left(1 + b_1 s + b_2 s^2 + \cdots + b_q s^q\right) \times \left(m_0 + m_1 s + \cdots + m_{p+q} s^{p+q} + r(s) s^{p+q+1}\right) \tag{1.19}$$

By equating the coefficients of powers of *s* on both sides, we are able to write the two equations in (1.16) and (1.17).

1.4 Conclusion

As it was seen through this chapter, many different methods have been developed to obtain symbolic expressions, which vary from graphs to numerical methods.

Among those methods described within this chapter, compacted nodal analysis was chosen to be used in this Thesis, because the inclusion of nullor element generates a reduced system of equations. Nullor properties allow also eliminating non dominant circuit elements, which is important to reduce the resulting symbolic expressions.

Chapter 2. Modeling using nullors

2.1 Modeling using nullors

Nullor element and its singular characteristics were introduced in the first chapter, there, it was shown that it can be used to represent a MOS transistor, but this element can be employed to model other circuit elements.

Different elements can be represented using nullors, taking into account the voltage and current relationships they have. At first, the controlled sources using nullors will be presented in table 2.1, figures were taken from [9].

2.2 MOS transistor modeling

In the first chapter, a simple model of MOS transistor was presented, as it can be seen from table 2.1, a VCCS is represented with two nullator-norator pairs. The way to obtain the model presented in figure 1.11 is considering that the negative terminals of a VCCS are connected to the same node, as it can be seen in figure 2.1.

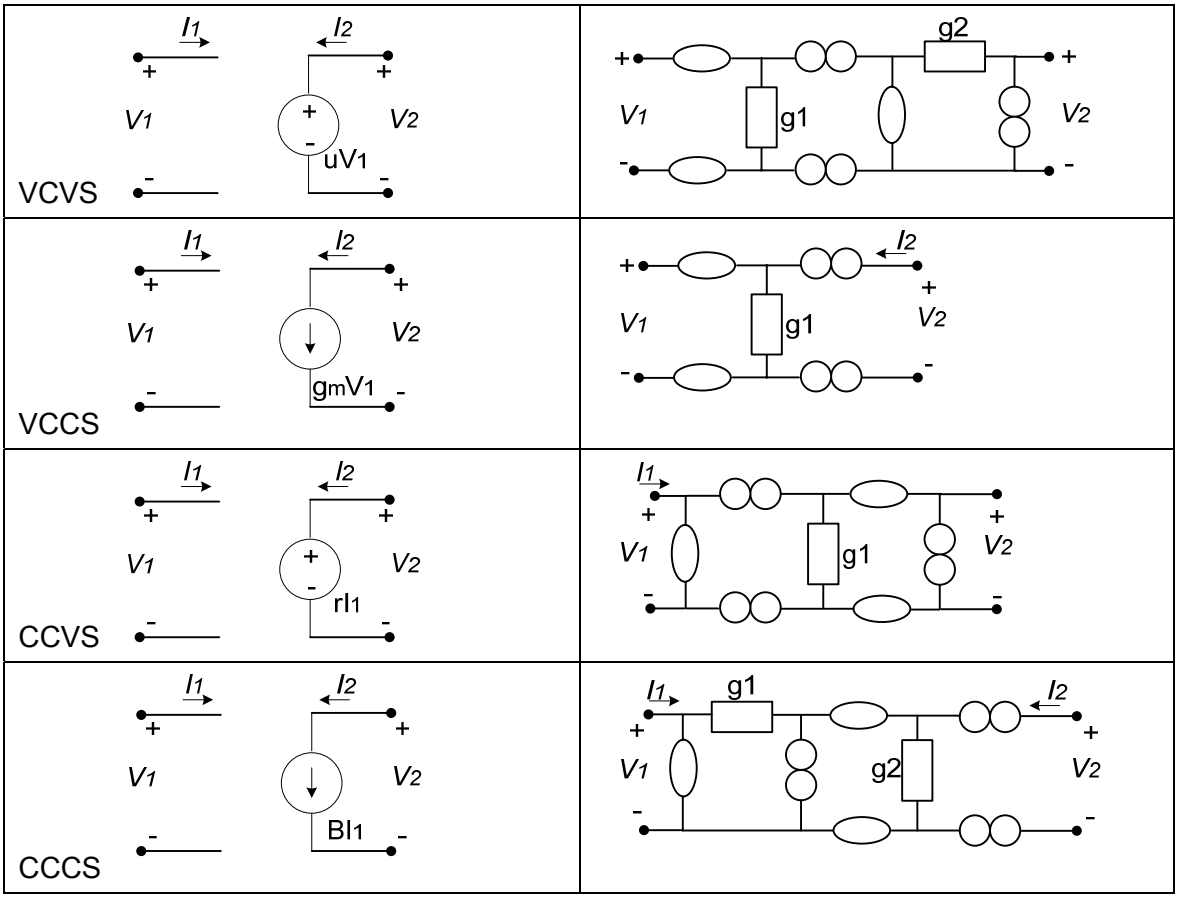


Table 2.1 Representation of controlled sources

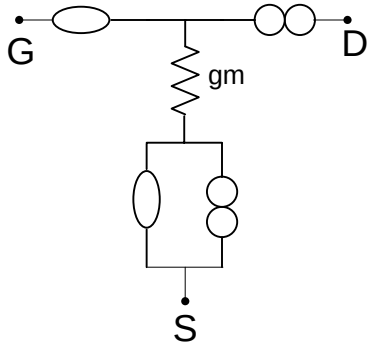


Figure 2.1 Simple model of MOSFET

Then, making use of nullors' properties, when a nullator and a norator are parallel connected, they can be considered as a short circuit, this way, simple model of MOS transistor is obtained.

When more complex models of transistors are required, like the small signal model presented in figure 2.2, as it appears on [10], elements should be added to obtain a complete equivalent.

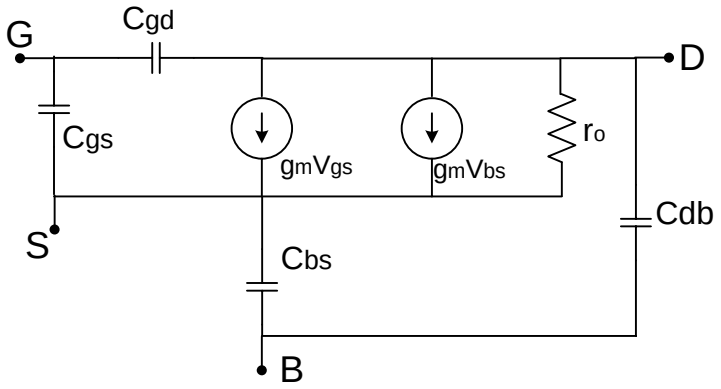


Figure 2.2 MOS transistor model using VCCS´s

Let's consider, for example, the case when adding parasitic capacitors and output conductance of MOS transistor, in this case, a resistance and two capacitors are added to the model, obtaining representation of figure 2.3.

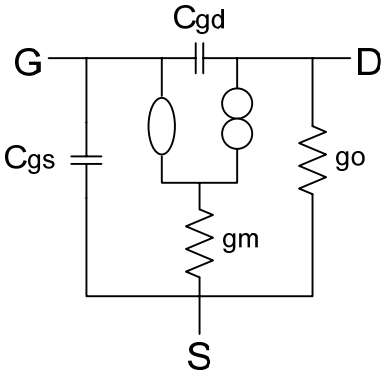


Figure 2.3 MOSFET model considering parasitic elements

If the resistance given for contacts is added to the model of figure 2.3, then the equivalent observed in figure 2.4 is obtained

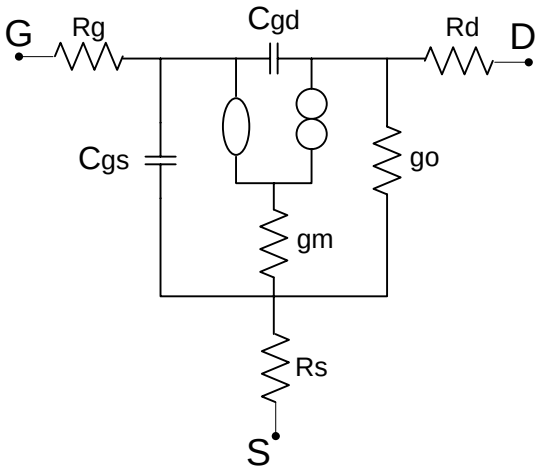


Figure 2.4 MOS transistor model considering contacts resistances

If substrate terminal wants to be taken into account, and the elements associated to this terminal are needed to be added to the model, then representation of figure 2.5 is obtained.

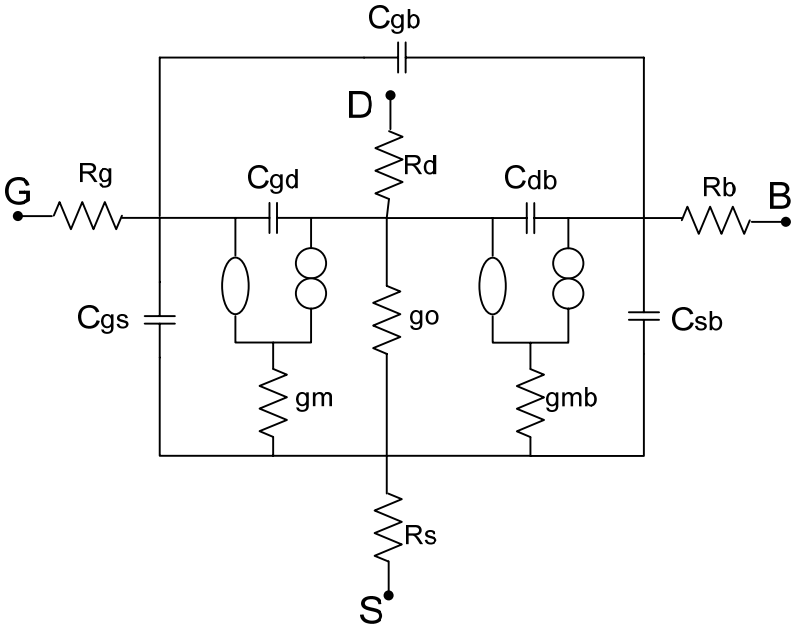


Figure 2.5 MOS transistor model considering substrate terminal and contacts resistances

When dealing with symbolic analysis, having short expressions is preferred, because these can give easier information to be interpreted, than bigger ones. So a simple representation of MOS transistor could be more useful to have a better understanding of circuit behavior. But it depends on the application, the necessary model to be employed.

2.3 Modeling other devices using nullors

2.3.1 Operational amplifier

It is known that input impedance of an operational amplifier is very large, ideally infinite, and output impedance, is ideally zero, for this reason, a nullor can be efficiently used to represent this device [3], as it can be seen in figure 2.6.

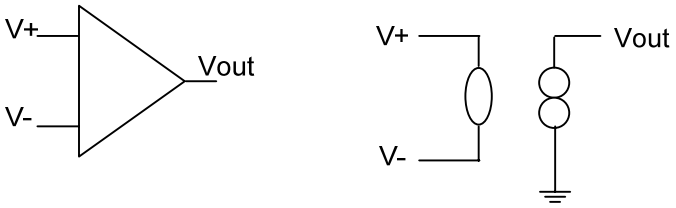


Figure 2.6 Equivalent of opamp using nullor

2.3.2 Operational transconductance amplifier (OTA)

This device, represented with its nullor equivalents, can be seen in figure 2.7, from this figure, the voltage across the conductance gm is just the differential voltage at the input port because the voltage across each nullator is zero.

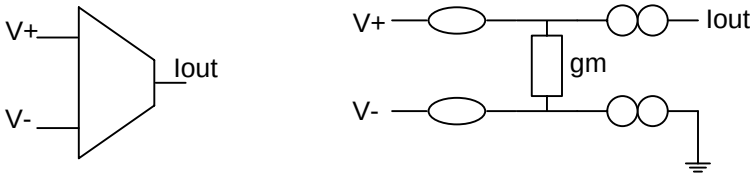


Figure 2.7 Equivalent of OTA using nullor

2.3.3 Current feedback operational amplifier (CFOA)

The nullor equivalent for this device is presented in figures 2.8. Open loop CFOA should have these characteristics [3]:
Infinite impedance at non-inverting input.
Zero impedance at inverting input.

Zero impedance at output.

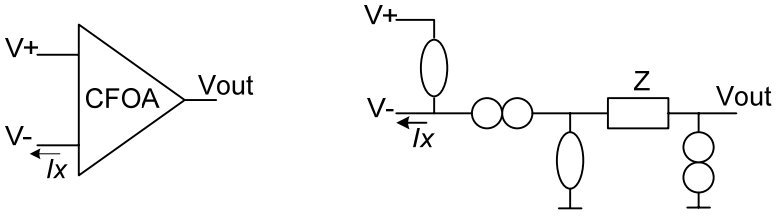


Figure 2.8 Equivalent of CFOA using nullor

2.3.4 Current conveyors

The current conveyor (CC) is a universal active device whose derivations are known as first generation CC (CCI), second generation CC (CCII), and third generation CC (CCIII). In [5] several of these current conveyors using nullors are presented and here repeated to show these structures. Their parasitic resistance at the terminal X (Rx) is included, so symbolic NA can be performed. In next figures these nullor equivalents of current conveyors are presented.

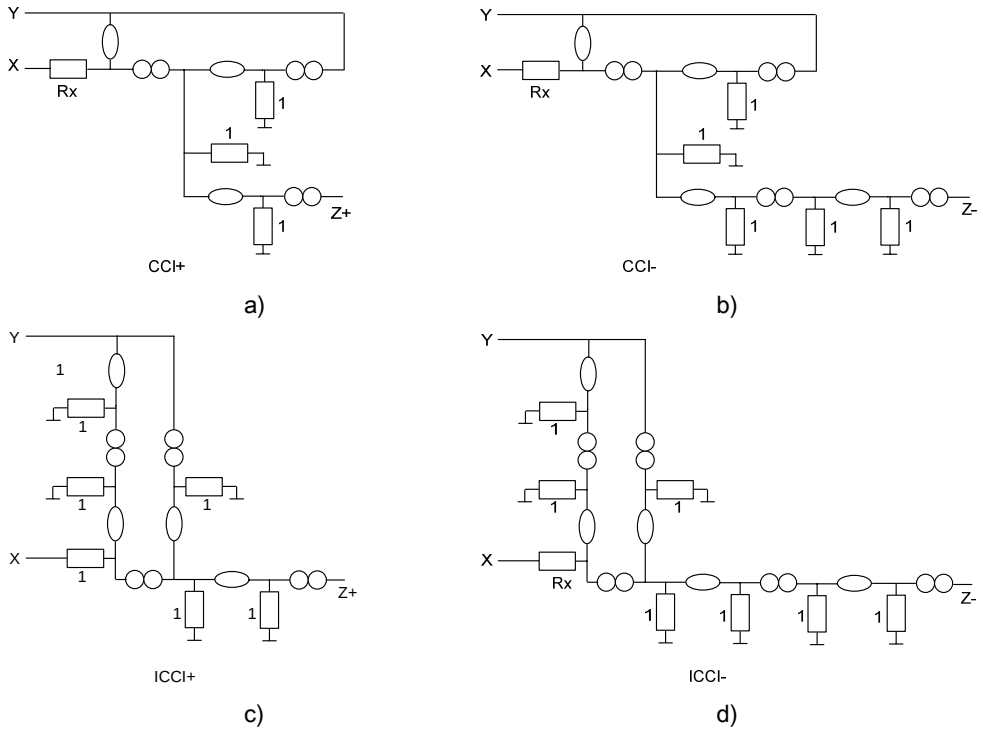


Figure 2.9 Equivalents of current conveyors using nullors a)CCI+ b)CCI- c)ICCI+ d)ICCI-

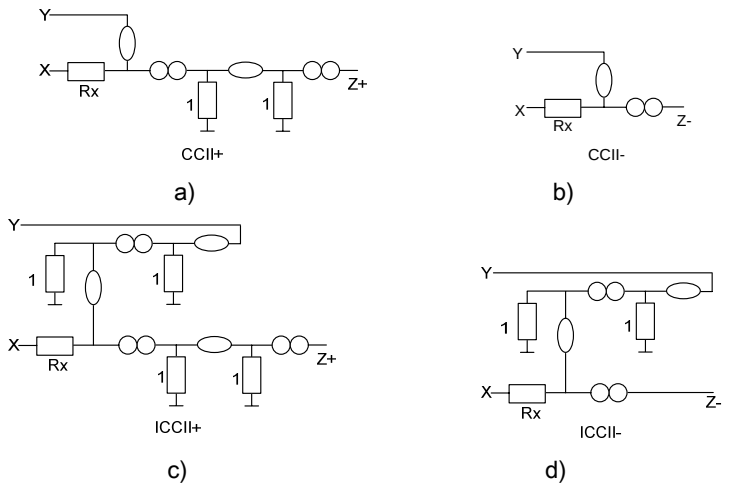


Figure 2.10 Equivalents of current conveyors using nullors a)CCII+ b)CCII- c)ICCII+ d)ICCII-

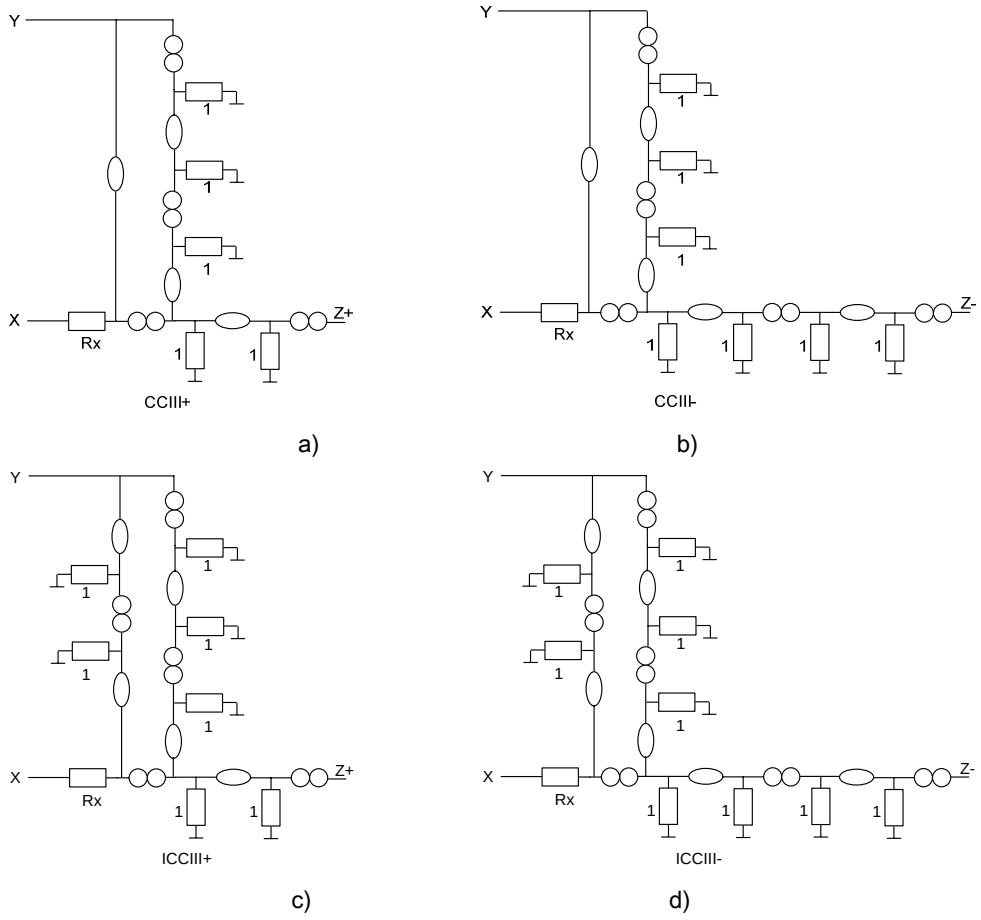


Figure 2.11 Equivalents of current conveyors using nullors a)CCIII+ b)CCIII- c)ICCIII+ d)ICCIII-

2.4 Data structure generation

In this section, the method that will be used throughout this Thesis will be presented, which consist in employing nullors to model devices, so they can be used with nodal analysis (NA) formulation.

The procedure that should be followed to create the NA formulation ($i=Yv$) to obtain a symbolic expression for a circuit is:

1) Model all circuit elements (active devices [5], controlled sources and independent voltage sources), by using nullors. In [5] and [11], the modeling processes using nullor equivalents include grounded admittances as much as possible, because they have only one entry in the NA formulation [12], while floating ones may have up to four entries requiring more computational work [1].

2) Number all nodes in circuit, because they will be used as indexes in further steps. Also label nullators and norators.

3) Describe the interconnection relationships of norators Pj, nullators Oj.

4) Now two sets should be created, because they will allow indexing at Y matrix and creating i and v vectors. Also two tables are listed, enumerating admittances within the circuit which will be used to fill admittance matrix.

a. Set ROW: It contains all nodes (ordered) calculated by using the interconnection relationships (IR's) and properties of the norator, whose nodes (m,n) are virtually short-circuited. These indexes are associated to rows and are used to fill vector i and the admittance matrix Y.
b. Set COL: It contains all nodes (ordered) calculated by using the IRs and properties of the nullator, whose nodes (m,n) are virtually short-circuited. These indexes are associated to columns and are used to fill vector v and the admittance matrix Y.

c. Tables for admittances: Admittances are structured into two tables: Table A consists of all nodes (ordered), and in each node is the sum of all admittances connected to it. Table B consists of all floating admittances and its nodes (m,n).

5) Use sets ROW and COL to fill vector ***i*** and ***v***, respectively. To fill the admittance matrix ***Y***: if in Table A, a node is included in sets ROW and COL (Cartesian product described in [13]), introduce that admittance(s) in ***Y*** with the corresponding row (from ROW index) and column (from COL index). For each floating admittance connected between nodes (m,n) in Table B, search node m in set ROW and node n in set COL (do the same but now search n in ROW and m in COL), if both nodes exist that admittance is introduced in ***Y*** with the corresponding row (from ROW index) and column (from COL index), and it is negative.

