# A New Technique for the Efficient Solution of Singular Circuits

Jaijeet Roychowdhury

AT&T Bell Laboratories, Murray Hill, NJ

**Abstract**

*Singular circuits are those that have a continuum of solutions (or no solution) and are characterized by rank deficiency in the (linearized) circuit matrix. Such circuits often arise in practice - examples are filters with poles at zero, chains of transmission gates that are off, and circuits that rely on charge storage and transfer. In this paper, a technique for the efficient solution of such circuits is presented. The method is based on solving for the minimum-least-squares solution of the singular system. Unlike traditional methods for least squares solution, the new approach exploits the sparsity of the circuit matrix, making it practical for large industrial circuits. The method is best for applications with relatively small singular subspaces, as is the case in most circuits. Applications to industrial designs testify to the efficacy of the new technique; an example in which more than a week of design time would have been saved is presented.*

## 1 Introduction

A basic operation in circuit simulation, as in many other numerical computations, is that of solving a system of linear equations $Ax = b$ for the vector $x$ (with $A$ a given sparse matrix and $b$ a known vector). This operation arises, for example, in the Newton-Raphson method for nonlinear equation solution[1], as well as during the implicit integration of ordinary differential equations[2]. In most practical situations, the matrix $A$ is assumed to be of full rank or *nonsingular*, i.e., with a nonzero determinant. This assumption makes it possible to use efficient techniques such as sparse LU factorization[3, 4] to obtain the solution $x$. The exploitation of sparsity is key to making large practical problems computationally tractable.

In many real-life situations, however, the nonsingularity assumption for $A$ does not hold. For example, a circuit consisting of two isolated parts (without a common ground) except for inductive coupling through a transformer has a singular matrix. The reason for this is that one of the coils of the transformer is floating with respect to the other. Despite the singularity, the circuit is meaningful because the designer is not interested in the absolute potential but in potential differences between nodes of interest. Singularity is also encountered in the DC analysis of circuits which have nodes with no path to ground not involving a capacitor[1]. Such situations are especially common in circuits that rely on charge storage and transfer.

An important practical situation in which singular matrices occur is when real circuitry is macromodelled using idealized elements. This is common in the hierarchical design of systems, with some blocks represented in transistor-level detail and others as simple idealized macromodels. Often, the idealization is designed with a singularity – an example is a filter with a pole at zero frequency. In practical realizations, parasitics often remove exact singularities; even so, the matrix may be numerically singular owing to the finite precision to which numbers are representated in computers. Numerical singularities and near-singularities also occur in circuits with high-impedance nodes, for example inside high-gain op-amps in open loop simulation.

Previous approaches towards solving singular circuits have all been based on modifying the circuit to remove its singularities. A common method to avoid floating node problems is to insert a small conductance gmin, usually across parasitic diodes in MOS and BJT devices. Checking the circuit's topology using graph-based rules is often used to detect structures that lead to matrix singularites; the designer is required to change the circuit to remove them before proceeding with the simulation. These techniques are not always effective, even for the subclass of singular circuits that they address. For example, small values of gmin can cause the Newton-Raphson method to take very large steps, leading to non-convergence. It is impossible to detect certain singularities by graph-based rules, for example in a tank circuit excited at its resonant frequency. As shown in Section 3, the process of finding singularities can cause serious delays in design – it took more than a week for the three singularities in the circuit of Figure 5 to be found.

This work attacks the problem at the matrix level and is therefore uniformly applicable to any circuit situation that causes a singularity. As such, it does not require the explicit detection of the singularity, nor does the circuit need to be changed. Information about the location of singularities is produced as a by-product of finding the "best" possible solution of the singular circuit. Unless the singularity is the result of an error in the circuit, the solution provided is what the designer is looking for. A key feature of the method is that it uses LU factorization to exploit sparsity; hence it is applicable to large circuits, unlike other methods for dealing with singular systems[1, 5] (see Section 2.1). The method is best suited for circuits with a small number of singularities (e.g., with only a few floating nodes). For most practical circuits, this does not constitute a limitation.

The new technique is presented in Section 2. Computational considerations, crucial for the practical applicability of the method, are also addressed in Section 2. In Section 3, the method is applied to two industrial examples. In Section 4, the key features of the new technique, as well as related future work, are summarized.

## 2 Exploiting sparsity in singular systems

In this section, an efficient method for obtaining the "best" solution of a singular circuit is presented. The definition and properties of the minimum least-squares solution, its connection with circuits, the rôle of the singular subspace and traditional approaches for solving singular systems are the subject of Section 2.1. An outline of the new technique and its computational properties is presented in Section 2.2.

