Efficient and Fast Simulation of RF Circuits and Systems via Spectral Method

1. Project Summary

The proposed research will result in a new spectral algorithm, preliminary simulator based on the new algorithm will be substantially more efficient in simulation of RF circuits and systems. Work to date indicates that the algorithm is inherently parallel and thus will be suitable for implementation in parallel computing systems. Such analyses are very important in improving productivity of design of RF power amplifiers through determination of stability regions. Stability is a significant problem in design. Currently design of RF power amplifier requires approximately 2 weeks, but its stabilization requires 6 months because it involves high degree bench experiments. The work will be based on the spectral algorithm developed earlier by the authors of this proposal. The computer implementation of the algorithm will be made available to be tested by practitioners.

The following tasks are proposed:

1. For both the spectral algorithm and harmonic balance methods error analysis will be performed to determine the degree and the number of terms to be used in approximating the solution within a given error tolerance.
2. The error analysis will be accompanied by a stability analysis in order to determine the effect of error during the entire computation process.
3. A new and very efficient linear equation system solver will be developed that utilizes the special matrix structures being characteristic for spectral algorithms. The solver will use problem oriented preconditioners.
4. Iterative algorithms will be combined with coefficient extrapolation to reduce the size of the linear equations to be solved.
5. The algorithms will be implemented using an object oriented approach, coded in C++. Visual C++ development system will be utilized in constructing the software structure. The concept of dynamic link libraries will be utilized in building the simulator to assure its flexibility, convenience in use, and adaptability to applications. The module structure of the software will allow its integration into simulators currently used in industry.
6. The simulator will be geared toward support for design of communication equipments, in particular it will include facilities for evaluation of stability regions of power amplifiers.

The software system to be developed will be modular and it is anticipated that the individual modules will be made available to the users as soon as they are developed. The modular structure will make modules available to other applications if needed. It is anticipated that some modules will be general in nature and will be applicable in solving other dynamic problems.
2. Table of Contents

3. Project Description
   3.1. Introduction
   3.2. Goals of Proposed Research
   3.3. Project Research
   3.3.1. Results from prior work
      3.3.1.1. Spectral and harmonic balance algorithms
      3.3.1.2. Circuit simulation studies
         3.3.1.3. Simulation of transmission lines
         3.3.1.4. Computer implementation
   3.4. Related work elsewhere
   3.5. Error control
   3.6. Refinement of algorithms
   3.7. Computer implementation

4. Plan of work and responsibilities

5. Appendix - Error analysis in SPEC

6. References

7. Budget
3. Project Description

3.1 Introduction

Current circuit simulators need improvement in the speed of simulation, better error control, and tools allowing control of multiple simulation runs for optimization, sensitivity analysis, and generation of design curves such as stability regions of RF power amplifiers.

Previous research has yielded an algorithm based on Chebyshev polynomials which is faster and contains better error control than existing circuit simulators. This algorithm, which we call the “spectral” algorithm, has been successfully applied to systems containing transmission lines and transistor circuits.

The spectral algorithm has many advantages over the methods used in currently available circuit simulators. In time domain simulations, only local truncation error (LTE) is controlled. LTE is defined under the assumption that there was no error in previous iterations. This assumption is invalid as computation proceeds in a sequence of operations. Control of LTE does not directly limit the global error and simulation results may not be accurate. The spectral algorithm controls the global error thus guaranteeing solution accuracy. Another advantage is the use of analytical transformations which lead to simple linear equations for numerical solutions. These transformations speed-up the execution of simulation. The spectral algorithm has also a great potential for parallel implementation which is of increasing importance.

Harmonic balance techniques present a useful alternative for spectral methods, when Chebyshev polynomials are replaced by the trigonometric function system and therefore instead of Chebyshev series, harmonic series are used to approximate the solution. It is well known that Chebyshev polynomials give the best minimax function approximation. Trigonometric series approximation also have advantageous convergence properties as well as nice and simple structure. A systematic comparison of the two techniques will be very important for construction of an algorithm exploiting appropriately the advantages of both methods.