2.5 Illustrative example.

A small circuit will be used as an example to clarify the steps that should be followed. In this case, it is a non inverting amplifier, shown in figure 2.12

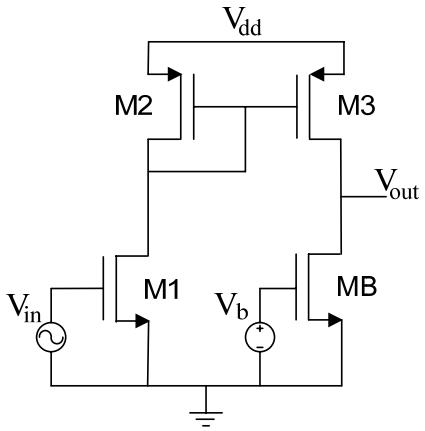


Figure 2.12 Non inverting amplifier

To substitute elements in circuit of figure 2.11 we start using model for transistors that appears in figure 2.3. For the case of the DC sources, the terminals between they are connected must be grounded. For the independent AC voltage source, the model using nullors of figure 2.13 will be employed.

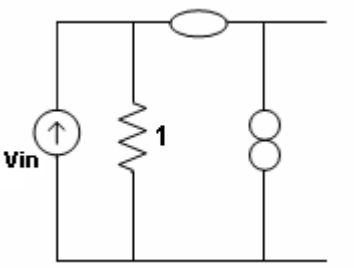


Figure 2.13 Nullor-based model for voltage source

Now the equivalent circuit for the non inverting amplifier is presented in figure 2.14, where the nodes connected to the DC voltage sources have been grounded, and the AC source at input has been substituted by the equivalent of figure 2.13.

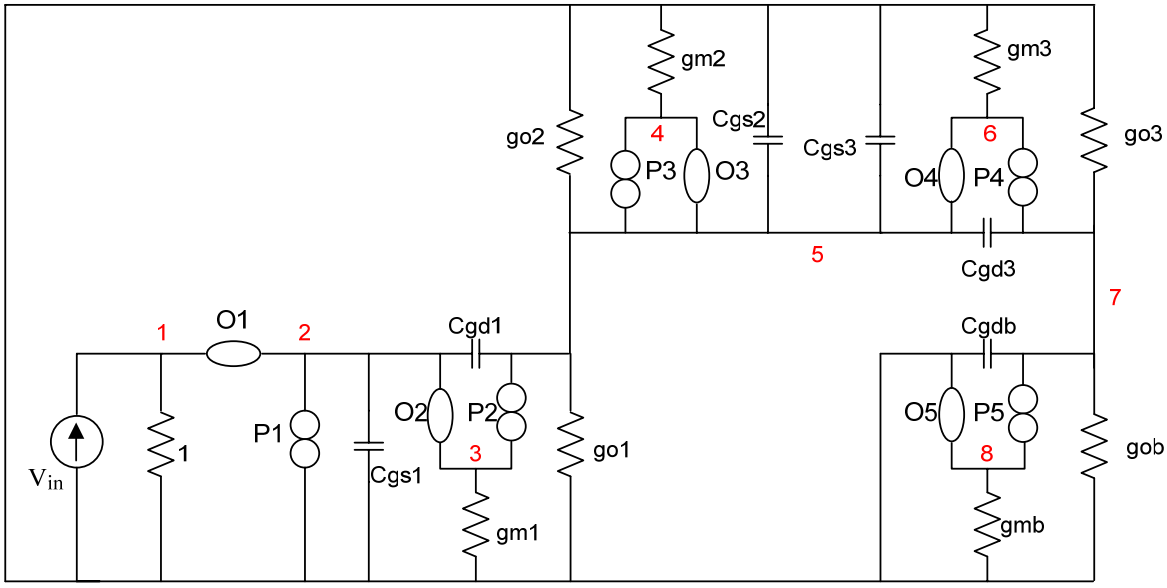


Figure 2.14 Non-inverting amplifier using nullor equivalents

As it can be seen in figure 2.14 the capacitors Cgd2 and Cgsb where taken out, the former because of the short circuit connection due to the diode connection, and the latter because of the result of grounding the biasing DC voltage source.

It can also been appreciated that nodes have been enumerated and norators and nullators have been labeled.

Now that the circuit is ready to be analyzed, the next steps can be presented.

Let's start describing the nullators and norators interconnection relationships by using two tables, which are now presented.

Nullator (O)	Associated nodes
O1	(1,2)
O2	(2,3)
O3	(4,5)
O4	(5,6)
O5	(8,0)

Norator (P)	Associated nodes
P1	(2,0)
P2	(3,5)
P3	(4,5)
P4	(6,7)
P5	(7,8)

Continuing with the steps now sets COL and ROW (named after column and row) will be created, they are constructed from previous tables.

Once the interconnection relationships (IR) have been defined, now it has to be checked which of norators and nullators are virtually connected, allowing a reduction in the order of the matrix.

For norators, P1 is virtually grounded; P2 and P3 are virtually connected and the same case applies for P4 and P5.
In case of nullators, O1 and O2 can be related in a unique set, the same case is for O3 and O4, and O5 is virtually grounded.

As it can be seen on tables, for the case of nullators, node 7 is not listed, but it has to be considered in COL set, which also happens for node 1 in norators considering this, the ROW and COL sets can be defined as:

ROW = {(1),(3,4,5),(6,7,8)}

COL = {(1,2,3),(4,5,6),(7)}

Now tables corresponding to admittances (A and B tables) should be created, the first one (A) contains admittances that are connected to each of the nodes (excluding datum node) and the second (B) corresponds to elements that are floated, that is, those elements whose nodes are not grounded.

In this way, tables A and B are presented as:

Table A

Nodes	Admittances
1	1
2	sCgs1+sCgd1
3	gm1
4	gm2
5	go1+go2+sCgd1+sCgd2+sCgs2+sCgs3
6	gm3
7	sCgd3+sCgdb+go3+gob
8	gmb

Table B

Floating admittances	Nodes
sCgd1	(2,5)
sCgd3	(5,7)

Once the sets and tables are ready, the next step can be applied.

To formulate the Cartesian product, ROW and COL sets are combined, for the present case; we have then the next combinations, corresponding to each of the entries of Y matrix.

(1,1)+(1,2)+(1,3)	(1,4)+(1,5)+(1,6)	(1,7)
(3,1)+(3,2)+(3,3)+ (4,1)+(4,2)+(4,3)+ (5,1)+(5,2)+(5,3)	(3,4)+(3,5)+(3,6)+ (4,4)+(4,5)+(4,6)+ (5,4)+(5,5)+(5,6)	(3,7)+(4,7)+(5,7)
(6,1)+(6,2)+(6,3)+ (7,1)+(7,2)+(7,3)+ (8,1)+(8,2)+(8,3)	(6,4)+(6,5)+(6,6)+ (7,4)+(7,5)+(7,6)+ (8,4)+(8,5)+(8,6)	(6,7)+(7,7)+(8,7)

According to this table representing Y admittance matrix, there are elements of tables A and B that correspond to entries of this matrix, in the case of table A, each of the elements correspond to a pair, that is 1 represents (1,1), 2 represents (2,2), etc.

For elements of table B there are two cases, the first one is if the entry is considered as it appears, let's take for instance Cgd1, which is connected between nodes (2,5), but it can also appear as connected between nodes (5,2), in both cases, the sign with which it entries the matrix is negative.

Then, filling in the matrix, we have:

1	0	0
gm1-sCgd1	gm2+go1+go2+sCgd1+sCg d2+sCgs2+sCgs3	-sCgd3
0	Gm3-sCgd3	go3+gob+sCgd3+sCgdb

Matrix $\boldsymbol{Y}$ is filled, but in order to have the system $\boldsymbol{i=Yv}$ complete, two vectors are necessary, one for currents and one for the variables to look after (voltages).

Voltages correspond to COL set, so the voltage vector is $\boldsymbol{v}=[v_{1,2,3},v_{4,5,6},v_7]^T$

The current vector is $\boldsymbol{i}=[v_{in}, 0, 0]^T$ which was created after checking which independent current sources are connected to each of the sets of nodes inside the ROW set.

Now the complete structure corresponding to the circuit can be presented

$$\begin{bmatrix} 1 & 0 & 0 \\ \text{gm1} - \text{sCgd1} & \text{gm2} + \text{go1} + \text{go2} + sCgd1 + sCgd2 + sCgs2 + sCgs3 & -sCgd3 \\ 0 & gm3 - sCgd3 & go3 + gob + sCgd3 + sCgdb \end{bmatrix} \begin{bmatrix} v_{1,2,3} \\ v_{4,5,6} \\ v_7 \end{bmatrix} = \begin{bmatrix} v_{in} \\ 0 \\ 0 \end{bmatrix}$$

As it can be seen, the generated system corresponds to a 3 x 3 admittance matrix.

To solve this last system of equations, different methods can be employed, for example, determinant decision diagrams (DDD). [14]-[16].

2.6 Conclusion

As it has been seen in this chapter, nullor is useful to model different devices, such as MOS transistor and other active devices like opamps or current conveyors.

In this chapter, a nodal formulation was presented, where it was observed the usefulness of modeling using nullors, which allow using only nodal analysis. The method was illustrated by using a circuit as example.

Chapter 3. Simplification approaches

3.1 Non dominant elements discrimination

There are different approaches to neglect non dominant elements within circuits, which allow simplifying symbolic expressions, in order shorter results easier to be handled can be obtained.

These approaches can be considered as methods to implement simplifications before (SBG), during (SDG) or after the generation (SAG) of symbolic expressions [12].

3.1.1 Simplification before generation

This method is based on the reduction of the original circuit, so the symbolic computations are performed on the simplified circuit [1].

Some heuristics can be considered to perform simplification before generation. For example, it can be taken into account that output conductance of MOS transistor is smaller than its transconductance [8], so it can be neglected in the circuit being analyzed.

Also, if the output conductance of a MOS is set in the output of a stage it should be taken into account (for those multi-stage amplifiers), if it is set in an input branch it could be taken away.

Some elements could also be taken away, like those capacitors belonging to transistors connected in a diode configuration, or any other element parallel-connected to a short circuit.

The gate to source capacitance tends to be larger than gate to drain capacitor [17] so Cgs capacitor could be considered and Cgd neglected.

However those capacitors are usually smaller than compensation or load capacitors, and could be discarded if the latter are present.

If a MOS transistor is used for biasing it could not be taken into account for the global symbolic expression.

There exists also an approach given by the signal-path approximation (SPA) to perform SBG, taken from reference [17] it can be summarized as:

Model each MOST and independent voltage sources with their nullor-based model, and do not include parasitic capacitors and output conductance for MOST diode connected. Then reconstruct the circuit so it can be observed which elements are not connected to the signal path and then neglected. Finally, nodes containing more capacitors, and which are closed to the output node, are considered, discarding those nodes with few capacitors connected to them.

3.1.2 Simplification during generation

It is applied in the formulation process of the system of equations of a network [17] trying to generate directly the simplified expression.

There are techniques proposed on [23], [24], [25] that don't generate the exact expression but directly build the wanted simplified expression by generating the terms one by one in decreasing order of magnitude, until the approximation error gets a defined value [8].

3.1.3 Simplification after generation

Usually, simplification after generation is used once the complete expression has been generated in an expanded format [1].

There are also some approaches to perform SAG, one is based in comparing numerical values for each parameter, to improve the interpretability of the expressions [18]. This comparison is done to neglect some elements which are compared to larger ones.

Another approach consist in heuristically consider that the transconductance of a MOS transistor operating in saturation is about 100 times larger than its output conductance, neglecting the latter.

Terms can also be eliminated from symbolic expression by cancelling common symbols between two terms into a sum of products by carrying out a quotient operation [17].

The moment matching lays in the simplification categories, and it works from the computation of the symbolic transfer function. The final expression is an order reduced one, but the symbolic expressions might increase.

In order the topics of the last chapter and the approach given for neglecting non dominants terms can be exemplified, a circuit will be next presented.

3.2 Illustrative example

The circuit used to exemplify the procedure is a p-Miller amplifier, represented in figure 3.1.

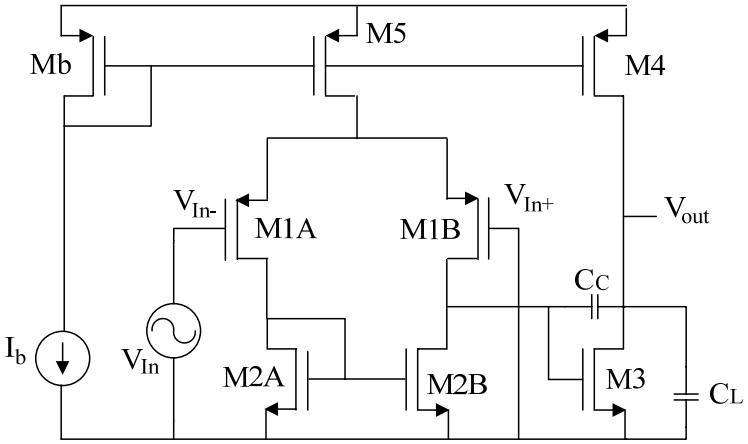


Figure 3.1 Miller amplifier

As it was established in last chapter, elements of a circuit should be substituted by their equivalents using nullors, so a NA formulation can be performed, this way, Miller amplifier using nullors can be represented as it appears on figure 3.2.

As a first approach of SBG, it can be seen that according to figure 3.1, there is a transistor diode connected, so, it can be appreciated that capacitor Cgd2a is short circuited in figure 3.2, this way; it can be eliminated.

In the same way, capacitors Cgs4 and Cgs5 are in parallel with a short circuit, which is a result of grounding those nodes, used for biasing, so those capacitors can be taken away and not considered for the symbolic expression.

It can also be noted that Mb equivalent was not taken into account, because it is a transistor used for biasing.

Redrawing the circuit again, without the capacitors, it appears on figure 3.3

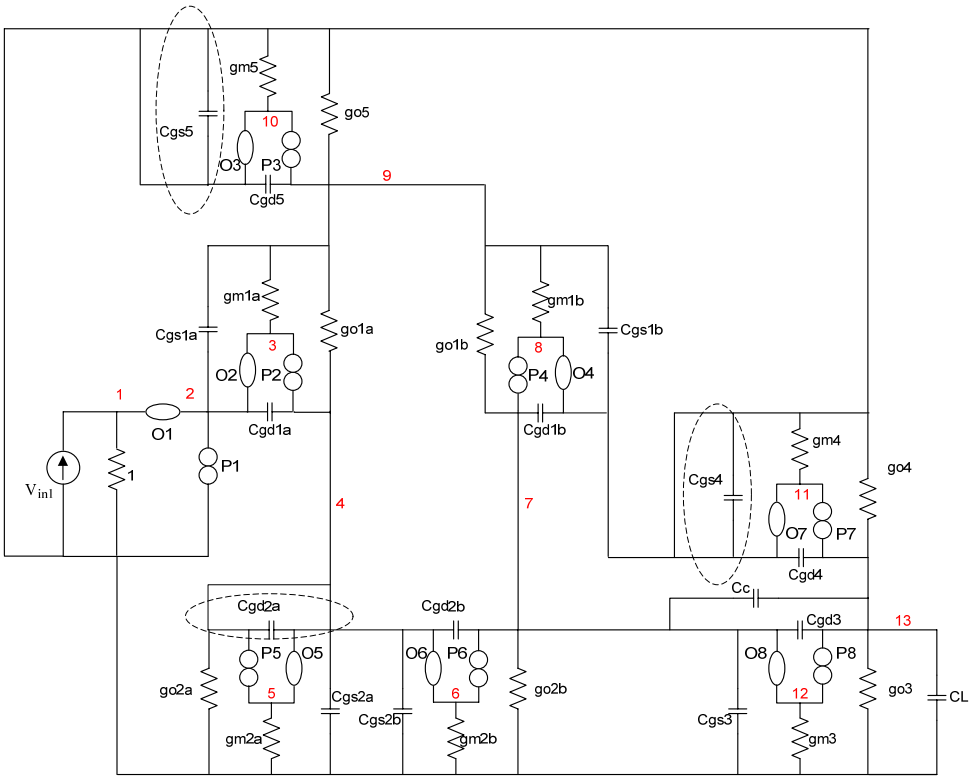


Figure 3.2 Miller amplifier using nullor equivalents

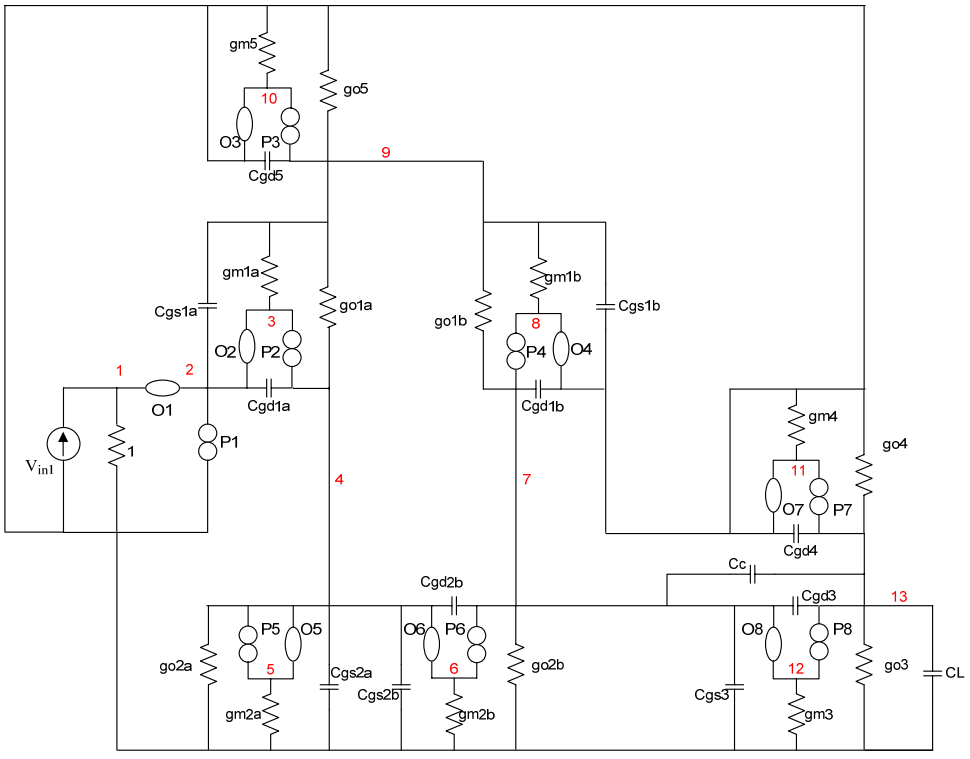


Figure 3.3 Equivalent circuit for Miller amplifier

Parting from circuit on figure 3.3 the ROW and COL sets for this circuit are:

ROW = {(1),(3,4,5),(6,7,8),(9,10),(11,12,13)}

COL = {(1,2,3),(4,5,6),(7,12),(9),(13)}

And the admittance tables for the circuit are:

Table A

Nodes	Admittances
1	1
2	sCgs1a+sCgd1a
3	gm1a
4	go1a+go2a+sCgd1a+sCgd2b+sCgs2a+sCgs2b
5	gm2a
6	gm2b
7	go1b+go2b+sCgd1b+sCgd2b+sCc+sCgd3+sCgs3
8	gm1b
9	go1a+go1b+gm1a+gm1b+go5+sCgd5+sCgs1a+sCgs1b
10	gm5
11	gm4
12	gm3
13	go3+go4+sCgd3+sCgd4+sCc+sCL

Table B

Floating admittances	Nodes
sCgd1a	(2,4)
sCgd1a	(2,9)
gm1a	(3,9)
sCgd2b	(4,7)
go1a	(4,9)
sCc+sCgd3	(7,13)
gm1b	(8,9)

Then the admittance matrix **Y** is:

1	0	0	0	0
gm1a-sCgd1a	go1a+go2a+sCgd1a+sCgd2b+sCgs2a+sCgs2b+gm2a	-sCgd2b	-gm1a-go1a	0
0	gm2b-sCgd2b	go1b+go2b+sCgd1b+sCgd2b+sCdg3+sCgs3+sCc	-gm1b	-sCc-sCgd3
-sCgs1a-gm1a	-go1a	0	go1a+go1b+gm1a+gm1b+go5+sCgd5+sCgs1a+sCgs1b	0
0	0	gm3-sCc-sCgd3	0	go3+go4+sCc+sCgd3+sCgd4+sCL

And vectors for current sources and voltages are:

$\boldsymbol{v}=[v_{1,2,3},v_{4,5,6},v_{7,12},v_9,v_{13}]^T$

$\dot{\boldsymbol{i}}=[v_{in}, 0, 0,0,0]^T$

In this case, the voltage transfer function representing amplifier gain will be considered, so v_{13}/v_{in} should be found, we get a symbolic expression consisting in 119 terms in numerator and 2147 in denominator, so simplifications are seen as a solution to try to get a shorter expression for this circuit.

To make a comparison between the resulting expression and the result obtained through a numerical simulation of the circuit, values are substituted into the expression, and then plotted. The two graphs can be seen in the figure 3.4, which shows that there is an agreement between both results.