[1] It is sometimes possible in such cases to use charge conservation to eliminate the singularity using a variant of the present approach; this is a topic of current work.



0-7803-3177-6 $5.00 © 1996 IEEE

IEEE 1996 CUSTOM INTEGRATED CIRCUITS CONFERENCE

### 2.1 The minimum least-squares solution

Given a singular square matrix $A$ and a vector $b$, the object is solve for $x$ satisfying:

$$Ax = b \tag{1}$$

If $A$ is nonsingular, there is always a unique $x$ satisfying Equation 1[6]. If $A$ is singular, however, two possibilities exist: if $b$ is not in the *range* $R(A)$ of $A$, no solution $x$ exists satisfying Equation 1; if $b$ is in the range space of $A$, a continuum of solutions $x$ satisfy the equation.

Despite the lack of a unique solution in the singular case, it can be shown[5] that a unique $x^*$ does exist that is the "best" solution to Equation 1, whether $A$ is singular or not. $x^*$ is called the *minimum least-squares solution* and is defined to be the smallest possible vector (i.e., of smallest norm) that minimizes $||Ax-b||$. It is appropriate to think of $x^*$ as the best solution because of the following properties:

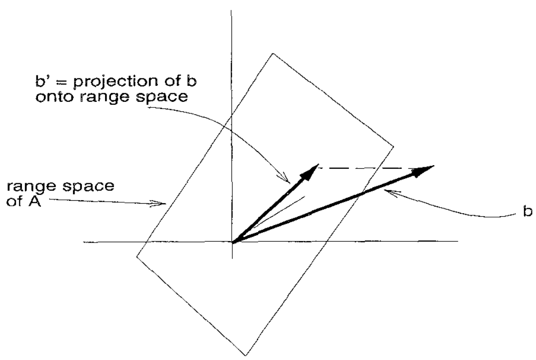


Figure 1: Orthogonal projection of $b$ on the range of $A$

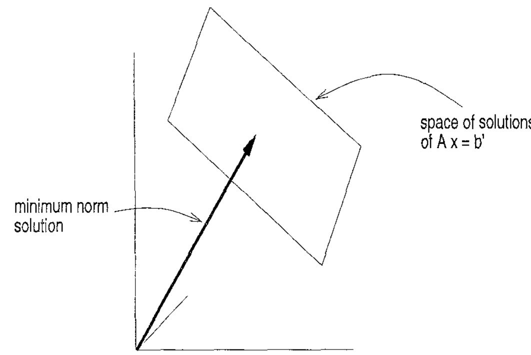


Figure 2: The MLS solution $x^*$

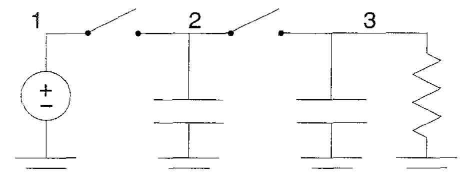


Figure 3: $b \in R(A)$: continuum of solutions

1. If $A$ is nonsingular, $x^*$ is the unique solution of Equation 1.
2. If $A$ is singular and $b$ is in the range of $A$, then $x^*$ is the solution of minimum norm that satisfies Equation 1.
3. If $A$ is singular and $b$ is *not* in the range of $A$, then there exists a unique vector $b'$ in the range of $A$ that is closest to $b$. $x^*$ is the solution of minimum norm satisfying $Ax = b'$.

The concept of the minimum least-squares (MLS) solution is illustrated in Figures 1 and 2.

It is useful to obtain an appreciation of how cases 2 and 3 above relate to circuits. Figures 3 and 4 show two circuits - the first is an idealization of a switched-capacitor circuit and the second a circuit segment generating a pole at zero frequency. Such structures are common in macromodels for system-level design. The circuit equations for Figure 3 can be shown to be an example of case 2, i.e., $b$ is in the range of $A$ and the circuit has an infinite number of solutions. The MLS solution for this case corresponds to a voltage of zero at node 2. The circuit of Figure 4 is an example of case 3 above; it does not have any solution at DC. For this case, $b'$ corresponds to setting the current source to zero; at this value, the circuit has an continuum of solutions. The MLS solution corresponds to a voltage of zero at node 1.

