[269] A. S. Besicovitch, *Almost Periodic Functions*. New York: Dover, 1954.
[270] I. W. Sandberg, "Volterra-like expansions for solutions of nonlinear integral equations and nonlinear differential equations," *IEEE Trans. Circuits Syst.*, vol. CAS-30, Feb. 1983.
[271] ——, "Theorems on the computation of the transient response of nonlinear networks containing transistors and diodes," *Bell Syst. Tech. J.*, vol. 49, pp. 1739–1776, Oct. 1970.
[272] E. B. Lee and L. Markus, *Foundations of Optimal Control Theory*. New York: Wiley, 1968.
[273] I. W. Sandberg and A. N. Willson, Jr. "Some theorems on properties of DC equations of nonlinear networks," *Bell System Tech. J.*, vol. 48, pp. 1–34, Jan. 1969.
[274] B. Widrow, J. M. McCool, M. G. Larimore, and R. J. Johnson, Jr., "Stationary and nonstationary learning characteristics of the LMS adaptive filter," *Proc. IEEE*, vol. 64, pp. 1151–1162, Aug. 1976.
[275] R. F. Brown, Compartmental system analysis: State of the art," *IEEE Trans. Biomed. Eng.*, vol. 27, pp. 1–11, Jan. 1980.
[276] M. Zakai, "On the optimal filtering of diffusion processes," *Z. für Wahrscheinlichkeitstheorie und verw. Geb.*, vol. 11, pp. 230–243, 1969.
[277] V. E. Beneš, "Exact finite dimensional filters for certain diffusions with nonlinear drift," *Stochastics*, vol. 5, pp. 65–92, 1981.
[278] D. J. Hill and P. J. Moylan, "General instability results for interconnected systems," *SIAM J. Contr. Opt.*, vol. 21, no. 2, pp. 256–279, Mar. 1983.

Irwin W. Sandberg (S'54–M'58–SM'73–F'73) was born in Brooklyn, NY, on January 23, 1934. He received the B.E.E., M.E.E., and D.E.E., degrees from the Polytechnic Institute of Brooklyn, Brooklyn, NY, in 1955, 1956, and 1958, respectively (Westinghouse Fellow 1956, Bell Telephone Laboratories Fellow 1957–1958).

Since 1958 he has been employed by Bell Laboratories, Murray Hill, NJ, where he presently is a Distinguished Member of Technical Staff in the Mathematics and Statistics Research Center. He has been concerned with analysis of radar systems for military defense, synthesis and analysis of active and time-varying networks, with several studies of properties of nonlinear systems, and with some problems in communication theory and numerical analysis. His more recent interests have included compartmental models, the theory of digital filtering, global implicit-function theorems, and functional expansions for nonlinear systems. He is a former Vice Chairman of the IEEE Group on Circuit Theory, and a former Guest Editor of the IEEE TRANSACTIONS ON CIRCUIT THEORY Special Issue on Active and Digital Networks.

Dr. Sandberg is a member of the American Association for the Advancement of Science, Eta Kappa Nu, Sigma Xi, Tau Beta Pi, and the National Academy of Engineering.

A Historical Review of Circuit Simulation

DONALD O. PEDERSON, FELLOW, IEEE

Abstract—**Within the Circuits and Systems (CAS) Society, developments in computer-aided circuit analysis and circuit design commenced in the early 1950's using the earliest digital computers. Initially, computer-aided circuit analysis of linear circuits was used in design optimization, design centering, and in determining the effects of parasitics on circuit performance. Although this use of computer-aided circuit analysis has continued, computer-aided design (CAD) and circuit design automation within the CAS Society are now principally concerned with problems associated with the overall design and evaluation of very large circuits and systems.**

This paper is a review of a major thread of CAD activity which has occurred within CAS from the earliest and remains of major interest. This thread involves computer-aided circuit analysis (circuit simulation) and its use in CAD systems. Fortunately, several excellent review papers have appeared within the past year or two to document well the technical milestones, as well as the problems of interest at the present time. It is possible then, in this paper, to concentrate on the developments in our present capability of circuit simulators, stressing the significant trends, noting some early developments which did not become major aspects, and observing the interchange between theory and practice.

Manuscript received September 6, 1983.
The author is with the Department of Electrical Engineering and Computer Sciences and the Electronics Research Laboratory, University of California, Berkeley, CA 94720.

I. INTRODUCTION

OVER TWO-THIRDS of the IEEE's first century had passed before the digital computer was used in the analysis of electrical circuits. A computer consisting of electromechanical relays was programmed in the very early 1950's to solve the algebraic equilibrium-condition equations of a linear electrical network in the sinusoidal steady state [23]. With "rapid" analysis available, it was possible to employ optimization techniques to achieve excellent designs of electric filters [1]. From this relatively late start from the standpoint of digital computers, there has been a steadily increasing use of computers in the analysis, evaluation and testing of electric and electronic circuits and systems.