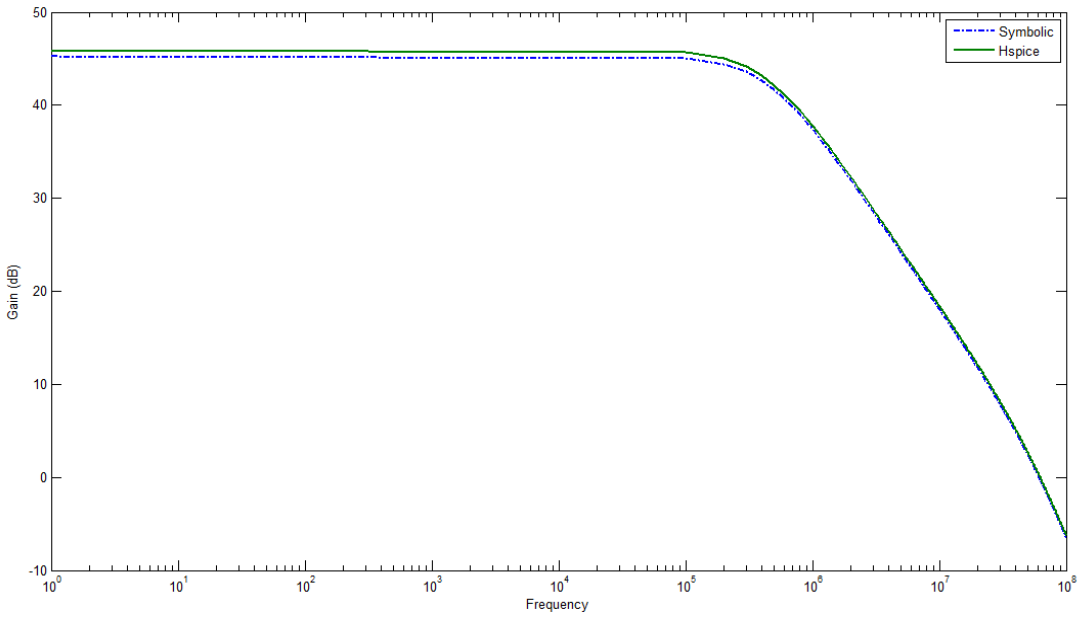


Figure 3.4. Output generated from symbolic transfer function, and comparison with numerical simulation

The phase response of the resulting expression is also compared to the one obtained through numerical simulation, the plot corresponding to this result can be seen in figure 3.5.

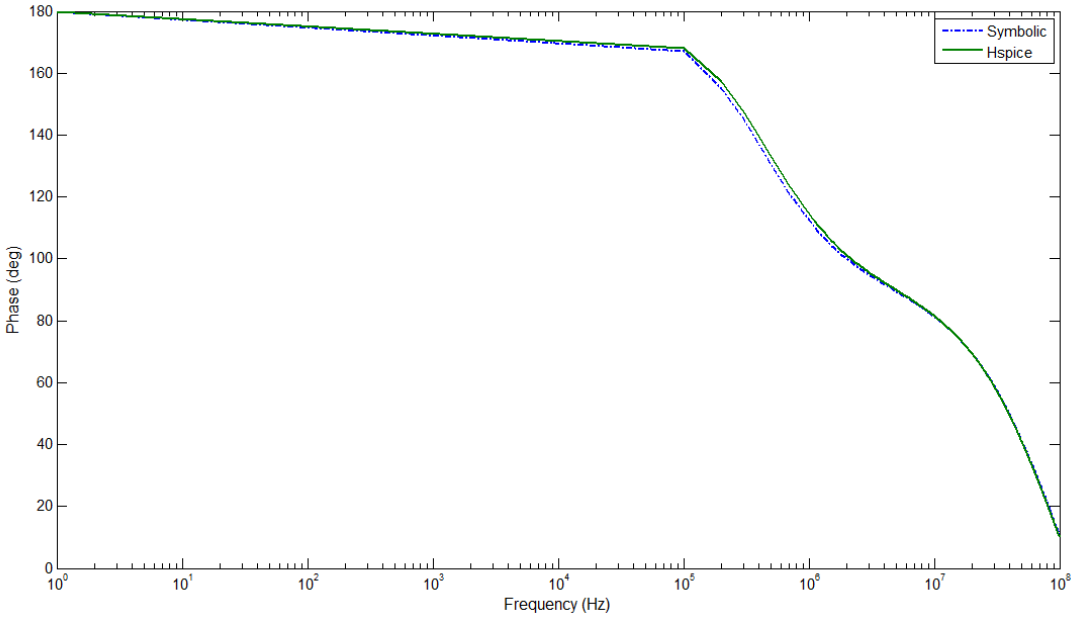


Figure 3.5 Phase response of Miller amplifier

In order to try to reduce this expression, a simplification before it is generated can be carried out. To achieve this, three resistors can be taken away (go5, go1a and go2a), which are connected in the first branch of the circuit, getting the configuration that appears on figure 3.6.

Generating the nodal formulation for system ***i=Yv*** again, and solving to get the voltage gain, a new expression is obtained, but it is still very large, because it has 83 terms in numerator and 174 in denominator

In the circuit analyzed, capacitors of compensation and load are of larger order than parasitic capacitors, which allows neglecting these.

Once again, transfer function is obtained, parting from the circuit without parasitic capacitors, it is now:

$$tf := (gm1a\ ((Cc\ go1b\ gm2b + Cc\ gm1b\ gm2b + gm1b\ gm2a\ Cc)\ s - gm3\ gm1b\ gm2b - gm3\ go1b\ gm2b - gm3\ gm1b\ gm2a))\ /\ (gm2a\ ((gm1b\ Cc\ CL + gm1a\ Cc\ CL + go1b\ Cc\ CL)\ s^2 + (go1b^2\ Cc + gm3\ go1b\ Cc + gm1a\ Cc\ go3 + go1b^2\ CL + go1b\ Cc\ go3 + gm3\ gm1b\ Cc + gm1a\ Cc\ go4 + gm1b\ go1b\ CL + go1b\ Cc\ go4 + gm1b\ Cc\ go4 + gm1b\ go1b\ Cc + Cc\ go1b\ gm1a + gm3\ gm1a\ Cc + gm1a\ go1b\ CL + gm1b\ Cc\ go3)\ s + gm1a\ go1b\ go3 + gm1b\ go1b\ go3 + gm1b\ go1b\ go4 + gm1a\ go1b\ go4 + go1b^2\ go4 + go1b^2\ go3))$$ (3.1)

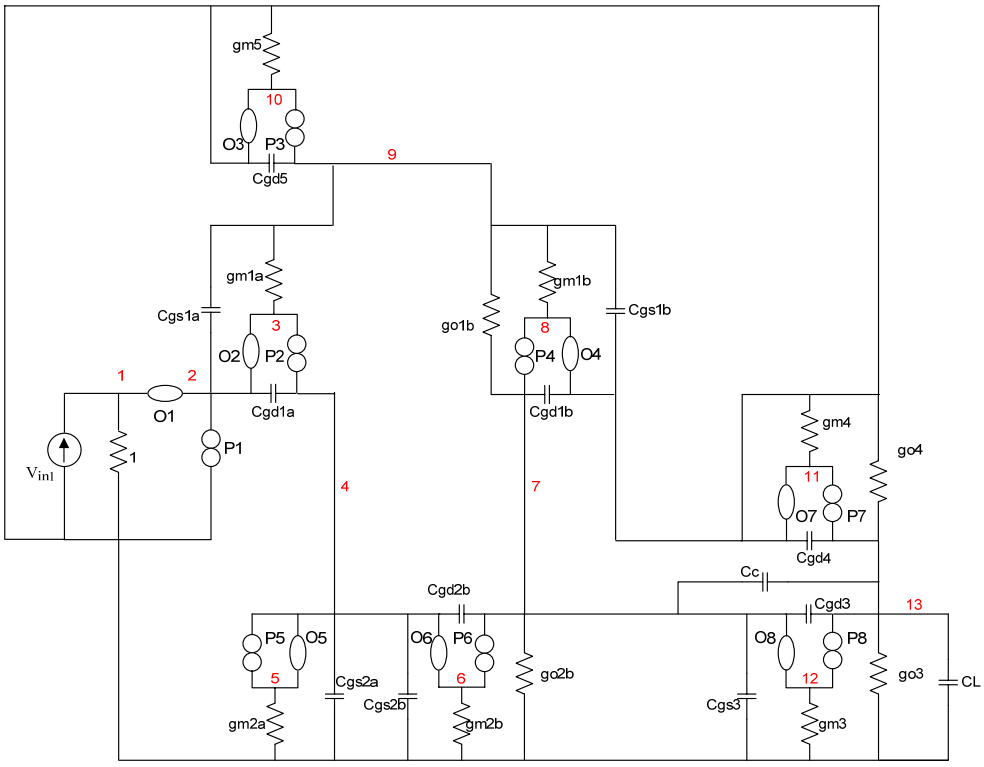


Figure 3.6 Miller amplifier after taking resistors out

Expression (3.1) has 6 terms in numerator and 24 in denominator, so it is significantly shorter than the first expression obtained, but it can still be done a little shorter, if a simplification after generation (SAG) is applied.

For this case, the way to get a shorter expression is neglecting those terms whose magnitude won't contribute significantly to the result; so, let's see which values could be

taken away. When the numerical simulation was held, some values were obtained; these are which can be seen next

$$go1a := 50.8667e-6:$$
$$go1b := 50.8667e-6:$$
$$go2a := 16.5503e-6:$$
$$go2b := 16.5503e-6:$$
$$go3 := 38.0886e-6:$$
$$go4 := 51.8343e-6:$$
$$go5 := 27.3602e-6:$$
$$gm1a := 721.5256e-6:$$
$$gm1b := 721.5256e-6:$$
$$gm2a := 543.4098e-6:$$
$$gm2b := 543.4098e-6:$$
$$gm3 := 1.6814e-3:$$
$$gm4 := 2.0155e-3:$$
$$gm5 := 1.1318e-3:$$

From these values, it can be noted that the transistor transconductance is bigger than output transistor conductance, taking this into account to reduce the expression, and grouping similar terms we can get the expression (3.2).

$$tf := (gm1a\ ((gm1b\ gm2a + gm1b\ gm2b)\ s \cdot Cc - gm3\ gm1b\ gm2b - gm3\ gm1b\ gm2a)) \Big/ (gm2a\ ((gm1a + gm1b)\ \cdot cC \cdot CL \cdot s^2 + (gm3\ gm1b + gm3\ gm1a)\ s \cdot Cc + gm1b\ go1b\ go4 + gm1a\ go1b\ go4 + gm1b\ go1b\ go3 + gm1a\ go1b\ go3)) \tag{3.2}$$

In this last expression, there are four terms in numerator and eight terms in denominator, which is a significant reduction from the original one, obtained using al terms.

A comparison between expression 3.2 and the response given for the circuit in HSPICE is presented in figure 3.7, so it can be seen that there is still an agreement between these two plots. The difference between the two plots is a result of taken away elements. In figure 3.8, it can also been seen the phase response for this amplifier and the comparison done with the numerical simulation.

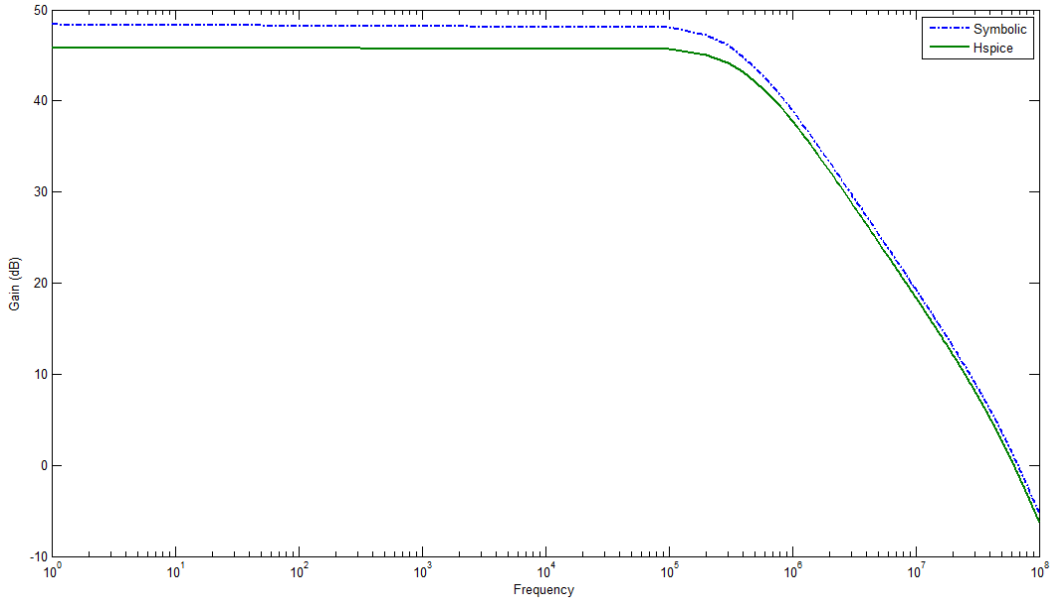


Figure 3.7 Comparison of plots resulting from simplified expression and numerical simulation

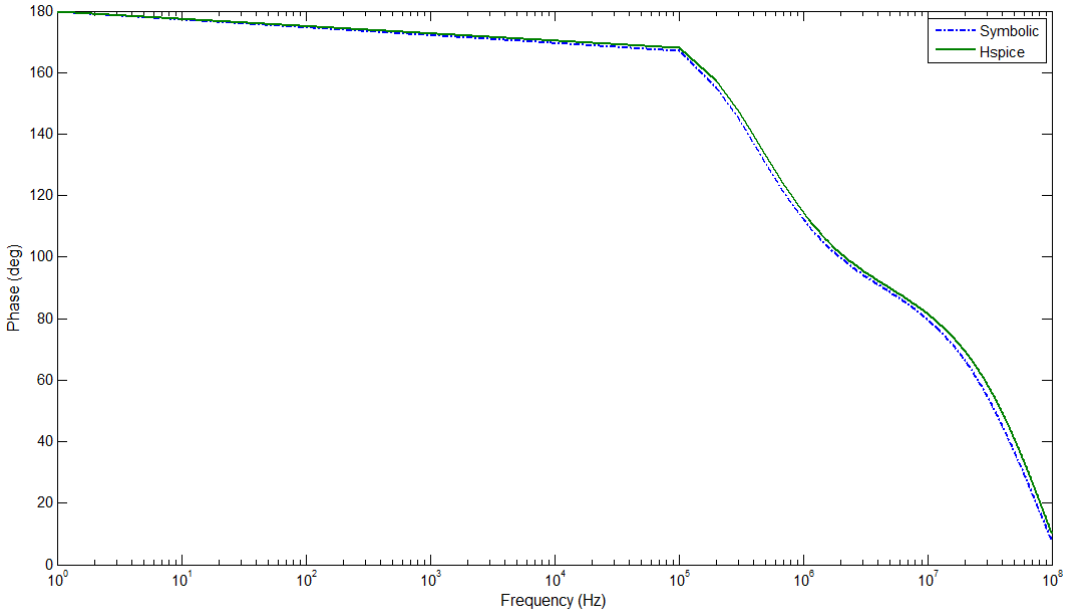


Figure 3.8 Phase response given for reduced expression of Miller amplifier

Finally, a table to summarize the results of simplifying expressions will be presented, where the number of terms in the expression can be seen, along with the method used to simplify it.

Expression	Number of terms in numerator	Number of terms in denominator
Complete	119	2147
First simplification before generation	83	174
Second simplification before generation	6	24
Simplified before and after generation	4	8

Table 3.1 Reduction of terms in symbolic expression

3.3 Conclusion

In this chapter, some heuristics were presented, which allow reducing symbolic expressions, so they can be easier to be interpreted.

There are different approaches for these simplifications, if a reduction on the circuit is performed, it is considered as a simplification before generation. If the simplification occurs when the circuit is being analyzed, it is considered as a simplification during generation. If the reduction is carried out when the expression has already been obtained, it is a simplification after generation.

A circuit was used as an example to illustrate the usefulness of these simplification approaches, and the agreement between plots comparing the result when using a numerical simulator and the obtained expression gives us the idea that the procedure is correct.

Chapter 4. Behavioral model generation of analog circuits

Considering what was mentioned in last chapters, some circuits will be analyzed following the given methodology.

4.1 Common source amplifier (resistive load)

A single transistor amplifier will be the first example presented here, it is the known inverter topology given by an amplifier with a resistive load and common source, the circuit at transistor level is presented in figure 4.1.

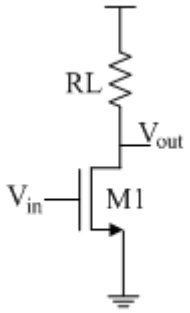


Figure 4.1 Common source amplifier

Using the model of transistor given in figure 2.3 and the model for voltage source at input taken from figure 2.13, the equivalent circuit is obtained, where it can be appreciated (figure 4.2) that there are two nullors.

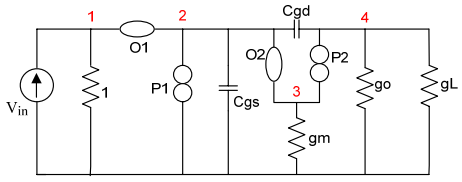


Figure 4.2 Nullor equivalent circuit of common source amplifier

As there are just few elements in this circuit, all of them will be taken into account to obtain the voltage transfer function of it.

Creating COL and ROW sets as already explained, we have:

COL ={(1,2,3),(4)}

ROW={(1),(3,4)}

Meanwhile, the tables where admittances are listed are here presented.

Table A

Nodes	Admittances
1	1
2	sCgs+sCgd
3	gm
4	sCgd+gL+go

Table B

Floating admittances	Nodes
sCgd	(2,4)

According to what was mentioned, the admittance matrix should be filled, performing a Cartesian product between COL and ROW sets, by doing this and filling in the adequate positions of the Y matrix, the system of equations corresponding to this amplifier is:

$$\begin{bmatrix} V_{in} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ gm - s\,Cgd & gL + go + s\,Cgd \end{bmatrix} \begin{bmatrix} V_{1,2,3} \\ V_4 \end{bmatrix}$$

And solving for V_4/V_{in}, the transfer function corresponding to this circuit is:

$$A_v = \frac{(-gm + s\,Cgd)}{(gL + go + s\,Cgd)} \qquad (4.1)$$

These result agrees with the one presented in [19]. A graph resulting from expression (4.1) can be seen in figure 4.3. And the phase response given for this result is given in figure 4.4

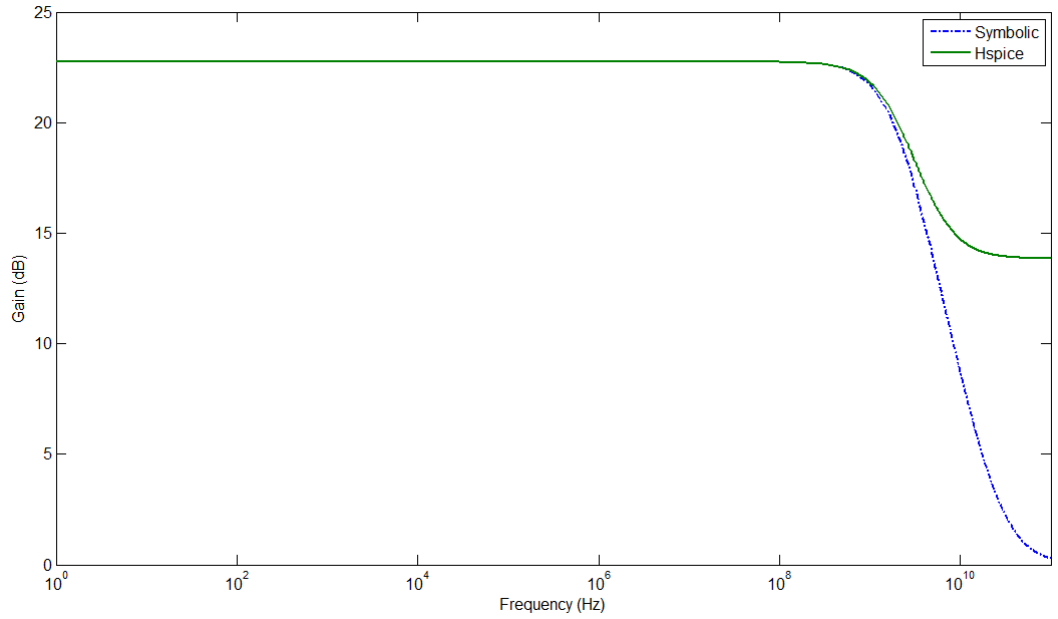


Figure 4.3 Symbolic and numeric results for common source amplifier with resistive load

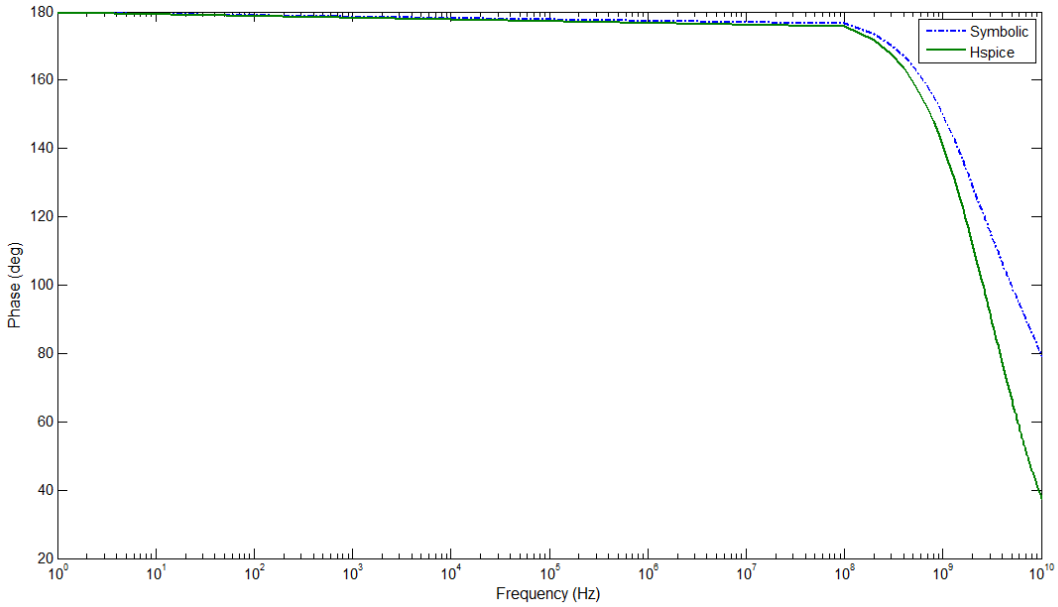


Figure 4.4 Phase response for common source amplifier with resistive load

4.2 Common source amplifier (active load)

A second single-ended stage circuit corresponds to the amplifier depicted in figure 4.5. As it is shown, the load resistance has been replaced by a transistor, using this as an active load.