The two examples above were simple linear ones. Most practical circuits are nonlinear, hence it is of interest to examine how the MLS solution generalizes to the nonlinear case. For an arbitrary nonlinear equation $f(x) = 0$, the analogue of the linear MLS solution is a minimum-norm $x^*$ that minimizes $||f(x)||$ locally. Unlike the linear case, there may be several local minima, leading to multiple values for $x^*$, one for each local minimum of $||f(x)||$. An important connection between the linear and nonlinear cases is obtained through a generalization of the well-known Newton-Raphson method for solving nonlinear equations. The usual Newton-Raphson iteration is:

$$J\delta x = -f(x), \quad J = \frac{\partial f}{\partial x} \tag{2}$$

For $J$ nonsingular, $\delta x$ is well-defined and unique. An immediate generalization when $J$ is singular is to use the MLS solution of Equation 2 for $\delta x$. It can be shown that this variation maintains the local quadratic convergence properties of the Newton-Raphson method; the difference is that it converges to a *local minimum* of $f(x)$[7, 8], while the nonsingular Newton-Raphson converges to a zero of $f(x)$. Hence the linear MLS solution can be used in the Newton-Raphson method to obtain nonlinear locally MLS solutions.

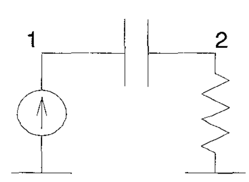


Figure 4: $b \notin R(A)$: no solutions

Let the size of $A$ (the number of rows or columns) be denoted by $n$, which for practical circuits can easily be in the thousands. An important concept is the dimension of the null space $N(A)$ of $A$[6]. Denote this (also called the dimension of the singularity of $A$) by $m$. In circuit terms, $m$ is the number of independent circuit segments that are not well-defined; for example, in Figures 3 and 4, $m$ is 1. In practical circuits, $m$ is usually much smaller than $n$ and rarely exceeds 20 or 30. It will be seen later that this fact is key to the practical applicability of the technique of this work.

Traditional approaches for obtaining the MLS solution use the singular value decomposition (SVD) or the QR decomposition of $A$[1, 5], which destroy sparsity and are hence inefficient for large circuits. A property of the MLS solution that is sometimes used is the so-called *normal equation*:

$$A'Ax^* = A'b \tag{3}$$

If $A$ is not square but of full rank with more rows than columns, it can be shown that $A'A$ is square and nonsingular. In such cases, the normal equation can be used with traditional nonsingular solution techniques to obtain the MLS This situation is not however easily applicable to circuits, which typically have square matrices; moreover, forming $A'A$ usually destroys sparsity to a considerable extent, reducing the effectiveness of sparse factorization techniques. The author is not aware of any existing approach for solving the MLS problem that exploits sparsity to the same degree as LU factorization techniques do for sparse nonsingular matrices.

### 2.2 Obtaining the MLS solution for sparse $A$

In this section, an outline of the new technique is presented.

Consider the most general case 3 above, with $b \notin R(A)$. The situation is illustrated in Figure 1. The first step in the new method is to obtain $b'$, the orthogonal projection of $b$ onto $R(A)$; this is the vector in $R(A)$ that is closest to $b$, i.e., $||b - b||'$ is the minimum least-square error achievable.

The equation $Ax = b'$ has a continuum of solutions, as illustrated in Figure 2. If $x_1$ and $x_2$ are two such solutions, then $x_1 - x_2$ in necessarily in the null space of $A$; hence in order to find the minimum norm solution $x^*$, the new technique starts with any solution $x$ of $Ax = b'$ and subtracts from it all its components in the null space of $A$. In other words, $x^*$ is obtained from any solution $x$ by subtracting from $x$ its projection onto $N(A)$.

In order to obtain numerical representations for the range space $R(A)$, the null space $N(A)$ and a solution $x$ to $Ax = b'$, the LU factorization of $A$ is used. Usually, LU factorization is performed only for nonsingular matrices; however, all square matrices, singular or otherwise, can be decomposed into LU factors. For concreteness, assume the Crout form of the LU algorithm, i.e., $L_{ii} = 1, \quad \forall i \in \{1, \ldots, n\}$. This makes $L$ always nonsingular; $U$ can be shown to have the same rank as $A$. The following example shows the LU factors of a matrix that is rank-deficient by 1 (i.e., $m = 1$):

$$A = \begin{bmatrix} 8 & 6 & 6 & 8 \\ 8 & 10 & 9 & 11 \\ 32 & 28 & 29 & 38 \\ 24 & 22 & 23 & 30 \end{bmatrix} \tag{4}$$

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 4 & 1 & 1 & 0 \\ 3 & 1 & 1 & 1 \end{bmatrix}, \quad U = \begin{bmatrix} 8 & 6 & 6 & 8 \\ 0 & 4 & 3 & 3 \\ 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{5}$$

The LU factors of a singular matrix can be obtained by small modifications to the procedure for the nonsingular case, as follows: when singularity is detected by the LU factorization routine, the remaining lower submatrices of L and U (of size $m$) are set to the identity and zero matrices of size $m$, respectively. Pivoting strategies and other heuristics used for nonsingular LU factorization carry over unchanged to this generalization.