3.2 Goals of Proposed Research

The overall goals of the proposed research are refinement of the spectral and harmonic balance algorithms developed previously and their implementation in the form of a prototype simulator. This simulator will be available for simulation of integrated circuits and in particular applied to components of wireless communications systems such as power amplifiers for analog and digital modulation in transmitters. Experience to date indicates that a simulator based on spectral techniques will require substantially less CPU times than currently available simulators, will be suitable for simulation of high-speed digital circuits and interconnections, and will be superior in application to analog, and mixed analog-digital circuits which represent the type of circuits used in portable communications equipment.
The following tasks are proposed:

- **Error Control:** Error control techniques will be developed for oscillatory/analog systems in order to obtain solutions with given tolerances and with minimal computational effort. This will allow automatic selection of integration parameters such as the number of windows and the degree of polynomial or trigonometric series used to represent circuit variables.

- **Stability Analysis:** The effect of round-off errors and error cummulation will be examined. A repeated stability analysis on different method variants makes us able to find the most appropriate methods for applications.

- **Refinement of Algorithms:** A new and very efficient solver of linear systems for the spectral and harmonic balance simulators will be developed. Specific matrix structures which were discovered to be characteristic for these algorithms [21] and which are important for building iterative solvers will be explored. The method will use a specific preconditioner to avoid instability.

  Coefficient extrapolation and iterative algorithms will be used to reduce the final size of the linear systems. These algorithms will be important in an efficient implementation of error control.

- **Computer Implementation:** The algorithms will be implemented using an object oriented approach, coded in the C++ programming language. They will be assembled into a spectral simulator which will form a basis for future commercial developments and applications. The special modular structure of the newly developed software makes it able to be integrated into simulators currently being used by the industry.

- **Simulator Testing:** The simulator will be applied in developing support for design of communications equipment. The following specific applications are proposed:

  - power amplifiers - use of multi-level SPEC program families for generation of design curves.
  
  - transmitters system/subsystems with digital modulation - evaluation of feasibility of simulation.

3.3 Proposed Research

3.3.1 Results from Previous Works of the Authors

3.3.1.1 Spectral and Harmonic Balance Algorithms
Research to date resulted in development of the spectral algorithm for simulation of electronic circuits and systems [24, 28, 29]. The algorithm differs substantially from those used in currently available simulators in the way that circuit equations are linearized, solutions are represented, and approximation errors are calculated. Approximation techniques based on harmonic series are known as harmonic balance methods. A comprehensive summary of the technical issues concerning this methodology is given, for example, in [37].

The circuit linearization is first performed along a trajectory (Newton-Kantorovich approach) yielding a system of linear time varying differential equations. The solution is determined in an iterative process with the second order approximation.

Circuit variables such as nodal voltages, branch currents, capacitor charges, and inductor fluxes can be represented in the form of a Chebyshev series,

\[ v_k = \sum_{r=0}^{a} \psi_{kr} T_r(x) \]

or as an harmonic series

\[ v_k = \sum_{r=0}^{a} (\phi_{kr} \cos(r\tau x) + \psi_{kr} \sin(r\tau x)) \]

where \( v_k \) represents the k-th variable, \( T_r(t) \) is the Chebyshev polynomial of degree \( \psi_{kr} \) are the unknown coefficients, and \( n \) is the order of expansion. In the harmonic series case, \( \tau \) is a given period, \( \phi_{kr} \) and \( \psi_{kr} \) are the unknown coefficients. The circuit equations are transformed into algebraic equations for the series coefficients which are constructed utilizing numerous properties of the Chebyshev polynomials [32,30] and trigonometric functions [37].

The extensive use of Chebyshev polynomial and trigonometric function properties in development of equations for expansion coefficients yields very simple, well structured final equations for computer solutions and is one of the reasons for superior numerical efficiency of the method.

The use of Chebyshev expansion is suggested by its convenience as well as by its good approximating properties. If \( f \in \mathbb{C}^1 [-1,1] \) and \( C_n \) is its nth order Chebyshev expansion, then

\[ U_n(f) \quad 11f - C_n 11_{\infty} \leq U_n(f) \left[ 4 + \frac{4}{\pi^2} byn \right], \text{ were } U_n(f) \text{ is the supremum norm of the difference of } f \text{ and its best nth degree polynomial approximation. Therefore Chebyshev series approximations are not much worse than the best uniform polynomial approximations of the same degree.} \]
Circuit simulators that are currently in use can control local truncation errors (LTE) only. Since LTE can accumulate, the global error in such simulators cannot be directly controlled. In contrast, the spectral techniques provide a means for controlling the global error in solutions. This is very important in all applications.