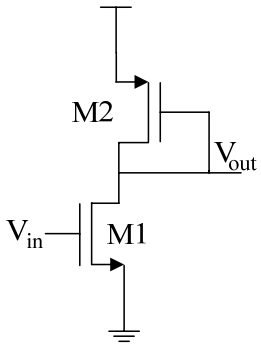


Figure 4.5 Common source amplifier with active load

This circuit will be substituted by the equivalents given for nullors, obtaining the circuit of figure 4.6

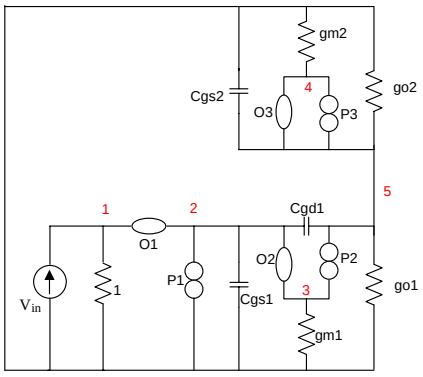


Figure 4.6 Common source amplifier with active load

As it can be appreciated, capacitor Cgd2 doesn't appear in figure 4.6 because of the short circuit connection present in M2.

Let's now formulate the nodal analysis for this circuit. So the corresponding sets and tables should be obtained.

In first instance, COL and ROW sets are formulated, and they are as it can be see now.

COL ={(1,2,3),(4,5)}
ROW={(1),(3,4,5)}

Tables for admittances are:

Table A

Nodes	Admittances
1	1
2	sCgs1+sCgd1
3	gm1
4	gm2
5	sCgd1+sCgs2+go1+go2

Table B

Floating admittances	Nodes
sCgd1	(2,5)

Now, formulating the CSE, we have:

$$\begin{bmatrix} V_{in} \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ gm1 - s\,Cgd1 & gm2 + s\,Cgd1 + s\,Cgs2 + go2 + go1 \end{bmatrix} \begin{bmatrix} V_{1,2,3} \\ V_5 \end{bmatrix}$$

The voltage transfer function given for this system is:

$$A_V = -\frac{gm1 - s\,Cgd1}{(Cgd1 + Cgs2)\,s + gm2 + go2 + go1} \qquad (4.2)$$

And a graph that illustrates the plot of this expression compared to one obtained through a numeric simulation is presented in figure 4.7, and the comparison for phase response is given in figure 4.8:

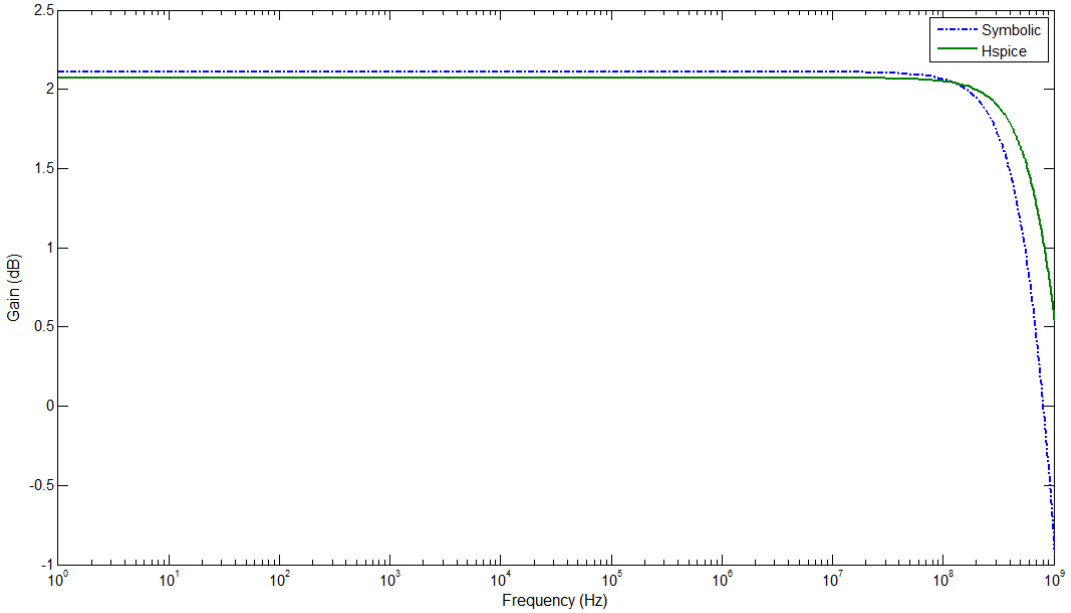


Figure 4.7 Symbolic and numeric results for common source amplifier with active load

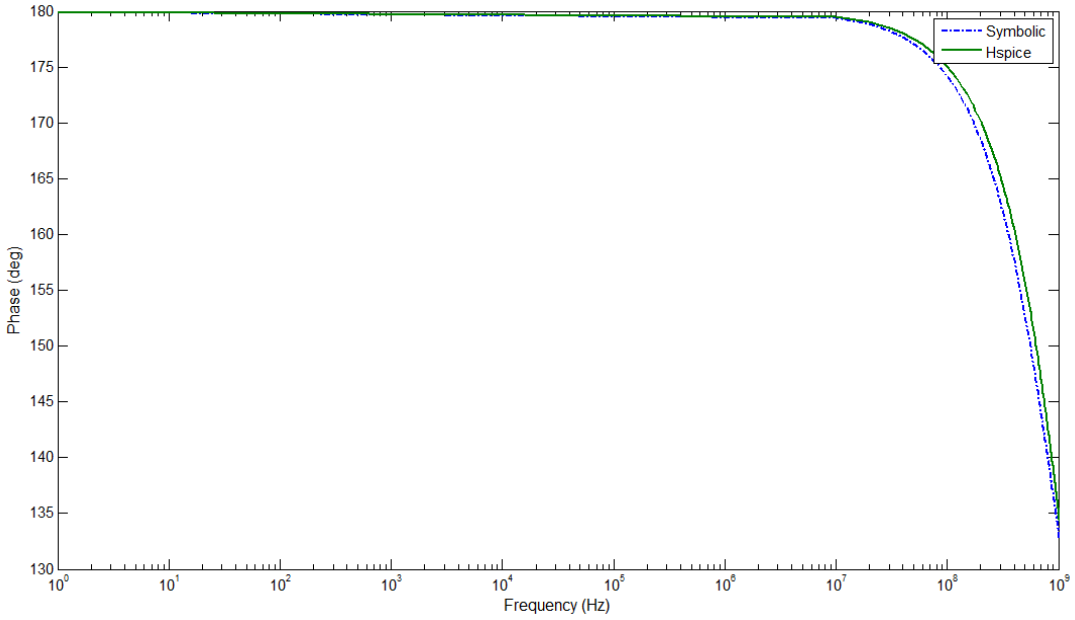


Figure 4.8 Phase response for common source amplifier with active load

Another example corresponding to one stage amplifiers is given by a differential pair. This will be presented now.

4.3 Differential pair

The circuit representing differential pair can be seen in figure 4.9

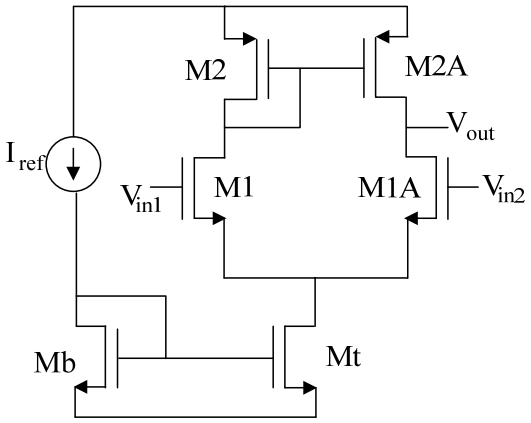


Figure 4.9 Differential pair

The equivalent circuit for differential pair when using the nullor equivalents can be seen in figure 4.10.

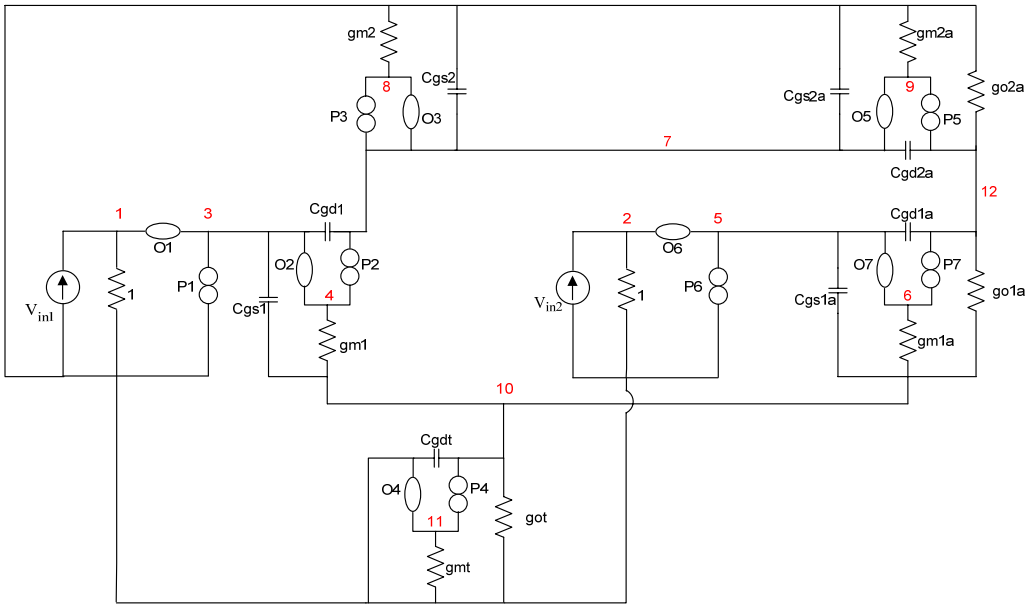


Figure 4.10 Differential pair equivalent circuit using nullors

From circuit in figure 4.10, it can be seen that conductances go1 and go2 were taken away, trying to obtain a reduced expression. Sets and tables for this circuit are obtained, these are:

ROW and COL sets:

ROW ={(1),(2),(4,7,8),(6,9,12),(10,11)}

COL = {(1,3,4),(2,5,6),(7,8,9),(10),(12)}

Tables for admittances:

Table A

Nodes	Admittances
1	1
2	1
3	sCgs1+sCgd1
4	gm1
5	sCgs1+sCgd1a
6	gm1a
7	sCgd1+sCgs2+sCgs2a+sCgd2a
8	gm2
9	gm2a
10	sCgs1+sCgs1a+sCgdt+gm1+gm1a+go1a+got
11	gmt
12	go1a+go2a+sCd2a+sCgd1a

Table B

Floating admittances	Nodes
sCgd1	(3,7)
gm1	(4,10)
sCgs1a	(5,10)
sCgd1a	(5,12)
gm1a	(6,10)
sCgd2a	(7,12)
go1a	(10,12)

The compact system of equations is:

$$\begin{bmatrix} Vin \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ gm1 - s\cdot Cgd1 & 0 & s\cdot Cgd1 + s\cdot Cgs2 + s\cdot Cgs2a + s\cdot Cgd2a + gm2 & -gm1 & 0 \\ 0 & gm1a - s\cdot Cgd1a & gm2a - s\cdot Cgd2a & -gm1a - go1a & go1a + go2a + s\cdot Cgd1a + s\cdot Cgd2a \\ -gm1 & -sCgs1a - gm1a & 0 & gm1 + go1a + gm1a + got + sCgs1 + sCgs1a + sCgdt & -go1 \end{bmatrix} \begin{bmatrix} V_{1,3,4} \\ V_{2,5,6} \\ V_{7,8,9} \\ V_{10} \\ V_{12} \end{bmatrix}$$

And solving for transfer function V_{12}/Vin we get expression (4.3).

In figure 4.11 there are plots to compare the results given for a numeric simulator and one obtained after numerical values were substituted in expression (4.3). The response obtained for phase can be seen in figure 4.12.

$$
\begin{aligned}
Av := \big(& (Cgd2a\ Cgs1a\ Cgd1 + Cgd2a\ Cgdt\ Cgd1 + Cgd2a\ Cgs1\ Cgd1)\ s^3 + (-gm2a\ Cgdt\ Cgd1 + go1a\ Cgd2a\ Cgd1 - gm2a\ Cgs1a\ Cgd1 - gm2a\ Cgs1\ Cgd1 \\
& + Cgd2a\ got\ Cgd1 + gm1\ Cgd2a\ Cgd1 - gm1\ Cgd2a\ Cgdt - gm1\ Cgd2a\ Cgs1a - gm1\ Cgd2a\ Cgs1 + gm1a\ Cgd2a\ Cgd1)\ s^2 + (go1a\ gm1\ Cgs2 \\
& - gm2a\ go1a\ Cgd1 + go1a\ gm1\ Cgs2a + gm1\ gm2a\ Cgs1a + gm1\ gm2a\ Cgs1 + gm1\ gm2a\ Cgdt - gm2a\ gm1a\ Cgd1 - gm2a\ got\ Cgd1 \\
& - gm1\ gm2a\ Cgd1 + go1a\ gm1\ Cgd1 - gm1\ Cgd2a\ got + gm1a\ gm1\ Cgd1 + gm1a\ gm1\ Cgs2 + gm1a\ gm1\ Cgs2a)\ s + gm1\ gm2a\ gm1a \\
& + go1a\ gm1\ gm2 + go1a\ gm1\ gm2a + gm1\ gm2a\ got + gm1a\ gm1\ gm2) \big/ \big((Cgdt\ Cgd2a^2 + Cgd2a\ Cgs1a\ Cgd1 + Cgs1a\ Cgd2a\ Cgs2 \\
& + Cgdt\ Cgd1a\ Cgd2a + Cgdt\ Cgd1a\ Cgs2a + Cgs1\ Cgd1a\ Cgs2a + Cgs1a\ Cgd1a\ Cgs2a + Cgdt\ Cgd1a\ Cgs2 + Cgs1a\ Cgd2a^2 + Cgdt\ Cgd2a\ Cgs2 \\
& + Cgdt\ Cgd2a\ Cgs2a + Cgs1a\ Cgd2a\ Cgs2a + Cgdt\ Cgd1a\ Cgd1 + Cgs1a\ Cgd1a\ Cgd1 + Cgs1a\ Cgd1a\ Cgs2 + Cgd2a\ Cgdt\ Cgd1 \\
& + Cgs1a\ Cgd1a\ Cgd2a + Cgs1\ Cgd1a\ Cgd1 + Cgs1\ Cgd2a^2 + Cgs1\ Cgd1a\ Cgs2 + Cgs1\ Cgd1a\ Cgd2a + Cgd2a\ Cgs1\ Cgd1 + Cgs1\ Cgd2a\ Cgs2 \\
& + Cgs1\ Cgd2a\ Cgs2a)\ s^3 + (gm1\ Cgd2a\ Cgd1 + gm1a\ Cgd2a\ Cgd1 + Cgd2a\ got\ Cgd1 + go1a\ Cgd2a\ Cgd1 + got\ Cgd2a^2 + Cgd2a^2\ go1a \\
& + gm1\ Cgd2a^2 + gm1a\ Cgd2a^2 + got\ Cgd1a\ Cgd2a + Cgs1a\ go1a\ Cgd1 + Cgs1a\ go1a\ Cgs2 + Cgs1a\ go1a\ Cgs2a + Cgs1a\ go1a\ Cgd2a \\
& + Cgdt\ go1a\ Cgd1 + go1a\ Cgd1a\ Cgs2 + go1a\ Cgd1a\ Cgs2a + go1a\ Cgd1a\ Cgd2a + got\ Cgd1a\ Cgd1 + Cgs1a\ Cgd2a\ gm2 + Cgs1\ Cgd1a\ gm2 \\
& + Cgd2a\ go1a\ Cgs2a + Cgdt\ Cgd1a\ gm2 + Cgs1\ go2a\ Cgd1 + Cgs1\ go2a\ Cgs2 + Cgs1\ go2a\ Cgs2a + gm1\ Cgd1a\ Cgd1 + gm1\ Cgd1a\ Cgs2 \\
& + gm1\ Cgd1a\ Cgs2a + Cgs1a\ Cgd1a\ gm2 + Cgs1a\ go2a\ Cgd1 + gm1a\ Cgd1a\ Cgd2a + gm1a\ Cgd1a\ Cgd1 + gm1a\ Cgd1a\ Cgs2 + Cgs1a\ go2a\ Cgs2 \\
& + Cgs1a\ go2a\ Cgs2a + Cgs1a\ go2a\ Cgd2a + gm1a\ Cgd1a\ Cgs2a + Cgs1\ Cgd2a\ gm2 + Cgs1\ go2a\ Cgd2a + Cgd2a\ go1a\ Cgs2 + go1a\ Cgd1a\ Cgd1 \\
& + Cgdt\ go1a\ Cgs2 + Cgdt\ go1a\ Cgs2a + Cgdt\ go1a\ Cgd2a + got\ Cgd2a\ Cgs2 + got\ Cgd2a\ Cgs2a + gm1\ Cgd1a\ Cgd2a + Cgdt\ Cgd2a\ gm2 \\
& + Cgdt\ go2a\ Cgd1 + Cgdt\ go2a\ Cgs2 + Cgdt\ go2a\ Cgs2a + gm1a\ Cgd2a\ Cgs2 + gm1a\ Cgd2a\ Cgs2a + got\ Cgd1a\ Cgs2 + got\ Cgd1a\ Cgs2a \\
& + Cgdt\ go2a\ Cgd2a + Cgs1\ go1a\ Cgd1 + Cgs1\ go1a\ Cgs2 + Cgs1\ go1a\ Cgs2a + Cgs1\ go1a\ Cgd2a + gm1\ Cgd2a\ Cgs2 + gm1\ Cgd2a\ Cgs2a)\ s^2 \\
& + (got\ Cgd1a\ gm2 + got\ go1a\ Cgs2a + Cgs1a\ go1a\ gm2 + go1a\ Cgd1a\ gm2 + gm1\ go2a\ Cgd2a + Cgd2a\ go1a\ gm2 + got\ go1a\ Cgd1 \\
& + got\ go1a\ Cgs2 + Cgs1a\ go2a\ gm2 + gm1\ Cgd1a\ gm2 + gm1a\ Cgd1a\ gm2 + Cgs1\ go2a\ gm2 + Cgdt\ go1a\ gm2 + got\ go2a\ Cgd1 + got\ go2a\ Cgs2 \\
& + got\ go2a\ Cgs2a + got\ go2a\ Cgd2a + got\ Cgd2a\ gm2 + go1a\ go2a\ Cgd1 + gm1a\ go2a\ Cgd1 + gm1a\ Cgd2a\ gm2 + gm1a\ go2a\ Cgs2 \\
& + gm1a\ go2a\ Cgs2a + gm1a\ go2a\ Cgd2a + go1a\ go2a\ Cgs2 + Cgdt\ go2a\ gm2 + Cgs1\ go1a\ gm2 + gm1\ Cgd2a\ gm2 + got\ go1a\ Cgd2a \\
& + gm1\ go2a\ Cgd1 + gm1\ go2a\ Cgs2 + gm1\ go2a\ Cgs2a + go1a\ go2a\ Cgs2a + go1a\ go2a\ Cgd2a + go1a\ gm1\ Cgd1 + go1a\ gm1\ Cgs2 \\
& + go1a\ gm1\ Cgs2a)\ s + got\ go2a\ gm2 + go1a\ go2a\ gm2 + go1a\ gm1\ gm2a + gm1\ go2a\ gm2 + gm1a\ go2a\ gm2 + go1a\ gm1\ gm2 + got\ go1a\ gm2) \big)
\end{aligned}
$$

(4.3)

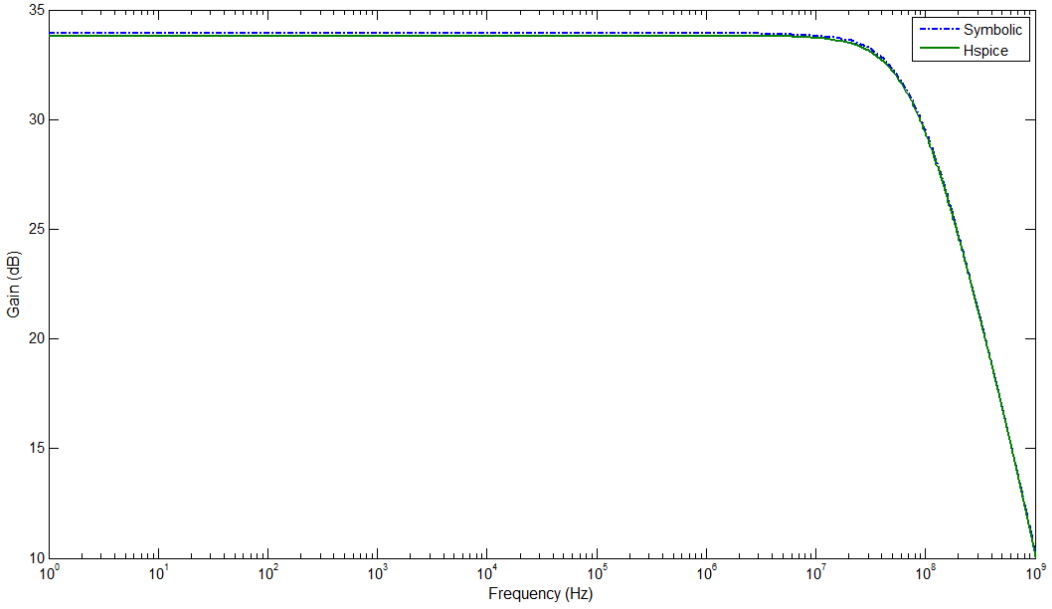


Figure 4.11 Symbolic and numeric results for differential pair amplifier

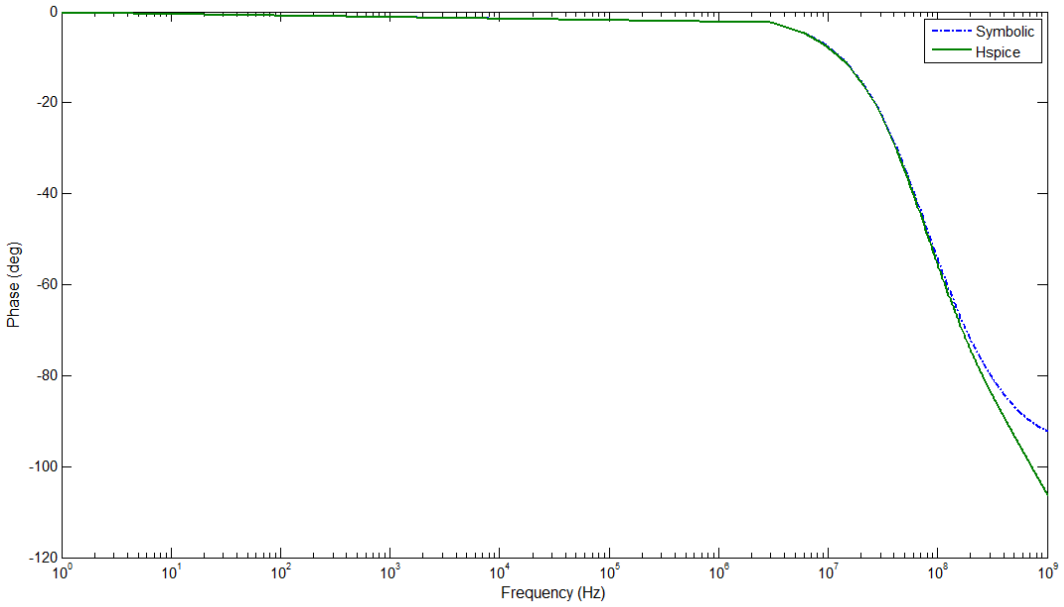


Figure 4.12 Phase response for differential pair amplifier

Now a three-stage circuit is presented.