Once the LU factors are available, they can be used to find $N(A)$, $R(A)$ and a solution to $Ax = b'$. First consider the problem of finding *a* solution $x^0$ of the consistent equation $LUx = b'$, assuming $b'$ has already been computed. Since L is nonsingular, we have $Ux = L^{-1}b'$. It can be shown the last $m$ rows of U can be chosen to be identically zero without loss of generality if $A$ is rank-deficient by $m$. Therefore *a* solution is to set the last $m$ variables of $x$ to zero, and solve for the first $n - m$ variables using normal reverse-substitution.

A slight variant of the above can be used to produce a basis for the null space $N(A)$. To obtain $m$ linearly independent vectors in $N(A)$, the last $m$ variables of $x$ are chosen as follows: $\{1, 0, \ldots, 0\}$, $\{0, 1, 0, \ldots, 0\}$, ..., $\{0, 0, \ldots, 1\}$. With each of these choices, $Ux = 0$ is solved for the top $n - m$ remaining variables using back-substitution. The resulting vectors form a basis for $N(A)$ and are linearly independent. Let the basis be denoted by $\{n^1, \ldots, n^m\}$.

The projection of $x^0$ onto $N(A)$ is obtained next. For this task, it does not suffice to have a basis that spans $N(A)$; it is necessary to obtain an *orthonormal basis*. Given the basis $\{n^1, \ldots, n^m\}$, the standard Gram-Schmidt orthogonalization procedure [6] is used to convert it into an orthonormal basis. Let the orthonormal basis thus obtained be denoted by $\{n^{o1}, \ldots, n^{om}\}$.

Once the orthonormal basis is available, the MLS solution $x^*$ is easily obtained from $x^0$ by subtracting the projection:

$$x^* = x^0 - \sum_{i=1}^{m} \langle x^0, n^{oi} \rangle n^{oi} \tag{6}$$

where $\langle \cdot, \cdot \rangle$ denotes the dot product. Equation 6 eliminates any component of $N(A)$ from $x^0$, leaving the MLS solution.

The above assumed that $b'$ had already been calculated. This is in fact the first step in finding the MLS solution, and it is achieved by first computing a basis for the *orthogonal complement of the range space* $R(A)$ of $A$, rather than for $R(A)$ itself. The orthogonal complement is chosen for two reasons: 1. its dimension can be shown [6] to be the same as $m$, hence the space is small, leading to computational efficiency, and 2. since the orthogonal complement of $A$ is the same as the null space of $A'$[6], it can be computed in the same manner as the basis for $N(A)$ as described above, using $A' = U'L'$. Once an orthonormal basis for the orthogonal complement is obtained, $b'$ is derived by subtracting from $b$ all its components in this space, analogous to Equation 6.

The computation involved in the above procedure is dominated by the LU factorization and the Gram-Schmidt orthonormalization steps. LU factorization is almost linear in the size of the matrix $n$ when $A$ is sparse. The Gram-Schmidt procedure requires $O(nm^2)$ operations, i.e., it is linear in $n$ and quadratic in $m$. For this reason, the method is best suited for small $m$, a situation that holds for most circuit applications.

## 3 Results

In this section, two industrial circuits with singularities are presented and the efficacy of the technique demonstrated.

The first circuit is a system-level design for a programmable line termination circuit. The block diagram of the circuit is shown in Figure 5. The purpose of the circuit is to present a programmable impedance to a communications line in order to match its characteristic impedance and minimize echoes and reflections (accurate line matching is especially important for data communications applications). The circuitry is mixed analog-digital. The line current is sensed, filtered and digitized before being input to the digital section of the circuit. A DSP-based calculation produces a digital representation of the voltage that the desired termination would generate; this voltage is then converted to analog form, filtered and placed on the line using a transducer. During system-level design, most of the blocks were macromodelled using ideal elements such as linear controlled sources.