3.3.1.2 Circuit Simulation Studies

The spectral method implemented in the form of a preliminary simulator, SPEC, was applied in the simulation of several MOS digital circuits. Simulation results were carefully compared with those produced by SPICE. The integration parameters in the spectral method and in SPICE were adjusted such that both methods produced equivalent results. In the procedure for adjustment of integration parameters, the differences between the circuit variables computed using SPEC and the corresponding variables obtained from SPICE were monitored. The simulation results were considered equivalent when all differences were within 0.1% of the maximum value of the corresponding variables. The CPU times needed for generation of equivalent results on the same computer were recorded. A sample of results is given in Table 1.

<table>
<thead>
<tr>
<th>Circuit</th>
<th>Spice CPU Time</th>
<th>SPEC CPU Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>4th order linear circuit</td>
<td>33.46s</td>
<td>0.67s</td>
</tr>
<tr>
<td>CMOS flip-flop</td>
<td>398s</td>
<td>135s</td>
</tr>
<tr>
<td>one inverter</td>
<td>1.31s</td>
<td>0.5s</td>
</tr>
<tr>
<td>shift register</td>
<td>135s</td>
<td>17.8s</td>
</tr>
<tr>
<td>memory cell</td>
<td>32.31s</td>
<td>14.3s</td>
</tr>
<tr>
<td>32 inverters</td>
<td>570.91s</td>
<td>351.1s</td>
</tr>
</tbody>
</table>

Table 1: Comparison of simulation using SPICE and SPEC

The spectral methods were also applied in simple examples of RF circuits, and in the waveform relaxation framework for circuit [6, 7, 23] and partitioned systems [27] producing even more spectacular savings in comparison with SPICE as shown for the example adders in Figure 1. We expect similar efficiency in comparison to digital simulators such as RELAX, SWEC, SLS, etc.

3.3.1.3 Simulation of Transmission Lines

Simulation of transmission lines together with circuits containing active devices is required in analysis and design of high speed circuits, electronic packages [25, 26, 20], and communications equipment [9]. Spectral methods provide an excellent foundation to develop efficient algorithms for the computation of transients in non-linear circuits interconnected by transmission lines. Algorithms for lossless transmission lines and lines with DC losses were developed [18, 19]. Preliminary work on an algorithm for lines with frequency dependent losses was conducted [5]. Computational experiments confirmed the
numerical superiority of spectral algorithms and convenience of their implementation in the SPEC package. Similar advantages are expected in applying harmonic balance techniques.

### 3.3.1.4 Computer Implementation

The SPEC circuit simulator currently contains the following completed modules: an intelligent input file parser, a Chebyshev Transformation module, a module that solves linear systems resulting from Chebyshev Transformations, a module containing implementation of primitive circuit elements, and SPEC Power Tools.

The input file parser allows for easy input of new models without modifying the input parser code. Each element is entered into the input file by specifying its type, connection nodes, and optional parameters. The input file sparser knows what each entry of the element list is by its position in the list. It matches these entries with an externally supplied list that defines the new model.

Figure 1: CPU time comparisons for SPICE and SPEC simulating adders in the relaxation framework [6, 7]

The Chebyshev Transformations and related algorithms have been implemented. Two of the more important and most often used subroutines consist of expansion transformation of a function into Chebyshev series and evaluation of the values of an expanded (transformed) function.

The Chebyshev applied to differential equations lead to large-sparse linear systems of equations. The linear solver module takes advantage of the sparsity of these matrices to provide a quicker solution [17, 35]. In order to control round-off error cummulation, splitting and preconditioners will be applied on the system before computing the solutions.
The module containing implementation of circuit elements consists of only simple elements. These currently include: resistors, inductors, capacitors, voltage sources, and current sources.

Simulation of a circuit often requires multiple runs with the sweeping (or more sophisticated changes) of key parameters in the circuit description.