4.4 Uncompensated amplifier

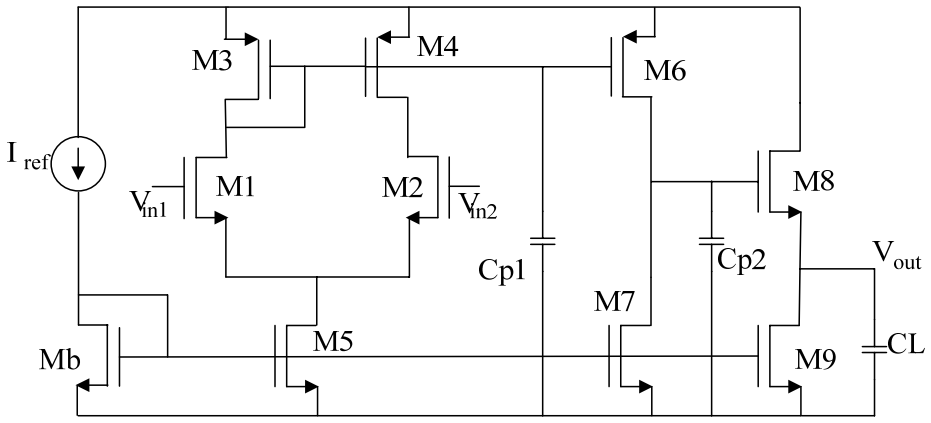


Figure 4.13 uncompensated amplifier

Substituting the circuit of figure 4.13 by using the nullor equivalents of each of the elements, the circuit of figure 4.14 is obtained.

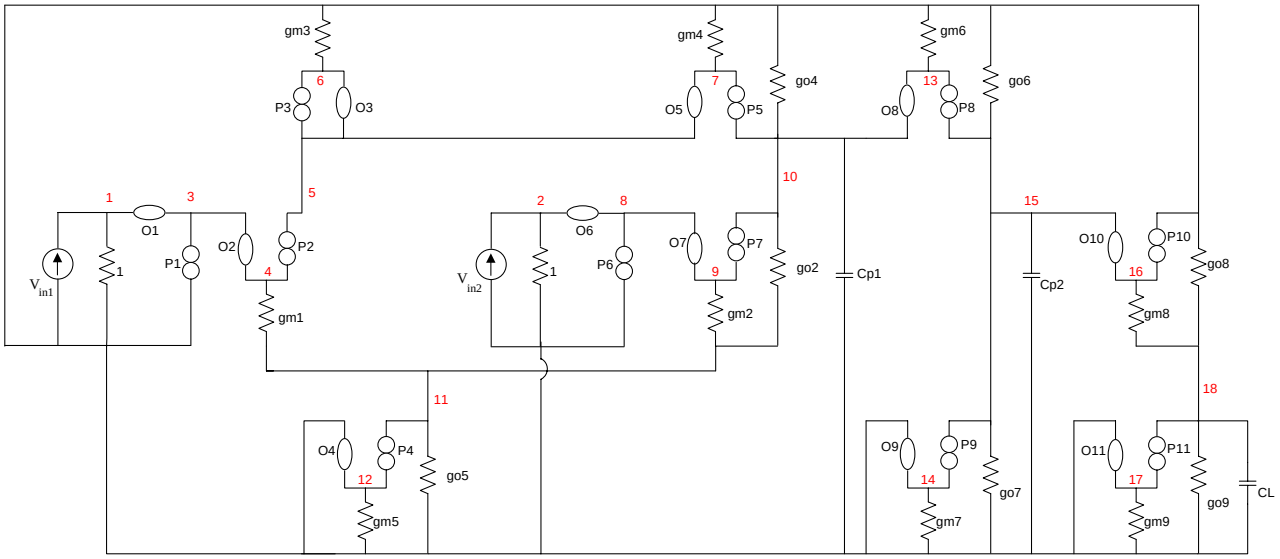


Figure 4.14 Uncompensated amplifier using nullors

As mentioned beforehand, drain-source transconductances were employed only in those branches corresponding to outputs (this circuit can be seen as a series of stages, each of those having and output). This way, the transconductances that were left are go2, go4, go5, go6, go7, go8 and go9. As it can be observed from figure 3.24, all parasitic capacitors were taken away, because the contribution to poles was mainly given capacitors named as Cp1, Cp2 and the load capacitor, C_L.

And the sets COL and ROW for this circuit are:

ROW ={(1),(2),(4,5,6),(7,9,10),(11,12),(13,14,15),(17,18)}

COL = {(1,3,4),(2,8,9),(5,6,7),(10,13),(11),(15,16),(18)}

And the tables corresponding to this circuit are:

Table A

Nodes	Admittances
1	1
2	1
3	0
4	gm1
5	0
6	gm3
7	gm4
8	0
9	gm2
10	go2+go4+sCp1
11	go2+go5
12	gm5
13	gm6
14	gm7
15	go6+go7+sCp2
16	gm8
17	gm9
18	go8+go9+sCL

Table B

Floating admittances	Nodes
gm1	(4,11)
gm2	(9,11)
go2	(10,11)
gm8	(16,18)

Constructing the system of equations corresponding to this circuit, we have:

$$\begin{bmatrix} V_{in1} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ gm1 & 0 & gm3 & 0 & -gm1 & 0 & 0 \\ 0 & gm2 & gm4 & go2+go4+s\,Cp1 & -go2-gm2 & 0 & 0 \\ -gm1 & -gm2 & 0 & -go2 & gm1+go1+gm2+go2 & 0 & 0 \\ 0 & 0 & 0 & gm6 & 0 & go6+go7+s\,Cp2 & 0 \\ 0 & 0 & 0 & 0 & 0 & -gm8 & go8+go9+gm8+s\,CL \end{bmatrix} \begin{bmatrix} V_{1,3,4} \\ V_{2,8,9} \\ V_{5,6,7} \\ V_{10,13} \\ V_{11} \\ V_{15,16} \\ V_{18} \end{bmatrix}$$

By solving this system, as in the case of the Miller amplifier, a very large expression is obtained, so, SAG should be performed.

After reducing the expression for the transfer function, the new one obtained is:

$$A_V = -\frac{gm8\,gm6\,gm1}{Cp1\,Cp2\,CL\,s^3 + Cp1\,Cp2\,gm8\,s^2 + (Cp1\,go6 + Cp1\,go7 + go4\,Cp2 + go2\,Cp2)\,gm8\,s + (go2\,go7 + go4\,go7 + go2\,go6 + go4\,go6)\,gm8} \quad (4.4)$$

Comparing the responses between the symbolic expression we have obtained and that given by an Hspice simulation we get the next graphs for magnitude and phase:

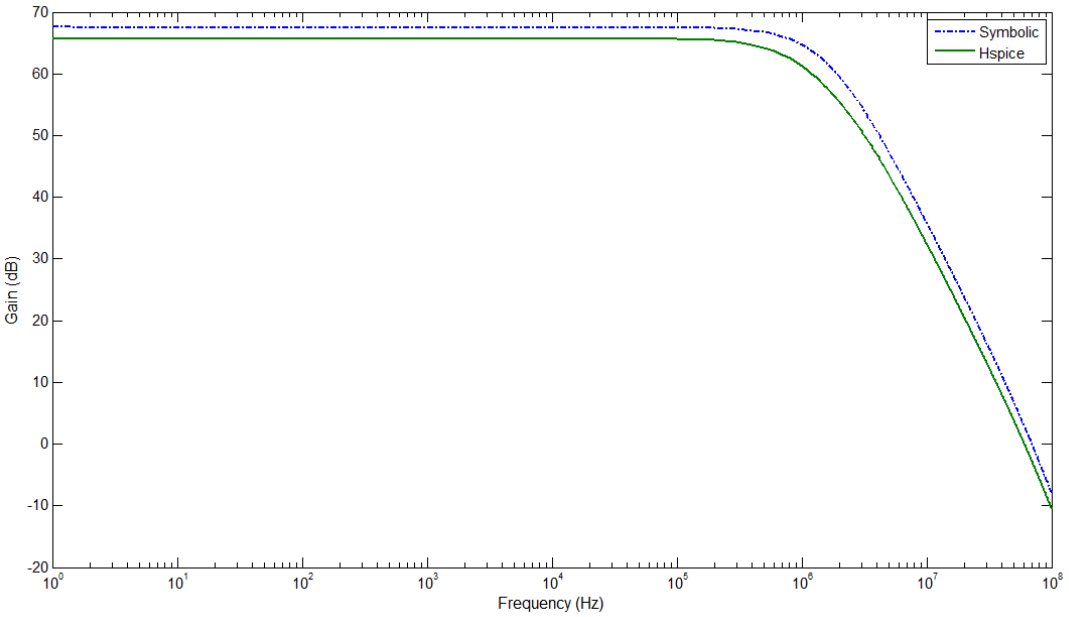


Figure 4.15 Symbolic and numeric results for uncompensated amplifier

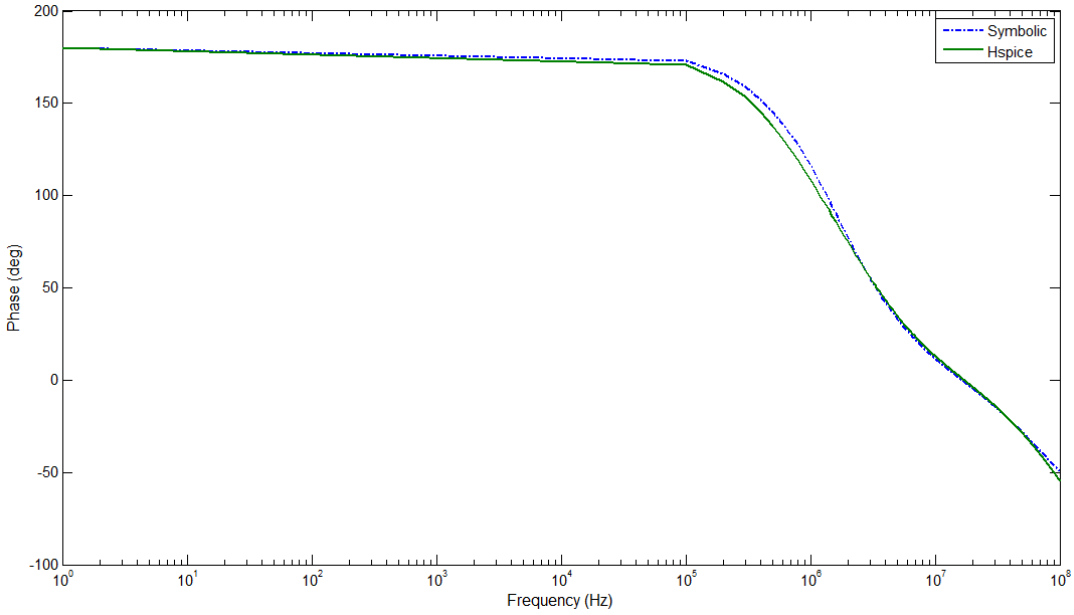


Figure 4.16 Phase response for uncompensated amplifier

4.5 Asymptotic Waveform Evaluation approach

As it was observed in previous chapters, when the number of elements in a circuit increases, so does the resulting expression, increasing also the order of it. In chapter one, the Padé approach was presented, which allows approximating a curve by using moments (Aymptotic Waveform Evaluation).

In this chapter, two examples will be presented to show how this method can be employed to obtain a reduced expression in some order.

4.6 Miller amplifier with AWE approach

Miller amplifier was presented in chapter 3 to illustrate the proposed method for the nodal formulation. Now it will be used to exemplify the approximation given by moments.

The expression representing Miller amplifier when using numerical values has a numerator of third order and a denominator of fourth order, which is now presented.

$$Av = \left(7.363891940\ 10^{11}\ s^4 + 2.427350752\ 10^{23}\ s^3 + 2.839700650\ 10^{34}\ s^2 + 3.451599276\ 10^{44}\ s - 6.451607630\ 10^{53}\right) \Big/ \left(8.193548824\ 10^{15}\ s^4 + 3.286128881\ 10^{26}\ s^3 + 3.145100272\ 10^{36}\ s^2 + 1.231027051\ 10^{45}\ s + 3.339898308\ 10^{51}\right) \tag{4.5}$$

In this case, the only symbolic variable to be used will be *s*, to show how this method can be employed.

To compute moments in order Padé approximation can be used, equation 1.10 will be here required, this way, first four moments resulting from expression (4.5) are:

$$M[0] = -193.1677864$$

$$M[1] = 0.00007130155122$$

$$M[2] = -2.609857244\ 10^{-11}$$

$$M[3] = 4.776172606\ 10^{-18}$$

Using these moments, system of equations (1.16) is created, so coefficients named *b* can be obtained. This system of equations is now presented.

$$\begin{bmatrix} -193.1677864 & 0.00007130155122 \\ 0.00007130155122 & -2.609857244\ 10^{-11} \end{bmatrix} \begin{bmatrix} b_2 \\ b_1 \end{bmatrix} = - \begin{bmatrix} 2.609857244\ 10^{-11} \\ -4.776172606\ 10^{-18} \end{bmatrix}$$

And solving this last system of equations, values of coefficients for denominator are calculated, these are:

$$b_1 = 0.00002207282740$$

$$b_2 = 8.012351799\ 10^{-12}$$

Using these values and those required moments, coefficients for numerator can be found, and they are:

$$a_0 := -193.1677864$$

$$a_1 := -0.004192457657$$

Once all the values are ready, the new transfer function representing Miller amplifier is:

$$H(s) = \frac{-0.004192457657\,s - 193.1677864}{8.012351799\ 10^{-12}\,s^2 + 0.00002207282740\,s + 1} \quad (4.6)$$

Performing a comparison between this last expression and graph given by the HSPICE simulation of the circuit, the plot of figure 4.17 is obtained. The phase response comparison can be observed in figure 4.18.

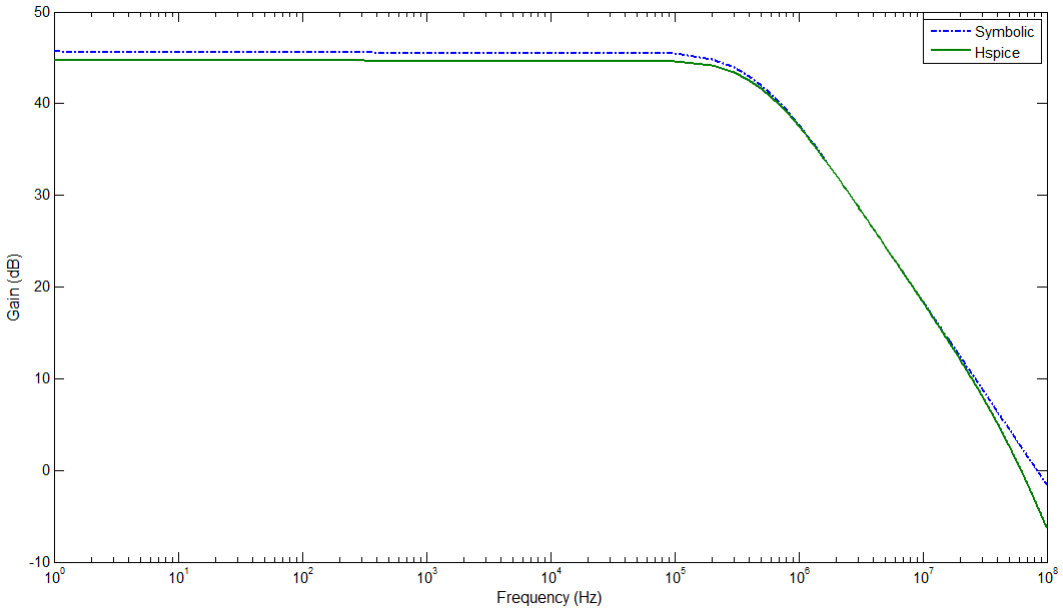


Figure 4.17 Response given for Miller amplifier with AWE approach

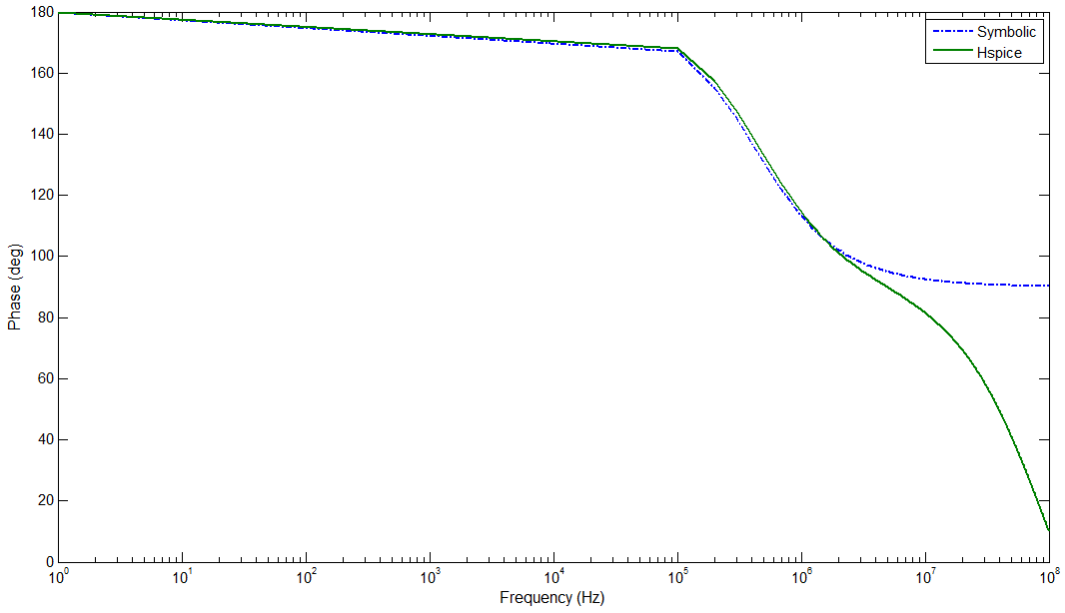


Figure 4.18 Phase response given for Miller amplifier with AWE approach

4.7 Uncompensated amplifier with AWE approach

In figure 4.10 this amplifier was presented, and the expression resulting after its analysis. In this case, a numerical expression with only a symbolic variable representing it is found, which is:

$$Av = \left(1536.\left(2.38350029\,10^{8}\,s^{5} + 2.755087082\,10^{20}\,s^{4} - 4.402376050\,10^{30}\,s^{3} - 4.199356001\,10^{41}\,s^{2} - 5.695516721\,10^{51}\,s - 1.948584729\,10^{61}\right)\right) \Big/ \left(7.676195087\,10^{18}\,s^{5} + 1.329024706\,10^{29}\,s^{4} + 3.508716518\,10^{38}\,s^{3} + 1.511009958\,10^{47}\,s^{2} + 3.295791001\,10^{54}\,s + 1.296371843\,10^{61}\right) \quad (4.7)$$

As it can be seen, this is a fifth-order expression in both, numerator and denominator; it will be reduced to a second order expression, using Padé approximation.

For this purpose we have to calculate the moments, which are:

$$M[0] = -2308.771329$$

$$M[1] = 0.0005862885330$$

$$M[2] = -1.221430006\,10^{-10}$$

$$M[3] = 1.214077215\,10^{-17}$$

Creating system given by expression (1.16) we have:

$$\begin{bmatrix} -2308.771329 & 0.0005862885330 \\ 0.0005862885330 & -1.221430006\,10^{-10} \end{bmatrix} \begin{bmatrix} b_2 \\ b_1 \end{bmatrix} = - \begin{bmatrix} 1.221430006\,10^{-10} \\ -1.214077215\,10^{-17} \end{bmatrix}$$

Solving this system, we obtain *b* coefficients:

$$b_1 = 7.059445983\,10^{-7}$$

$$b_2 = 1.263634118\,10^{-13}$$

Which can be used along with appropriated values of moments to calculate *a* coefficients:

$$a_0 = -2308.771329$$

$$a_1 = -0.001043576115$$

Then using the *a* and *b* coefficients, the order reduced system is:

$$H(s) = \frac{-0.001043576115\,s - 2308.771329}{1.263634118\,10^{-13}\,s^2 + 7.059445983\,10^{-7}\,s + 1} \quad (6.4)$$

And making a comparison between response given by a numerical simulator for this amplifier and that obtained from last expression, the plot showed in figure 4.19 is found. The phase response comparison for this approach can be seen in figure 4.20.