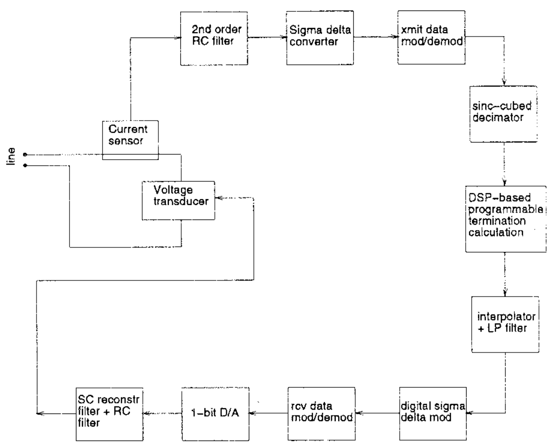


Figure 5: Programmable line termination system

The circuit as simulated by the designers contains three singularities ($m = 3$). Two of the singularities are due to deliberately designed poles at DC in the $\text{sinc}^3$ decimator and the interpolator/LP-filter blocks. A third singularity is the result of a benign error in the DSP macromodel, where the output of a sensing controlled current source is connected to a floating node. The circuit would not simulate in a proprietary internal simulator because the initial operating point computation would terminate with a singular matrix error. The existing topology check in the simulator having failed to locate the problems, it took the designers more than a week to find the causes of the singularities and devise workarounds.

Applying the new technique to the circuit resulted in its solution in four Newton-Raphson iterations. In addition, the degree of rank-deficiency $m = 3$ was found and information produced about the location of the singularities.

The second circuit, shown in Figure 6, consists of a chain of 32 transmission gates. The gates are alternately on and off. If the transmission gates were ideal, half the drains and sources in the chain would be floating ($m = 16$). In a well-modelled circuit, however, parasitics and leakage elements prevent exact singularities, hence the matrix is not exactly singular but nearly so. The circuit does not converge with the Newton-Raphson algorithm because the near-singular matrix leads to large steps, taking the algorithm outside its region of convergence, from which it does not recover.

When simulated with the new technique with appropriate threshold parameters, the circuit converges in 8 Newton-Raphson iterations. $m$ is detected to be 16 and information produced about the locations of the near-singularities.

## 4 Conclusion

A new technique has been presented for solving circuits with singularities. The method is based on finding the minimum

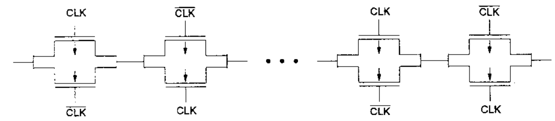


Figure 6: Chain of transmission gates

least-squares solution of $Ax = b$ efficiently when $A$ is sparse. LU factorization and Gram-Schmidt orthogonalization are the main computational steps in the new technique.

The method is applicable to large VLSI designs, unlike traditional techniques for least-squares solution that do not take advantage of the sparsity of circuit matrices. It is best suited for circuits that have a small number of singularities because the computation involved in Gram-Schmidt orthonormalization increases as the square of the number of singularities.

The application of the method to a system-level design with singularities and to an almost-singular circuit with off transmission gates has been presented in this paper. It is relevant to a number of other areas as well – RF circuits and harmonic balance, analysis of tank circuits and oscillators, transient analysis, optimization, homotopy methods, solution using charge conservation and AHDL simulation are examples. Work is currently continuing in application to these areas.


### Acknowledgments

The author is grateful to D.A. Rich for pointing out the need for the analysis of singular circuits in industrial design. He and H.S. Fetterman contributed the two examples. Thanks are due to K. Mayaram for useful discussions about singular circuits and charge-sharing and to Kishore Singhal for pointing out the rôle of the normal equations in MLS problems. R.C. Melville and M.H. Wright brought Björck's survey of least-squares methods [5] to the author's notice and P. Feldmann pointed out the application of the technique to RF CAD problems. Special thanks are due to K. Mayaram and D.A. Rich for suggesting CICC as a forum for this work and for their help in making the submission possible.


## References


[1] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes – The Art of Scientific Computing.* Cambridge University Press, 1989.

[2] C.W. Gear. *Numerical initial value problems in ordinary differential equations.* Prentice-Hall series in automatic computation. Prentice-Hall, Englewood Cliffs, N.J., 1971.

[3] K.S. Kundert and A. Sangiovanni-Vincentelli. *Sparse Version 1.3a User's Guide.* EECS Dept., University of California, Berkeley, July 1988.

[4] U. Schendel. *Sparse matrices: numerical aspects with applications for scientists and engineers.* Ellis-Horwood, 1989.

[5] Å. Björck. Least Squares Methods. Technical report, Dept. of Math., Linköping University, May 1991.

[6] K. Hoffman and R.A. Kunze. *Linear Algebra.* Prentice-Hall, 1971.

[7] E.L. Allgower and K. Georg. *Numerical Continuation Methods.* Springer-Verlag, New York, 1990.

[8] J.M. Ortega and W.C. Rheinboldt. *Iterative solution of nonlinear equations in several variables.* Academic Press, 1970.