### 3.4 Related Work Elsewhere

The problem of accelerating simulation of integrated circuits is of significant, practical and theoretical importance and thus attracted attention of many researchers. Three other major approaches, different than the one which is the subject of this proposal, can be distinguished: waveform relaxation (WR), piecewise linear approximation (PWLA), and asymptotic waveform evaluation (AWE).

Waveform relaxation is an elegant idea of extending iterative techniques to differential equations [15]. Proposed integration methods were based on time marching techniques and convergence problems were encountered because of time discretization. Currently, there appears to be less activity related to WR. Spectral techniques will provide a better basis without time discretization for further improvement in WR [22].

PWLA is based on a series of linear approximations. In application to larger circuits with many non-linearities, PLA becomes inefficient because many segments for linearization are required to accomplish a solution with reasonable error. In addition, error estimates are not adequate as they do not account for error propagation from segment to segment. One idea in improving this technique is the application of piece-wise quadratic or cubic polynomials. Cubic splines are used in [38] to approximate circuit characteristics, and a simulation study is presented to find the optimal number of windows for a given error tolerance.

Recently, AWE is attracting the attention of several scientists and there is notable publication activity associated with this approach. AWE is based on a Pade approximation to the transfer function of a linear circuit. The approximation utilizes only the most dominant zeros and poles. The transient analysis is equivalent to evaluating the approximated transfer function. This approach can dramatically speed up transient analysis of a linear circuit at the expense of accuracy [33, 13]. An interesting least squares procedure for determining best Pade approximations is given in [39].

The main problem that AWE suffers from is lack of good linearization. The only scheme worked out for this method so far is piecewise linearization. However, the approximation error is difficult to estimate. Besides, for circuits with many nonlinearities, piecewise linearization (which is a linearization at an operating point) will not be efficient as it will require many segments and thus many repetitions of the AWE procedure. Linearization along the trajectory (which is used in spectral techniques) is not utilized in AWE. In addition, research effort in AWE, as observed in the publications, focuses on simulation of digital circuits, which are easier to linearize with the piecewise scheme [1, 11, 37]. But,
even in these applications a lot of heuristics are used to obtain reasonable Pade approximants.

There are also cases when such an approximation is not desirable at all. Some strongly nonlinear oscillatory circuits which work with “large signals” (especially RF circuits, for example power amplifiers) may exhibit very complicated (“chaotic”) behavior controlled by the nonlinearities. An approximation with very dim error estimates may unpredictably effect the model behavior.

Thus the application of AWE is rather limited to circuits which are linear or easy to linearize. For example, AWE could be applied to create macromodels of large, linear subnetworks reducing the complexity of the entire system. Circuit extractors produce subnetworks of passive, linear elements which add sometimes hundreds if not thousands of nodes. Macromodels created by AWE could be processed together with “nonlinear core” by a general purpose (time marching or spectral) simulator.

3.5 Error Control

Existing simulators based on time marching techniques do not have direct control of global error. The global error results from the LTE in each integration step. A typical simulation involves thousands of integration steps and LTE may accumulate.

Spectral techniques are able to provide very efficient global error control and allow for selection of integration parameters (degree of series, number of windows) which minimize the computational effort necessary for simulation. More detailed discussion of error control and some preliminary results are given in Appendix A. The error and computational effort estimates that will be developed for the spectral algorithm will form a basis for automatic adjustment of integration parameters before and during the simulation.

In order to develop an efficient error control system we have to combine the estimates of errors resulting from different sources. First, a nonlinear differential equation is linearized. The linearization error is a quadratic function of the increment of the solution, where the coefficients depend on the derivatives of the right hand side functions. During the Newton-Kantorovich iteration scheme we need better and better accuracy in the consecutive steps, where the linearization error becomes smaller. Therefore the number of terms in the Chebyshev and harmonic expressions has to be increased step by step. More terms lead to more accurate solution of that step, but this is achieved with a higher computational effort. However more accurate solutions of each iteration step imply faster convergence with smaller number of iteration steps to obtain a given accuracy of the final solution. Therefore in minimizing the overall computational cost we have to find a balance between the accuracy of each step and the order of the Chebyshev or harmonic series. This error analysis also has to be contained with the error resulting from the iterative solution of the large-scale system of linear algebraic equations to find the unknown Chebyshev coefficients. Better accuracy of each needs more terms with higher cost of each iteration, however if the number of iterations decreases then there is a decrease in the overall computation cost as well. Higher order approximation also increases the size of the linear
equation system to be solved numerically and as a result the solution becomes more expensive. Higher system dimensions also result in larger accumulated error.