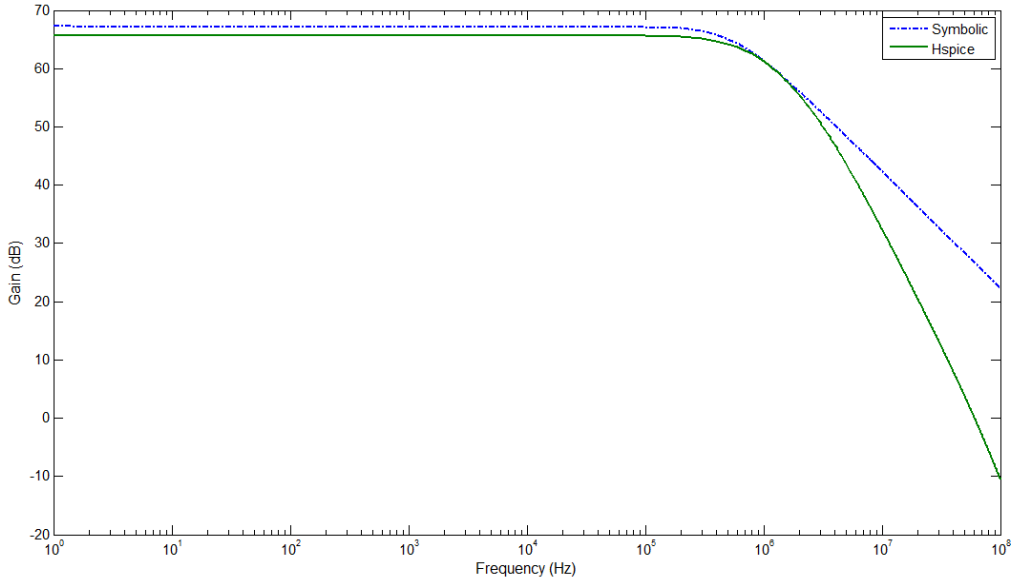


Figure 4.19 AWE approach for uncompensated amplifier

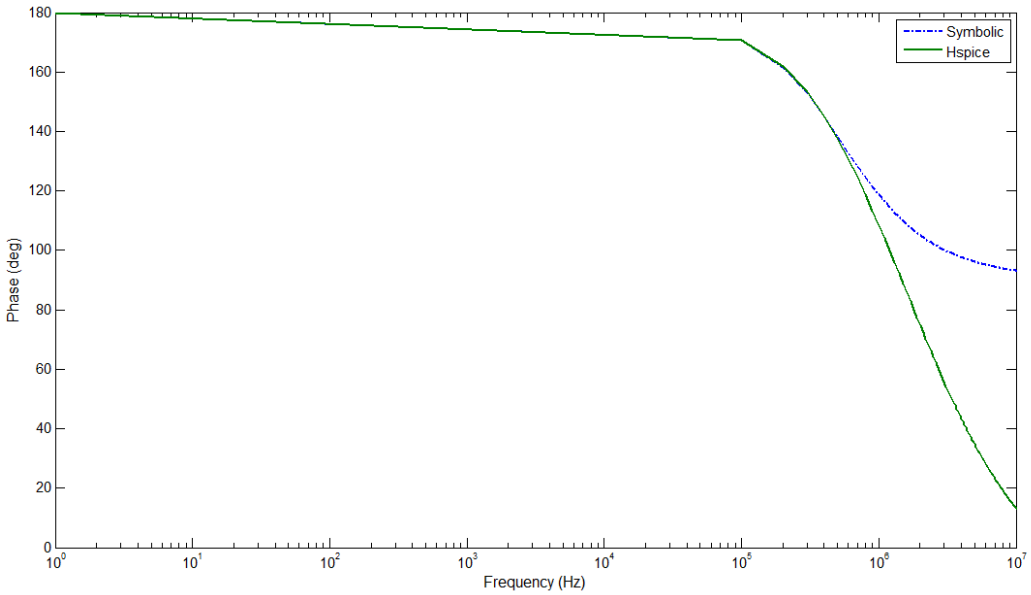


Figure 4.20 AWE approach for uncompensated amplifier (phase response)

4.8 Conclusion

Several examples of amplifiers where presented along this chapter, used to illustrate the method of nodal analysis formulation, and also the way to obtain reduced expressions through simplifications.

According to the graphs, it can be seen that there is an agreement in the results obtained when using a numerical simulator (Hspice) and when employing the proposed formulation to obtain symbolic expressions.

Asymptotic Waveform Evaluation represents a good method when reducing order of expressions. However in symbolic analysis it can be used only when dealing with semi-symbolic expressions, because the process of obtaining moments makes growing the length of expressions.

Chapter 5. Symbolic noise analysis

5.1 Noise analysis in amplifiers

It is important to perform noise analysis in circuits, because noise can determine the amplitudes of signals a circuit is able to manage. Noise is also related to parameters like power dissipation, speed and linearity [20].

There are different noise models used in circuit simulators, among those, BSIM models are often taken as a reference. In Spice2, for example, there are different models for flicker and thermal noise, according to the level. In table 5.1 these models are presented [21].

As it was seen in previous chapters, nullor is useful when modeling MOS transistor, in figure 5.1 a model of a transistor is shown, where sources of flicker and thermal noise are added to this device, this model can be used to obtain noise characteristics of circuits containing MOS transistors [9].

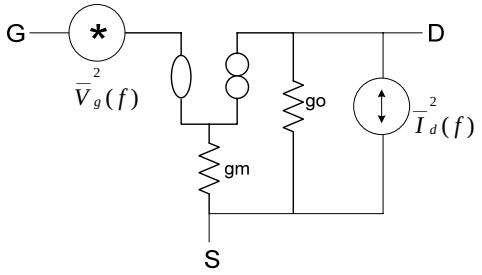


Figure 5.1 MOS transistor with associated noise sources

Hspice Models	Flicker Noise	Thermal Noise
NLEV = 0	$S_{ID} = \frac{K_F I_{IDS}^{AF}}{C_{ox} L_{eff}^2 f}$	$S_{channel} = \frac{8kTg_m}{3}$ (Also for NLEV = 2)
NLEV = 1	$S_{ID} = \frac{K_F I_{IDS}^{AF}}{C_{ox} W_{eff} L_{eff} f}$	
NLEV = 2,3	$S_{ID} = \frac{K_F g_m^2}{C_{ox} W_{eff} L_{eff} f^{AF}}$	$S_{channel} = \frac{8kT}{3} B(v_{gs} - v_{TH}) \frac{1 + a + a^2}{1 + a} GDSNOI$ $a = 1 - \frac{v_{dr}}{v_{dsat}}$ Linear $a = 0$ Saturation

Table 5.1 Spice models for noise [22]

Using model of figure 5.1, noise analysis will be performed on amplifiers through this chapter, obtaining output noise of these circuits. The method employed in the previous chapters will be used to obtain the expressions.

5.2 Common source amplifier

Let's start analysis with common source amplifier with resistive load, whose transistor representation appears on figure 4.1, and the nullor equivalent using model for noise analysis is seen in figure 5.2

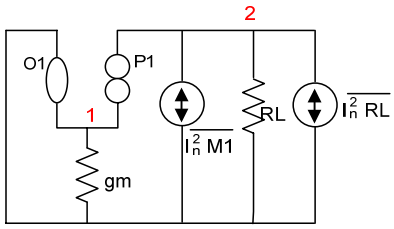


Figure 5.2 Equivalent for common source amplifier

In this case, the noise source for the transistor ($\overline{I_n^2 M1}$) can be represented by two components, one used for thermal and the second for the flicker noise, meanwhile, the resistor includes a source to represent the thermal noise on it.

Sets ROW and COL corresponding to this circuit are:
ROW={(1,2)}
COL ={(2)}

And the table for admittances in this circuit is:

Table A

Nodes	Admittances
1	gm
2	gL

The equation that represents the output noise voltage can be found using ROW, and COL sets and admittance table as:

$[gL]^2[V_{1,2}] = [\overline{I_n^2 M1} + \overline{I_n^2 RL}]$

Then, for this circuit, the output noise voltage per unit bandwidth is:

$$\overline{V_n^2, out} = \left(4kT \frac{2}{3} gm + \frac{K}{CoxWL} \cdot \frac{1}{f} \cdot g_m^2 + \frac{4kT}{R_L} \right) R_L^2 \qquad (5.1)$$

Simulating the response of this circuit in Hspice, we get the graph of figure 5.3, where it can also be observed the response given by plotting the symbolic expression (5.1)

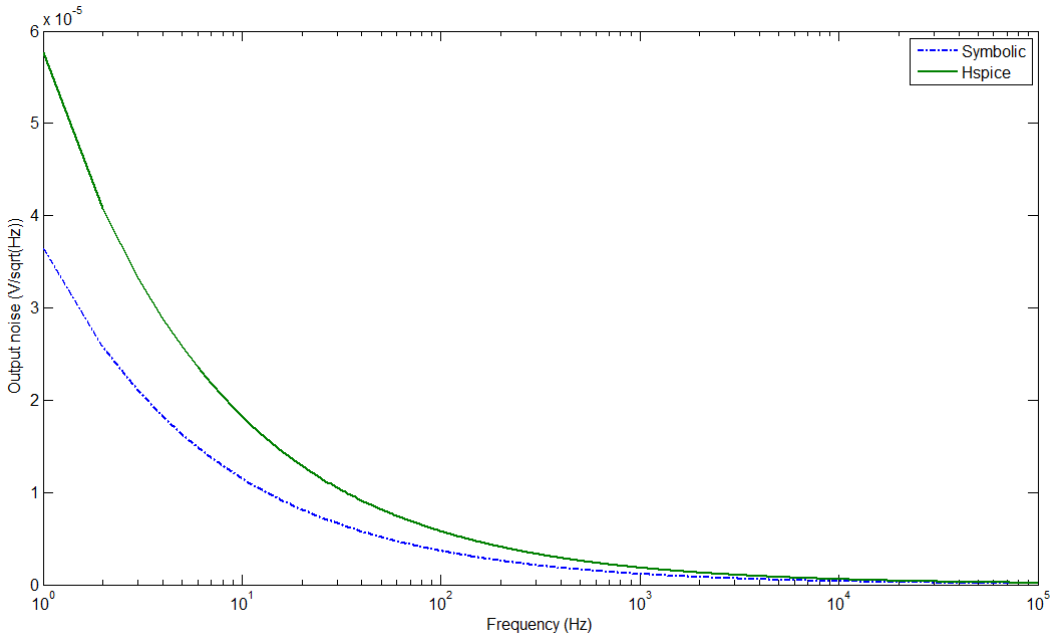


Figure 5.3 Numerical and symbolic expression responses (common source amplifier with resistive load)

5.3 Common source amplifier with active load

Now, applying analysis to an amplifier with a transistor as a load (active load), which is shown at transistor level on figure 4.5. After substituting the transistors by its equivalent with noise sources, circuit of figure 5.4 is obtained.

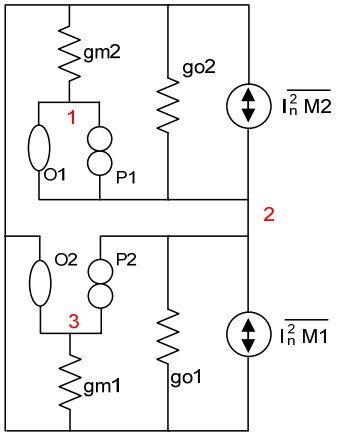


Figure 5.4 Equivalent circuit for common source amplifier, using model of transistor for noise

ROW and COL sets should be obtained for this circuit, and these are:

ROW: {(1,2,3)}

COL: {(1,2)}

and the table for admittances is:

Table A	
Nodes	Admittances
1	gm2
2	go2+go1
3	gm1

Filling the admittance matrix and the vectors for voltages and currents, the system corresponding to this inverting amplifier is:

$[\overline{I_n^2 M1} + \overline{I_n^2 M2}]$=[gm2+go2+go1][$V_{1,2}$]

By solving this system, the noise at output can be expressed as:

$V_{1,2} = \overline{V_n^2, out} = ([\overline{I_n^2 M1} + \overline{I_n^2 M2}])/((gm_2)^2 + (go_2)^2 + (go1)^2)$ (5.2)

The graphs comparing the response between the symbolic obtained expression and Hspice simulation can be seen in the figure 5.5.

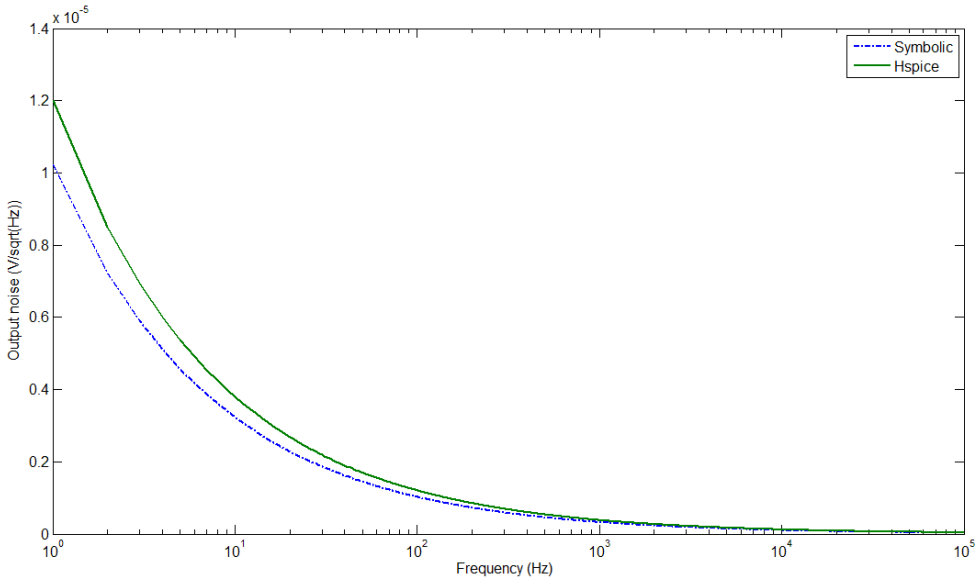


Figure 5.5 Numerical and symbolic expression responses (common source amplifier with active load)

5.4 Non-inverting amplifier

Now a non inverting amplifier is presented. This circuit is shown in figure 2.12. The equivalent circuit using nullors is depicted in figure 5.6.

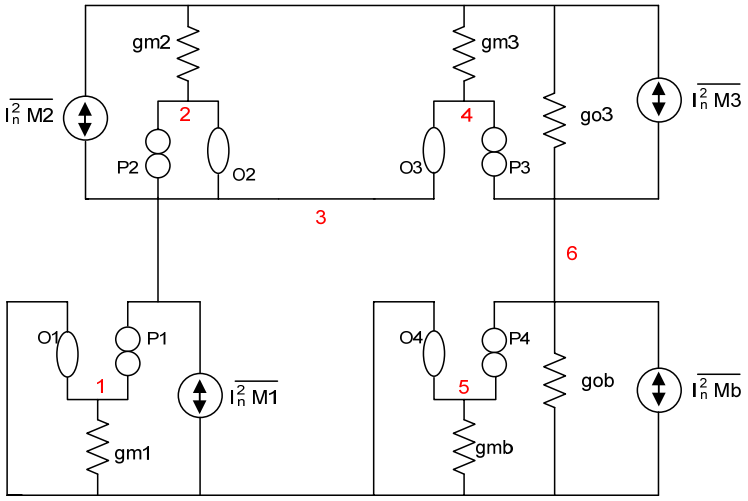


Figure 5.6 Equivalent circuit for non inverter amplifier using noise sources

Finding ROW and COL sets for this circuit we get:

ROW = {(1,2,3),(4,5,6)}

COL={(2,3,4),(6)}

and the table for admittances is:

Table A

Nodes	Admittances
1	gm1
2	gm2
3	0
4	gm3
5	gmb
6	go3+gob

So the system of equations representing this circuit is:

$$\begin{bmatrix} gm2^2 & 0 \\ gm3^2 & go3^2 + gob^2 \end{bmatrix} \begin{bmatrix} \overline{V^2_{n\,1,2,3}} \\ \overline{V^2_{n\,4,5,6}} \end{bmatrix} = \begin{bmatrix} \overline{I^2_n M_1} + \overline{I^2_n M_2} \\ \overline{I^2_n M_3} + \overline{I^2_n M_b} \end{bmatrix}$$

And solving for voltage at output $\overline{V^2_{n\,4,5,6}}$, the response is given as:

$$\overline{V^2_{n,\,out}} = \frac{-gm3^2\,\overline{I^2_n M_1} - gm3^2\,\overline{I^2_n M_2} + gm2^2\,\overline{I^2_n M_b} + gm2^2\,\overline{I^2_n M_3}}{gm2^2\left(go3^2 + gob^2\right)} \qquad (5.3)$$

Using numerical values for this expression in order a comparison with an Hspice simulation of the circuit can be done, the graph of figure 5.7 is obtained.

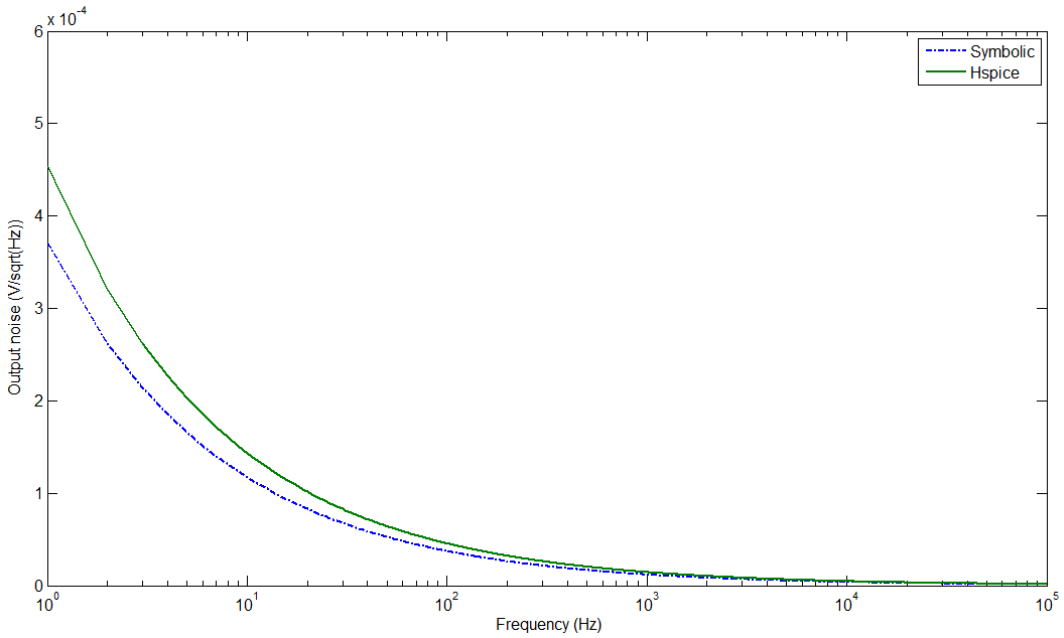


Figure 5.7 Numerical and symbolic expression responses (non-inverting amplifier)

5.5 Differential pair

The circuit representing differential pair can be seen in the figure 4.7. Substituting this circuit by its equivalent using noise sources of MOS transistor, the circuit of figure 5.8 is obtained.