The error and stability analysis of the methodology has to consider these errors and related trade-offs in a simulated system and must find the best computational strategy to obtain a required accuracy of the final solution with a minimum expense. In the proposed research we will develop two error control strategies. The first one give an optimal (or close to optimal) strategy before computations actually start. The second method will allow to make adaptable adjustments.

3.6 Refinement of Algorithms

a) Development of an iterative linear solver: The use of dedicated linear solver is one of the best ways to improve the performance of a simulator based on spectral techniques because solution of linear systems is repeated many times during a simulation. The matrix structure that is characteristic for the spectral method is very sparse and banded as shown in Figure 2 and thus is suitable for construction of an optimized solver.

This structure suggests the construction of a sparse matrix solver with a special permutation algorithm, to collapse the bands to positions around the main diagonal, where the elements magnitude decrease with the distance from the main diagonal as follows:

$$|a_{ij}| \leq \frac{1}{c^{i-j}}$$

where $a_{ij}$ are elements of the system matrix and $c > 1$ [21].

Bounds for the elements [21] will be improved and an iterative algorithm will be constructed for further improvement of the solver. In order to reduce the round-off error cummulation, multi-splitting ([40]), and/or an appropriate preconditioner ([41],[42]) will be applied.

b) Coefficient extrapolation: The relation between the zero and the extrema expansion coefficients [Appendix A] will be used to develop iterative coefficient extrapolation algorithms. These algorithms will have the feature that for a given set of extrema coefficients of order $n$, it will be possible to extrapolate the expansion of order $2n$ from the already expanded order $n$ without recalculating the full new expansion of order $2n$. The extrapolation will be based on the combination of the current extrema expansion and a new zero expansion of order $n$. This algorithm will be the basis for developing an efficient error controlled expansion for general functions.
3.7 Computer Implementation

The algorithms are implemented using an object oriented approach, coded with the C++ programming language and are assembled into a spectral simulator with an interface compatible with SPICE. The simulator will have extensions (such as the model compiler, Fourier and Lyapunov analysis modules) which will make it more convenient to use. The software will be built using Visual C++ development system and the architecture of the entire package will be based on newest software technology used in industry including the Microsoft’s Component Object Model. The concept of dynamic link libraries and other software development tools available from Microsoft will also be utilized in the work.

4. Plan of Work and Responsibilities

The overall development of simulator and in particular user and model interfaces will be supervised by Dr. O. Palusinski, ECE. Development of numerical algorithms for iterative solver, errors estimates, and harmonic balance method will be supervised by Dr. F. Szidarovszky, SIE, who also will have the responsibility in the construction of module for computing Lyapunov exponents and Poincare maps.

Year 1

(a) Developing the harmonic balance model  
(b) Improving the spectral model  
(c) Error control and stability analysis  
 -developing a general scheme for error control and stability analysis  
 -refinement of error estimates for oscillatory systems  
 -developing estimates for computational effort in simulation of RF circuits  

Year 2

Refinement of Algorithms

(a) Development of an iterative solver with an appropriate splitting and preconditioner
(b) Coefficient extrapolation methods
(c) Construction of compiler for models of circuit elements given in analytical form

Year 3

(a) Comparison of spectral and harmonic balance techniques with digital simulators
(b) Computer implementation

- Development of module for element models given in the form of look-up tables
- Finalizing the harmonic balance and spectral modules
  - Development of module for computing Lyapunov exponents and Poincare maps
Appendix A

Error Control

As it has been stated in the main body of the proposal, the first kind of error is the result of linearization. Let $F$ be a twice Frechet differentiable mapping from a Banach space into itself, then

$$
\|F(a) - L_{a_o}(a)\| \leq \sup_{c \in [a, a + \epsilon]} \|F''(c)\| \|a - a_o\|^2,
$$

(1)