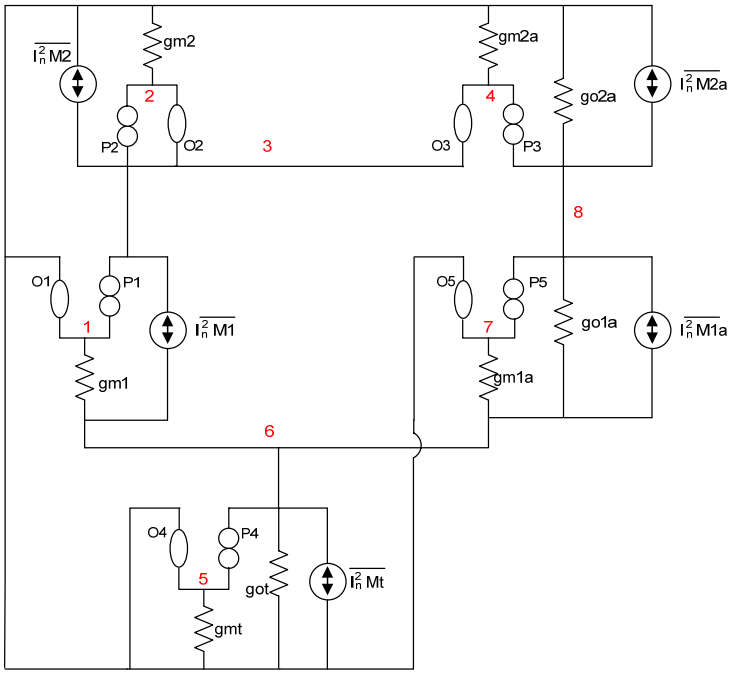


Figure 5.8. Equivalent circuit for differential pair using noise sources

ROW and COL sets are for this circuit are given by:

ROW = {(1,2,3),(4,7,8),(5,6)}

COL={(2,3,4),(6),(8)}

And the admittances tables are:

Table A

Nodes	Admittances
1	gm1
2	gm2
3	0
4	gm2a
5	gmt
6	got+gm1+gm1a+go1a
7	gm1a
8	go1a+go2a

Table B

Floating admittances	Nodes
gm1	(1,6)
go1a	(6,8)
gm1a	(6,7)

The system corresponding to this circuit is now presented:

$$\begin{bmatrix} gm2^2 & gm1^2 & 0 \\ gm2a^2 & go1a^2 + gm1a^2 & go1a^2 + go2a^2 \\ 0 & got^2 + gm1^2 + gm1a^2 + go1a^2 & go1a^2 \end{bmatrix} \begin{bmatrix} \overline{V^2}_{n,2,3,4} \\ \overline{V^2}_{n,6} \\ \overline{V^2}_{n,8} \end{bmatrix} = \begin{bmatrix} \overline{I^2}_{n,M1} + \overline{I^2}_{n,M2} \\ \overline{I^2}_{n,M1} + \overline{I^2}_{n,M1a} + \overline{I^2}_{n,Mt} \\ \overline{I^2}_{n,M1a} + \overline{I^2}_{n,M2a} \end{bmatrix}$$

And the noise at output can be expressed by the next equation:

$$\overline{V^2}_{n,out} = (-gm2^2 go1a^2 \overline{I^2}_{n,M2a} - go1a^2 gm2a^2 \overline{I^2}_{n,M2} + gm1a^2 gm2^2 \overline{I^2}_{n,M1} - gm1a^2 gm2a^2 \overline{I^2}_{n,M1} - gm1a^2 gm2a^2 \overline{I^2}_{n,M2} + gm1a^2 gm2^2 \overline{I^2}_{n,Mt}$$
$$+ go1a^2 gm2^2 \overline{I^2}_{n,M1} + go1a^2 gm2^2 \overline{I^2}_{n,Mt} - go1a^2 gm2a^2 \overline{I^2}_{n,M1} + gm2a^2 gm1^2 \overline{I^2}_{n,M1a} + gm2^2 got^2 \overline{I^2}_{n,M1a} - gm1a^2 gm2^2 \overline{I^2}_{n,M2a}$$
$$+ gm2a^2 gm1^2 \overline{I^2}_{n,M2a} + gm2^2 gm1^2 \overline{I^2}_{n,M1a} + got^2 gm2^2 \overline{I^2}_{n,M1a} + got^2 gm2^2 \overline{I^2}_{n,Mt} - got^2 gm2a^2 \overline{I^2}_{n,M1} - got^2 gm2a^2 \overline{I^2}_{n,M2} + gm1^2 gm2^2 \overline{I^2}_{n,M1}$$
$$- gm1^2 gm2a^2 \overline{I^2}_{n,M1} - gm1^2 gm2a^2 \overline{I^2}_{n,M2} + gm1^2 gm2^2 \overline{I^2}_{n,Mt}) / (go1a^2 gm2a^2 gm1^2 + gm2^2 go1a^2 got^2 + gm2^2 go1a^2 gm1^2 + gm2^2 go2a^2 got^2$$
$$+ gm2^2 go2a^2 gm1^2 + gm2^2 go2a^2 gm1a^2 + gm2^2 go2a^2 go1a^2) \qquad (5.4)$$

Comparing the response obtained with numerical values for expression 5.4 and the simulation performed in Hspice, the graphs of figure 5.9 can be observed:

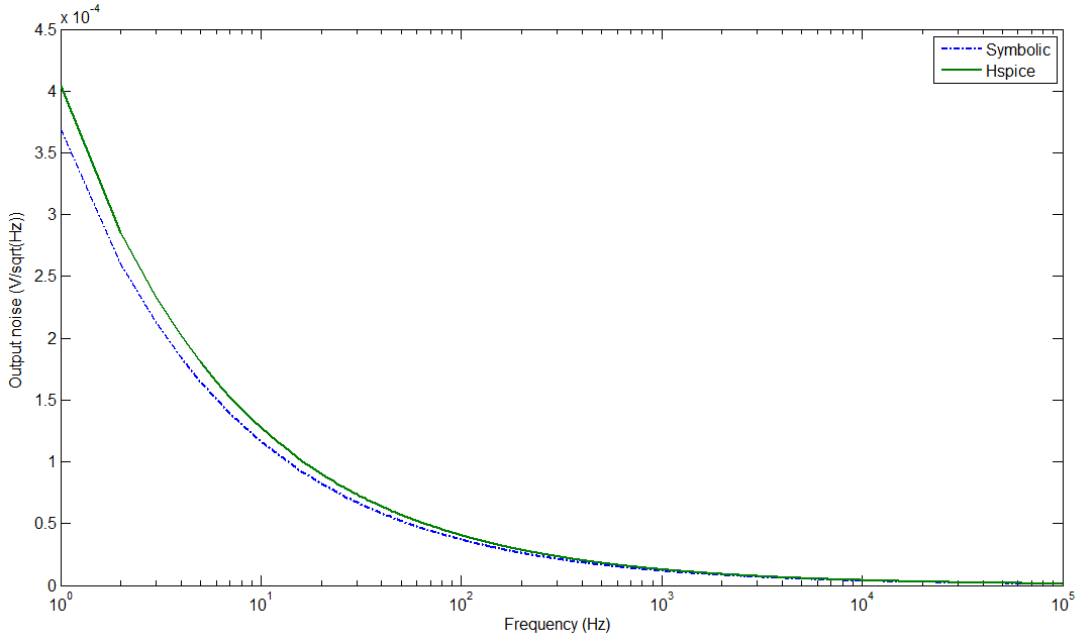


Figure 5.9 Numerical and symbolic expression responses (differential pair amplifier)

5.6 Miller amplifier

Miller amplifier is represented in figure 3.1. Meanwhile, the circuit equivalent with the noise sources associated to it can be seen in the figure 5.10.

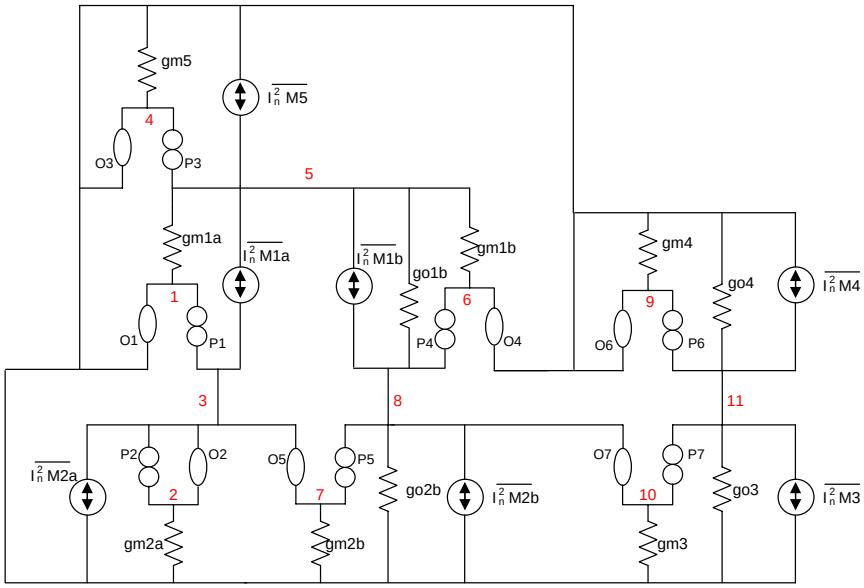


Figure 5.10 Equivalent circuit for Miller amplifier

ROW and COL sets are given by:

ROW = {(1,2,3),(4,5),(6,7,8),(9,10,11)}

COL={(2,3,7),(5),(8,10),(11)}

And the admittances tables are:

Table A

Nodes	Admittances
1	gm1a
2	gm2a
3	0
4	gm5
5	gm1b+go1b
6	gm1b
7	gm2b
8	go1b+go2b
9	gm4
10	gm3
11	go3+go4

Table B

Floating admittances	Nodes
gm1a	(1,5)
gm1b	(5,6)
go1b	(5,8)

The nodal analysis formulation corresponding to this circuit is:

$$\begin{bmatrix} gm2a^2 & gm1a^2 & 0 & 0 \\ 0 & gm1b^2 & go1b^2 & 0 \\ gm2b^2 & gm1b^2 + go1b^2 & go2b^2 & 0 \\ 0 & 0 & gm3^2 & go3^2 + go4^2 \end{bmatrix} \begin{bmatrix} \overline{V^2}_{n,2,3,7} \\ \overline{V^2}_{n,5} \\ \overline{V^2}_{n,8,10} \\ \overline{V^2}_{n,11} \end{bmatrix} = \begin{bmatrix} \overline{I^2}_{n,M1a} + \overline{I^2}_{n,M2a} \\ \overline{I^2}_{n,M5} + \overline{I^2}_{n,M1a} + \overline{I^2}_{n,M1b} \\ \overline{I^2}_{n,M1b} + \overline{I^2}_{n,M2b} \\ \overline{I^2}_{n,M3} + \overline{I^2}_{n,M4} \end{bmatrix}$$

And solving this system of equations to obtain the output noise voltage, the resulting expression is:

$$\overline{V^2}_{n,out} = -(gm1b^2 gm3^2 gm2a^2 \overline{I^2}_{n,M2b} - gm1b^2 gm3^2 gm2b^2 \overline{I^2}_{n,M1a} - gm1b^2 gm3^2 gm2b^2 \overline{I^2}_{n,M2a} - gm1b^2 gm2a^2 go2b^2 \overline{I^2}_{n,M4} - gm2a^2 gm1b^2 gm3^2 \overline{I^2}_{n,M5}$$
$$- gm2a^2 gm1b^2 gm3^2 \overline{I^2}_{n,M1a} - gm1b^2 gm2a^2 go2b^2 \overline{I^2}_{n,M3} + gm2a^2 gm1b^2 go1b^2 \overline{I^2}_{n,M4} + gm2a^2 gm1b^2 go1b^2 \overline{I^2}_{n,M3} - gm2a^2 go1b^2 gm3^2 \overline{I^2}_{n,M5}$$
$$- gm2a^2 go1b^2 gm3^2 \overline{I^2}_{n,M1b} - gm2a^2 go1b^2 gm3^2 \overline{I^2}_{n,M1a} + gm2a^2 go1b^4 \overline{I^2}_{n,M3} + gm2b^2 gm1a^2 gm3^2 \overline{I^2}_{n,M1b} + gm2b^2 gm1a^2 gm3^2 \overline{I^2}_{n,M1a}$$
$$- gm2b^2 gm1a^2 go1b^2 \overline{I^2}_{n,M4} + gm2b^2 gm1a^2 gm3^2 \overline{I^2}_{n,M5} - gm2b^2 gm1a^2 go1b^2 \overline{I^2}_{n,M3} + gm2a^2 go1b^4 \overline{I^2}_{n,M4}) / (gm1b^2 gm2a^2 go2b^2 go3^2$$
$$+ gm1b^2 gm2a^2 go2b^2 go4^2 - gm2a^2 gm1b^2 go1b^2 go3^2 - gm2a^2 gm1b^2 go1b^2 go4^2 - gm2a^2 go1b^4 go4^2 - gm2a^2 go1b^4 go3^2 + gm2b^2 gm1a^2 go1b^2 go3^2 +$$
$$gm2b^2 gm1a^2 go1b^2 go4^2)$$

(5.5)

The graph comparing the response given by Hspice and the symbolic expression can be seen in the figure 5.11.

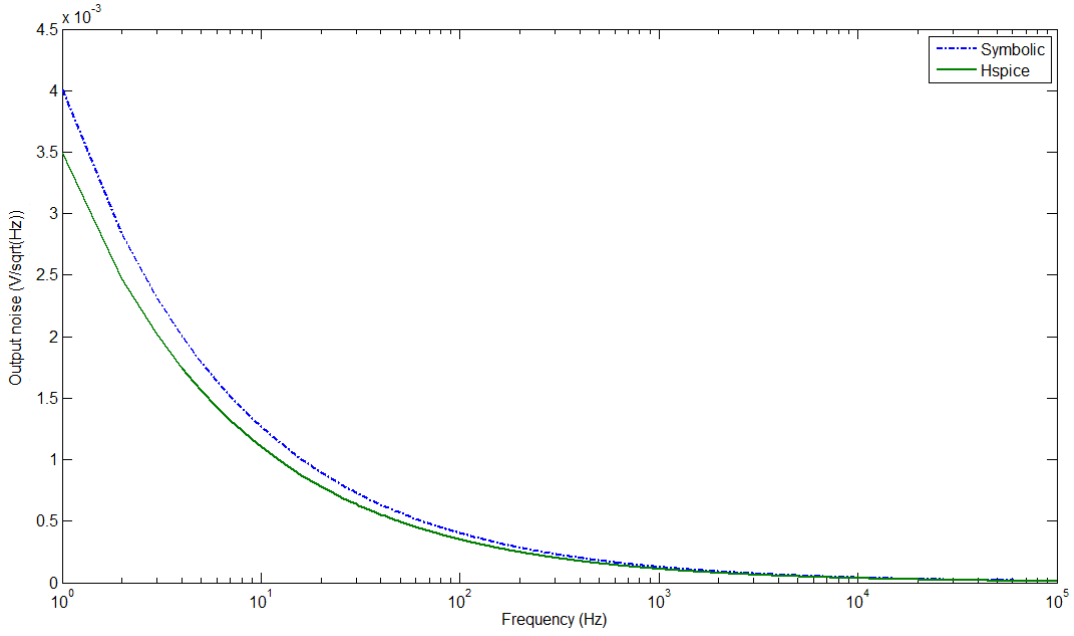


Figure 5.11 Numerical and symbolic expression responses (Miller amplifier)

5.7 Uncompensated amplifier

The circuit corresponding to this amplifier can be observed in figure 4.10, the equivalent circuit with nullors and the associated noise sources appears in figure 5.12.

ROW and COL sets are given by:
ROW = {(1,2,3),(4,5,8),(6,7),(9,10,11),(13,14)}
COL={(2,3,4),(6),(8,9),(11,12),(14)}

And the admittances tables are:

Table A

Nodes	Admittances
1	gm1
2	0
3	gm3
4	gm4
5	gm2
6	gm1+gm2+go2
7	gm5
8	go2+go4
9	gm6
10	gm7
11	go6+go7
12	gm8
13	gm9
14	go8+go9

Table B

Floating admittances	Nodes
gm1	(1,6)
gm2	(5,6)
go2	(6,8)
gm8	(12,14)

The system of equations corresponding to this circuit is:

$$\begin{bmatrix} gm3^2 & gm1^2 & 0 & 0 & 0 \\ gm4^2 & gm2^2+go2^2 & go2^2+go4^2 & 0 & 0 \\ 0 & gm1^2+gm2^2+go2^2 & go2^2 & 0 & 0 \\ 0 & 0 & gm6^2 & go6^2+go7^2 & 0 \\ 0 & 0 & 0 & gm8^2 & go8^2+go9^2 \end{bmatrix} \begin{bmatrix} \overline{V^2}_{n,2,3,4} \\ \overline{V^2}_{n,6} \\ \overline{V^2}_{n,8,9} \\ \overline{V^2}_{n,11,12} \\ \overline{V^2}_{n,14} \end{bmatrix} = \begin{bmatrix} \overline{I^2}_{n,M1}+\overline{I^2}_{n,M3} \\ \overline{I^2}_{n,M1}+\overline{I^2}_{n,M2}+\overline{I^2}_{n,M5} \\ \overline{I^2}_{n,M2}+\overline{I^2}_{n,M4} \\ \overline{I^2}_{n,M6}+\overline{I^2}_{n,M7} \\ \overline{I^2}_{n,M8}+\overline{I^2}_{n,M9} \end{bmatrix}$$

By solving this system of equations for output voltage, the expression is:

$$\overline{V^2}_{n,out} = (-gm4^2\,gm1^2\,gm8^2\,go2^2\,\overline{I^2}_{n,M7} + gm1^2\,gm8^2\,gm6^2\,gm3^2\,\overline{I^2}_{n,M1} + gm1^2\,gm3^2\,go4^2\,go6^2\,\overline{I^2}_{n,M9} - gm2^2\,gm8^2\,gm3^2\,go4^2\,\overline{I^2}_{n,M6}$$
$$+ go2^2\,gm3^2\,go4^2\,go6^2\,\overline{I^2}_{n,M9} + gm2^2\,gm3^2\,go4^2\,go7^2\,\overline{I^2}_{n,M8} - go2^2\,gm8^2\,gm6^2\,gm4^2\,\overline{I^2}_{n,M3} + go2^2\,gm8^2\,gm6^2\,gm3^2\,\overline{I^2}_{n,M1} - gm2^2\,gm8^2\,gm6^2\,gm4^2\,\overline{I^2}_{n,M1}$$
$$+ go2^2\,gm8^2\,gm6^2\,gm3^2\,\overline{I^2}_{n,M5} - go2^2\,gm8^2\,gm6^2\,gm4^2\,\overline{I^2}_{n,M1} - go2^2\,gm8^2\,gm3^2\,go4^2\,\overline{I^2}_{n,M6} - go2^2\,gm8^2\,gm3^2\,go4^2\,\overline{I^2}_{n,M7} + gm2^2\,gm8^2\,gm6^2\,gm3^2\,\overline{I^2}_{n,M5}$$
$$+ gm2^2\,gm3^2\,go4^2\,go6^2\,\overline{I^2}_{n,M9} + gm2^2\,gm3^2\,go4^2\,go6^2\,\overline{I^2}_{n,M8} + gm2^2\,gm3^2\,go4^2\,go7^2\,\overline{I^2}_{n,M9} + go2^2\,gm3^2\,go4^2\,go7^2\,\overline{I^2}_{n,M8} + go2^2\,gm3^2\,go4^2\,go7^2\,\overline{I^2}_{n,M9}$$
$$+ go2^2\,gm3^2\,go4^2\,go6^2\,\overline{I^2}_{n,M8} - gm2^2\,gm8^2\,gm3^2\,go4^2\,\overline{I^2}_{n,M7} - gm3^2\,gm2^2\,gm8^2\,gm6^2\,\overline{I^2}_{n,M4} - gm3^2\,go2^2\,gm8^2\,gm6^2\,\overline{I^2}_{n,M4} + gm4^2\,gm1^2\,go2^2\,go7^2\,\overline{I^2}_{n,M8}$$
$$- gm4^2\,gm1^2\,gm8^2\,go2^2\,\overline{I^2}_{n,M6} + gm4^2\,gm1^2\,gm8^2\,gm6^2\,\overline{I^2}_{n,M4} + gm4^2\,gm1^2\,gm8^2\,gm6^2\,\overline{I^2}_{n,M2} + gm4^2\,gm1^2\,go2^2\,go6^2\,\overline{I^2}_{n,M8} + gm4^2\,gm1^2\,go2^2\,go6^2\,\overline{I^2}_{n,M9}$$
$$+ gm1^2\,gm8^2\,gm6^2\,gm3^2\,\overline{I^2}_{n,M2} + gm4^2\,gm1^2\,go2^2\,go7^2\,\overline{I^2}_{n,M9} + gm1^2\,gm8^2\,gm6^2\,gm3^2\,\overline{I^2}_{n,M5} - gm1^2\,gm8^2\,gm6^2\,gm4^2\,\overline{I^2}_{n,M3} - gm1^2\,gm8^2\,gm6^2\,gm4^2\,\overline{I^2}_{n,M1}$$
$$- gm1^2\,gm8^2\,gm3^2\,go2^2\,\overline{I^2}_{n,M7} - gm1^2\,gm8^2\,gm3^2\,go2^2\,\overline{I^2}_{n,M6} - gm1^2\,gm8^2\,gm3^2\,go4^2\,\overline{I^2}_{n,M7} - gm1^2\,gm8^2\,gm3^2\,go4^2\,\overline{I^2}_{n,M6} + gm1^2\,gm3^2\,go2^2\,go6^2\,\overline{I^2}_{n,M9}$$
$$+ gm1^2\,gm3^2\,go2^2\,go6^2\,\overline{I^2}_{n,M8} + gm1^2\,gm3^2\,go4^2\,go7^2\,\overline{I^2}_{n,M9} + gm1^2\,gm3^2\,go4^2\,go6^2\,\overline{I^2}_{n,M8} + gm1^2\,gm3^2\,go2^2\,go7^2\,\overline{I^2}_{n,M8} + gm1^2\,gm3^2\,go2^2\,go7^2\,\overline{I^2}_{n,M9}$$
$$+ gm1^2\,gm3^2\,go4^2\,go7^2\,\overline{I^2}_{n,M8} - gm2^2\,gm8^2\,gm6^2\,gm4^2\,\overline{I^2}_{n,M3} + gm2^2\,gm8^2\,gm6^2\,gm3^2\,\overline{I^2}_{n,M1}) / (gm4^2\,gm1^2\,go2^2\,go6^2\,go8^2 + gm4^2\,gm1^2\,go2^2\,go7^2\,go8^2$$
$$+ gm2^2\,gm3^2\,go4^2\,go6^2\,go9^2 + gm1^2\,gm3^2\,go2^2\,go6^2\,go8^2 + gm2^2\,gm3^2\,go4^2\,go7^2\,go8^2 + go2^2\,gm3^2\,go4^2\,go6^2\,go8^2$$
$$+ gm4^2\,gm1^2\,go2^2\,go7^2\,go9^2 + gm1^2\,gm3^2\,go4^2\,go6^2\,go9^2 + gm1^2\,gm3^2\,go4^2\,go6^2\,go8^2 + gm4^2\,gm1^2\,go2^2\,go6^2\,go9^2 + gm1^2\,gm3^2\,go4^2\,go7^2\,go8^2$$
$$+ gm1^2\,gm3^2\,go4^2\,go7^2\,go9^2 + go2^2\,gm3^2\,go4^2\,go6^2\,go9^2 + gm2^2\,gm3^2\,go4^2\,go6^2\,go8^2 + gm1^2\,gm3^2\,go2^2\,go6^2\,go9^2 + go2^2\,gm3^2\,go4^2\,go7^2\,go8^2$$
$$+ go2^2\,gm3^2\,go4^2\,go7^2\,go9^2 + gm1^2\,gm3^2\,go2^2\,go7^2\,go8^2 + gm1^2\,gm3^2\,go2^2\,go7^2\,go9^2$$

(5.6)

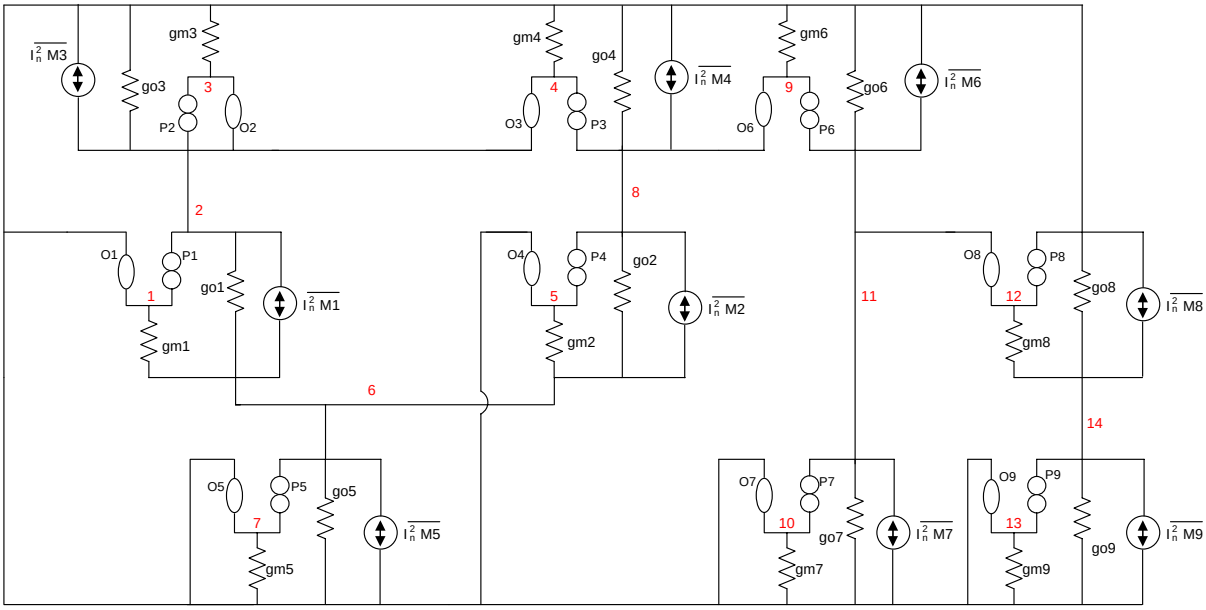


Figure 5.12 Equivalent circuit for uncompensated amplifier using noise sources

The graph in figure 5.13 corresponds to the response given in Hspice and the plot corresponding to expression 5.6.