Where $L_{a_o}$ is the linear Taylor polynomial of $F$ around $a_o$ and $F''$ denotes the second Frechet derivative of $F$. In our case $F$ denotes the nonlinear right-hand-sides of the differential equations to be solved, therefore $F''$ can be easily obscured in a closed form, and its norm can be estimated. The difference between the solutions of the nonlinear and linear differential equations can be estimated as follows. Let

$$
\dot{x} = h(t, x, x(t_o)) = x_o
$$

$$
\dot{z} = A(t)z + f(t), z(t_o) = x_o
$$

be a pair of nonlinear and linear differential equations, where $A(t)$ is matrix, $x, h, z, t$ are all vectors. Denote

$$
\varepsilon(t, x) = h(t, x) - [A(t)x + f(t)],
$$

and notice that the norm of this vector is estimated in (1). Therefore the nonlinear equation can be rewritten as

$$
\dot{x} = A(t)x + f(t) + \varepsilon(t, x).
$$

The solutions of the nonlinear and linear equations can therefore be written as

$$
x(t) = \Phi(t, t_o)x_o + \int_{t_o}^{t} \phi(t, \tau)[f(\tau) + \varepsilon(\tau, x(\tau))]d\tau
$$

and

$$
z(t) = \Phi(t, t_o)x_o + \int_{t_o}^{t} \phi(t, \tau)f(\tau)d\tau
$$

where $\Phi(t, t_o)$ is the fundamental matrix of the linear equation (see for example Szidarovszky and Bahill, 1997). Simple subtraction slows plot
\[ x(t) - z(t) = \int_{t_0}^{t} \phi(t, \tau) \epsilon(\tau, x(\tau)) d\tau \]

therefore

\[ \| x(t) - z(t) \| \leq \int_{t_0}^{t} \| \phi(t, \tau) \| \| \epsilon(\tau, x(\tau)) \| d\tau \]

estimating the error of linearization.

The second kind of error is the result of estimating the solution with a finite Chebyshev expansion. Let \( f \) denote the exact solution and \( C_n \) its \( n \)th order Chebyshev series approximation. The following results are known from the literature (see for example, Judd, 1998):

Fact 1. Assume that \( f \in C[-1,1] \), then there exists a finite constant \( B \) such that for all \( n \geq 2 \),

\[ \| f - C_n \|_{\infty} \leq \frac{B \log n}{n^2} \]

The result shows how fast \( C_n \) converges to \( f \) as \( n \to \infty \) in the supremum norm. In the weighted least squares norm an analogous statement is the following:

Fact 2.

Where \( C_r \) is the coefficient of the \( n \)th order Chebyshev polynomial \( T_r \) is the infinite Chebyshev expansion of function \( f \).

In order to see how fast the right hand side converges to zero, we need estimates of the Chebyshev coefficients \( c_r \). Such a result is given next.

Fact 3. Assume that \( f \in C[-1,1] \), then there is a finite number \( C \) such that for all \( n \geq 2 \),

\[ \| f - C_n \|_{\infty} \leq \frac{B \log n}{n^2} \]

Combining Facts 2 and 3 we see that

showing that the convergence is very fast for sufficiently smooth functions.

The third kind of error is the result of the numerical solution of the large-scale linear equation system to get the unknown coefficients. In the iteration process the error depends on the number \( I \) of iteration steps: The particular form of the error function depends on the special iteration scheme to be developed in the framework of the proposed research. The
error function will be an upper bound for the largest error is the Chebyshev coefficients. The error in the solution is then bounded by

In each linearization step, the three kinds of errors add up to give overall error bound for the approximating solution. The required error tolerance in the consecutive linearization steps should be decreasing. If ... are the required error tolerances for the consecutive linearization steps, then we have a complicated nonlinear optimization problem: minimizing the entire computation cost subject to the constraint that the sum of the three error bounds is not greater than in the nth step, k=1,2,... In this optimization problem the objective function as well as the constraints depend on the selected window size, number of Chebyshev terms in each window, as well as the number of iteration steps in solving the linear equations.

In the proposed research we plan to develop the particular forms for each cost item, and as good possible error terms. We also plan to develop a methodology to find an optimal (or close to optimal) computational strategy before computations begin as well as an adaptive adjustment scheme for automatic error control during the computation process.

References