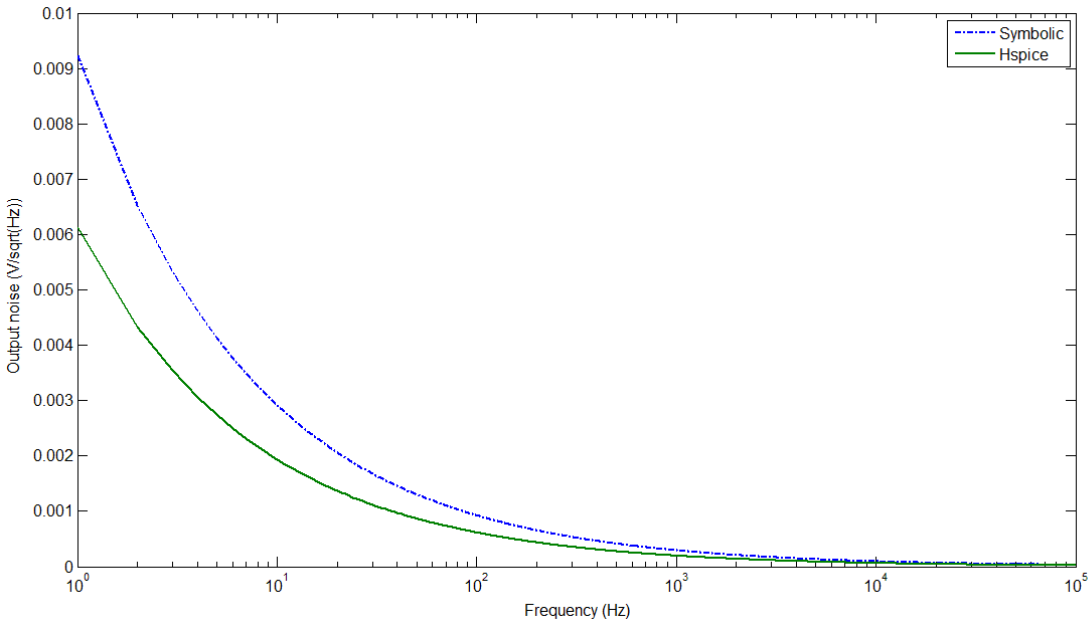


Figure 5.13 Numerical and symbolic expression responses (uncompensated amplifier)

5.8 Conclusion

There exist different models used to represent noise in circuits, in this chapter; a model based on nullors to represent noise in MOS transistor was presented.

The noise-based model for MOS transistor was substituted in amplifier circuits, so a noise at output expression could be found.

It was observed that the plots are in a good agreement with the graph obtained with a numerical simulation. So it can be said that the symbolic expressions representing noise at output of the circuit are correct.

General conclusions

In the first chapter there were presented different methods that have been used to obtain symbolic expressions of circuits, those methods include graph, flowchart and analytical methods. In that chapter it was also introduced the topic concerning to asymptotic waveform evaluation, which allows, with the use of moments reducing the order of expressions.

In first chapter, there was a briefly introduction to nullor concept, but it was in the second chapter where this element was considered to model different active devices, such the opamp, ota or current conveyors, but most importantly the MOS transistor, which can have different representations according to the necessary application.

It was in second chapter, where a methodology to perform nodal analysis was presented, allowing the substitution of MOS transistors by the nullor equivalents and then performing pure nodal analysis, taking advantage of the reduction of order of matrices due to the properties of nullors. A circuit was presented in this chapter to illustrate the way an amplifier can be treated by using the mentioned methodology.

In the third chapter, some techniques were introduced, which can be used to perform simplification of symbolic expressions before they are created, during the process of formulation or after it. Most of these techniques are based on heuristics, which are the result of knowledge on circuit behavior under different conditions, such biasing, position of transistor in the circuit or region of operation of MOS transistor. Making use of such techniques allows the reduction of symbolic expressions and then obtaining expressions easier to be handled, as it was shown with the use of a circuit as an illustrative example in this chapter. With the use of a numerical simulator and the agreement presented with the expressions obtained through the given methodology, it could be appreciated that the result obtained through symbolic analysis is correct.

In the fourth chapter, several examples of amplifiers were presented, varying from a small circuit to a larger one, where it could be appreciated the usefulness of neglecting some elements to create a reduced expression having the result an acceptable behavior compared to the response given by a numerical simulator.

In this chapter, two examples using Padé approach were analyzed, it was seen the usefulness of this technique when dealing only with a symbolic variable, because if this methodology is employed having a semi-symbolic or fully symbolic expression, it leads to an exponential grow of symbolic terms making the result very difficult to be interpreted or handle.

In last chapter of this Thesis, the already given methodology was employed to obtain the output noise of the amplifiers being analyzed throughout this work. The obtained result was possible due to the employment of an adequate model of MOS transistor with the appropriated noise sources attached to it.

Publications

E. Tlelo-Cuautle, E. Martínez-Romero, C. Sánchez-López, Sheldon X.-D. Tan
Symbolic Formulation Method for Mixed-Mode Analog Circuits using Nullors
IEEE, ICECS, 2009, pp. 856-859

E. Tlelo-Cuautle, C. Sánchez-López, E. Martínez-Romero, Sheldon X.-D. Tan
Symbolic analysis of analog circuits containing voltage mirrors and current mirrors
Analog Integrated Circuit and Signal Processing, Volume: 65, Issue: 1, pp.: 89-95, 2010

E. Tlelo-Cuautle, E. Martínez-Romero, C. Sánchez-López, Sheldon X.D.-Tan
Symbolic Behavioral Modeling of Low Voltage Amplifiers
7th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE 2010), pp.: 510-514, Chiapas, Mexico, 2010

E. Martínez-Romero, E. Tlelo-Cuautle, C. Sánchez-López, S. X.-D. Tan (2010).
Symbolic noise analysis of low voltage amplifiers by using nullors, SM2ACD, IEEE, Tunisia, 2010

References

[1] F.V. Fernández, A. Rodríguez-Vázquez, J.L. Huertas, G. Gielen, Symbolic analysis techniques: applications to analog design automation, IEEE Press, Piscataway, NJ, 1998.

[2] H. Floberg, Symbolic analysis in analog integrated circuit design, Kluwer Academic Publishers, USA, 1997

[3] Águila M., Jorge, Análisis simbólico de circuitos integrados analógicos, Tonantzintla, Puebla, 2004 (Thesis).

[4] C. Sánchez-López, F.V. Fernández and E. Tlelo-Cuautle, Generalized admittance matrix models of OTRAs and COAs, Microelectronics Journal, 2010

[5] Tlelo-Cuautle, E., Sanchez-Lopez, C., Moro-Frias, D. Symbolic analysis of (MO)(I)CCI(II)(III)-based analog circuits. International Journal of Circuit Theory and Applications, pp. : 649-659, 2009

[6] P.M. Lin, Symbolic Network Analysis. Elsevier, 1991

[7] S. X.-D. Tan, L. He, Advanced Model Order Reduction Techniques in VLSI Design, Cambridge University Press, 2007

[8] R. A. Rutenbar, G.E. Gielen, B.A. Antao, Computer-Aided Design of Analog Integrated Circuits and Systems, IEEE Press, Piscataway, NJ, 2002.

[9] Sánchez, López Carlos. Development of a symbolic method for noise analysis in analog circuits, Tonantzintla, Puebla, 2006 (Thesis).

[10] Sedra, A.S., Smith K.C., Circuitos microelectrónicos, Oxford University Press.

[11] E. Tlelo-Cuautle, C. Sánchez-López, E. Martínez-Romero, S. X. –D. Tan, Symbolic análisis of analog circuits containing voltage mirrors and current mirrors, Analog Integrated Circuits and Signal Processing, 2010. Volume: 65, Issue: 1, pp.: 89-95, 2010

[12] M. Fakhfakh, E. Tlelo-Cuautle, F.V. Fernández, Design of Analog Circuits through Symbolic Analysis, Bentham, Sciences Publishers Ltd., 2010.

[13] C. Sánchez-López, D. Moro-Frías, E. Tlelo-Cuautle, Improving the formulation process of the system of equations of analog circuits, SM²ACD, pp. 102-106, Erfurt, Germany, October 7-8, 2008.

[14] R. Shi, S. X.-D. Tan, Canonical symbolic analysis of large analog circuits with determinant decision diagrams, IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems, vol. 19, no. 1, pp. 1-18, 2000.

[15] W. Verhaegen, G. Gielen, Efficient DDD-based symbolic analysis of linear analog circuits, IEEE Trans. on Circuits and Systems II, vol. 49, no. 7, pp. 474-487, 2002.

[16] S. X.-D. Tan, Symbolic analysis of analog circuits by boolean logic operations, IEEE Trans on CAS-II, vol. 53, no. 11, pp. 1313-1317, 2005.

[17] C. Sánchez-López, E. Tlelo-Cuautle, Novel SBG, SDG and SAG techniques for symbolic analysis of analog integrated circuits. SM²ACD, pp. 17-22, Erfurt, Germany, October 7-8, 2008.

[18] E. Sánchez-Sinencio, Special section on symbolic circuit analysis techniques, IEEE TCAS-II Analog and Digital Signal Processing, Vol 45, No. 10, October 1998.

[19] E. Sánchez-Sinencio, Low voltage amplifier design techniques, Tutorial in IEEE MWSCAS, Cancun, 2009.

[20] Razavi, B., Design of Analog CMOS Integrated Circuits, Mc Graw Hill, 2001

[21] http://www-device.eecs.berkeley.edu/~bsim3/get.html

[22] C. Sánchez-López, E. Tlelo-Cuautle. Symbolic Computation of NF of Transistor Circuits. IEICE transactions on fundamentals of Electronics, Communications and Computer Sciences, Vol.E87-A No.9, September 2004

[23] Q. Yu, C. Sechen, A unified approach to the approximate symbolic analysis of large analog integrated circuits. IEEE Transactions on Circuits and Systems, Part I, Vol. 43, No. 8, pp. 656-669, August 1996.

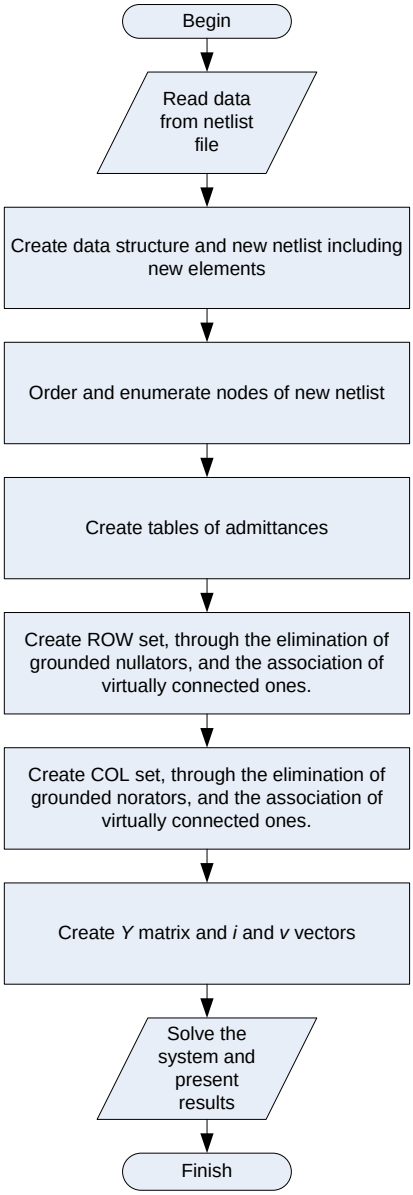
[24] P. Wambacq, G. Gielen, W. Sansen, F. Fernández. Approximation Turing expresión generation in symbolic análisis of analog ICs. Alta Frequenza-Rivista di Elettronica, Vol. 5, No. 6, pp 48-55, November-December 1993.

[25] P. Wambacq, F. Fernández, G. Gielen, W. Sansen, A. Rodríguez-Vázquez, Efficient symbolic computation of approximated small-signal characteristics. IEEE Journal on Solid-State Circuits, Vol. 30, No. 3, No. 3, pp. 327-330, March 1995.

Appendix

Algorithm implementation

To employ the method given in chapter two to create the system of equations representing nodal analysis $\dot{i} = Yv$, it can be done via a computational implementation, taken into account the steps already presented. To observe how this algorithm should be implemented, a flow chart is now depicted, in which the necessary steps to obtain the symbolic expressions are presented.



The algorithm is implemented using Maple, because it can deal directly with symbolic expressions.

As it was written down, for the first step of the algorithm, it is necessary to read data, in this case, the elements of the circuit are read from a netlist with .sp or .cir extension, which are files used for Spice simulations.

In second step of algorithm, a new netlist is created, where the nullor model of transistors and independent voltage sources are used, also, the dc voltage sources are grounded.

Parting from the new netlist, composed of nullors (norators, nullators) and any other passive device, each of these elements is set in a structure where the nodes of each of these elements, along with their value are listed. These structures will allow the correct management of elements, so they could be ordered in tables for admittances and in the case of nullors, they could be used to create ROW and COL sets.

For the third step of flow chart, once the data structures are ready, in the structure corresponding to admittances, it should be found which of the elements have a connection to datum node and any other node within the circuit, to create table A, where elements connected to each of the nodes of the circuit are enumerated.

Also in this step, elements connected between two nodes should be found, to create table B, corresponding to floating admittances.

For the next step, in the structure of nullators, it ought to be found which of these elements are connected to ground, in order they could be eliminated. Then a cycle should be implemented to look for nullators with a common connection, so they could be grouped to form COL set.

Almost the same should be done with norators, it should be found which of them are grounded connected to discard them, and those virtually connected should be together to create ROW set.

Once the ROW and COL sets have been created, perform a Cartesian product between these two sets, in order **Y** matrix can be formed. Also form **v** and **i** vectors, to have the $\mathbf{i} = \mathbf{Y}\mathbf{v}$ system complete.

Then, solve the system for the required parameter and present the results.

Software in Maple

To exemplify the software developed to obtain symbolic expressions, some fragments of Maple code will be showed in this section.

File reading and creation of data structure

```
#File reading
while line ≠ ".END" do
  line := readline(arch2);
  if line[1] = "R" then
    Resistor[il] := sscanf(line, "%s %d %d %f");
    il := (il) + 1;
  fi;
```

What we can see in this code, is how to read a file, using command readline, employing this command, each line within the file is read, and it is interrupted when the line reads a ".END" inside the file.

It can also be seen how to form the data structure that will allow the creation of the new netlist, it is done by creating a structure for each of the different elements of the original circuit, that is, resistors, capacitors, transistors, sources, etc. In the code presented, we see how resistor structure is created, that is, each element of the circuit is listed into a corresponding structure where similar elements are grouped, each having an index, the associated nodes and numerical values.

For the case of non NA compatible elements, they are substituted by their equivalents using nullors, and these new elements are used in the new netlist, including the addition of new nodes resulting of the substitution of nullor equivalents.

Also in the new netlist, elements that were connected originally to a DC voltage source are grounded, so it can be seen which of those can be neglected (in case of nullators and norators).

Once the new netlist has been created, the different nodes of it are sorted, because they will be used as indexes for ROW and COL sets and taken as a reference for admittance tables. In this case, a fragment of the code for sorting the nodes is presented.

```
while valor ≠ 1 do
valor := 1 :
 for i from 0 to maxnod − 2 do
   if nodo[i] > nodo[i + 1] then
     temp1 := nodo[i] :
     nodo[i] := nodo[i + 1] :
     nodo[i + 1] := temp1 :
     valor := 0 :
   fi:
 od:
od:
```

Now the tables for admittances should be created, and this is done by checking the node connections of each of these elements, to know whether they are connected to ground or between two nodes (floating admittances)

A piece of code where it is verified which admittances are connected to each node can here be seen.

```
for i from 1 to var − 1 do
 for j from 1 to o − 1 do
    if (elem[j][2] = node[i] or elem[j][3] = node[i]) then
      admit[i] := elemento[j][1] + admit[i];
    fi:
od:
od:
```

In the case of enumeration of floating admittances, the code is here presented.

```
for j from 1 to o − 1 do
  if (elem[j][2] ≠ 0 and elem[j][3] ≠ 0) then
    floated[cont][1] := elemento[j][1];
    floated[cont][2] := elem[j][2];
    floated[cont][3] := elem[j][3];
    cont := cont + 1;
 fi:
od:
```

For the creation of COL and ROW sets, it should be verified which nodes of nullators and norators have a terminal connected to ground, so they could be neglected, it also should be verified which of these elements are sharing a node, so they can be set together in sets corresponding to column and row of *i=Yv* system.

A fragment of the corresponding code can be seen now. First, the elimination of grounded nodes is presented.

```
for i from 1 to p − 1 do
  if (nullat[i][2] ≠ 0 and nullat[i][3] ≠ 0) then
    nnullat[j][1] := nullat[i][1] :
    nnullat[j][2] := nullat[i][2] :
    nnullat[j][3] := nullat[i][3] :
   j := j + 1 :
 fi:
od:
```

Now the union of nullators is presented, at least a part of the entire procedure to perform this action.

```
for ii from 1 to j − 1 do
 for jj from 1 to j − 1 do
if  (nnullat[ii][2] = nnullat[jj][2] or nnullat[ii][2] = nnullat[jj][3])then
  union1[k] := ({nnullat[jj][2], nnullat[jj][3]} union {nnullat[ii][2], nnullat[ii][3]});
  k := k + 1;
break;

 elif (nnullat[ii][3] = nnullat[jj][2] or nnullat[ii][3] = nnullat[jj][3]) then
  union1[k] := ({nnullat[jj][2], nnullat[jj][3]} union {nnullat[ii][2], nnullat[ii][3]});
  k := k + 1;
 break;
  fi;
  od:
  od:
```

The same kind of procedure is performed for norators.

Once sets ROW and COL have been created, it is now possible to do Cartesian product to fill Y matrix. Cartesian product is now presented.

```
for i from 1 to colrownum do
 for j from 1 to colrownum do
  A2 := { } :
       T := cartprod([rowu[i], colu[j]]) :
      while not T[finished] do A1 := T[nextvalue]( );
     A2 := A2 union {A1};
    end do:
 MAT[i][j] := A2;
 od:
 od:
```

To fill the admittance matrix, these procedures are followed:

```
for i from 1 to colrownum do
  for j from 1 to colrownum do
    for k from 1 to var − 1 do
      test := verify(par[k], MAT[i][j], `subset`);
    if test = true then
    mtrx[i][j] := grounded[k][2] + mtrx[i][j] :
      else
    mtrx[i][j] := mtrx[i][j] :
    fi:
    od:
   od:
 od:
for i from 1 to colrownum do
 for j from 1 to colrownum do
   for k from 1 to 2·maxfloat do
     test := verify(floating[k][1], MAT[i][j], `subset`);
   if test = true then
   mtrx[i][j] := floating[k][2] + mtrx[i][j] :
 fi:
 od:
 od:
od:
```

```
for i from 1 to colrownum do
  for j from 1 to colrownum do
  A[i, j] := mtrx[i][j];
 od:
od:
```

Then the vectors for current sources and voltages (variables) are created, which can be seen next.

```
for i from 1 to colrownum do
check := {Fuenteac[1][2]} subset rowu[i];
  if check = true then
  vectinput[i] := 1;
  fi:
 od:

for i from 1 to colrownum do
 B[i] := vectinput[i] :
od:
```

Finally, the solution for the system should be found, which is realized with this command.

```
solu := linsolve(A, B) :
```

Then the desired expression is found among the solution given by linsolve command results.

The same procedure as the one presented in this appendix should be followed to implement an algorithm to obtain output noise of circuits.