The phrase that best characterizes the circuits and systems of today is large scale. To illustrate the large-scale problem, consider today's very-large-scale-integrated (VLSI) circuit [16], [39], which may consist of hundreds of thousands of separate components, principally transistors. In the electrical design of a VLSI circuit, in contrast to the

0098-4094/84/0100-103$01.00

electrochemical processing and material aspects, there is involvement with computer and digital system operation and design, functional, logical, electrical, and physical design, verification, and testing. One must first study the overall function to be achieved with respect to the required data- or signal-handling process and to partition the function into smaller entities. Next, each subfunction is designed to achieve the logic requirements. At the electrical level, a sequence of designs of small segments is made with particular attention to the electrical interaction of these parts. This is followed by the physical design, i.e., the actual layout, of the patterns of metal, semiconductor, and insulating material by which the components and the interconnections are achieved. At each step of the design process, procedures (computer aids) must be developed to achieve verification of the electrical, logic, and functional behaviors of the segments. Computer aids are also critical in devising test sequences in terms of electrical and logical input sequences to assure that the circuit as manufactured is performing as designed.

Many of these design and evaluation activities are or have become the province of other IEEE Societies. It is possible then to restrict the scope of this paper to a topic which has always been central to the CAS Society, viz., computer-aided circuit analysis, now referred to as circuit simulation.

Many excellent review papers have recently appeared on circuit simulation, cf., [26] and optimization [8], [59], which are major topics of interest within the CAS Society, as well as on logical and functional design and testing which appear in other journals of the IEEE. An excellent tutorial introduction to circuit simulation is also available [66]. The main focus of this paper is the history of the computer-aided analysis of electronic circuits. The principal sections of this paper constitute the author's overview of the emergence of circuit simulation with particular attention to several of the critical decisions and developments which had to be made. Because of the CAS Society interest, a short section is included on the topic of optimization.

Within CAS, another significant research interest concerns graphs and graph theory. The researchers in this area have seen application possibilities in the placement and routing programs for printed circuit PC boards and now ICs. Placement and routing techniques for ICs have been based upon the use of "standard cells" and "gate arrays." For the former, sets of basic circuits (building blocks) of standard height and various widths are predesigned electronically and physically. Each block accomplishes a specific electronic or logical function. These standard cells are placed or fixed on avenues with routing channels between the avenues. The problem, then, is to choose the best location for each standard cell on the best avenue and then to route the interconnections between the cells. Feller (of RCA, with the MP2D program and its predecessors) was one of the first to produce a workable system, one which is still in use [21]. Similarly, Gummell and his staff achieved a successful system at Bell Labs—the LTX system [57], [58].

In the gate-array approach, also called a master-slice technique [12], circuit modules in IC are placed at specific grid locations. Placement and routing programs are used to achieve the interconnection of the modules and implement the complete IC design. A large number of commercially available design systems are now available for the gate-array technology. The standard-cell and gate-array design methodologies are often referred to as semi-custom design methods.

At many locations, both industrial and academic, circuit and system researchers have attacked the PCB and IC placement and routing problem and have contributed significant results. The interested reader is referred to an excellent review article on layout by Soukup [70]. In a recent paper, Newton has examined the broad range of layout problems for VLSI circuits [54].

II. Early Theoretical, Algorithmic, and Program Developments

As mentioned above, from the early 1950's, circuit and system researchers have always been interested in the theoretical problems of computer-aided circuit analysis, optimization, and automated design. In many cases, results from applied mathematical physics and mathematics have been identified and used to advantage.

The problem of the "best" way to formulate circuit equations had to be addressed early. The common mesh-equation formulation was not well suited for evaluation with the digital computer of those days. The capacitive elements as well as the high-resistance sources were troublesome with this formulation. In early work, Bashkow looked at alternate forms of a complete description of an electrical network and devised the A matrix [2]. This in part emanated out of his studies of dynamical systems as described in Whittaker and Watson. The A matrix is widely accepted as the forerunner of the state-variable description of an electrical network and was used in the early years of CAD as the basis of equation formulation in CAD programs [38]. With subsequent developments, however, it became clear that the state-variable approach, although very significant for theoretical studies, is not the best basis for computer programs for circuit analysis. As brought out below, the modified-nodal and sparse-tableau formulations are better. For one who has observed the developments in this field throughout the 30 years of its existence, the fierce disagreements of just ten years ago on this subject make for interesting and frustrating memories.

A decade after the first use of relay digital computers for circuit analysis, one of our first CAD experts, the late Frank Branin, came into our field from chemistry. He was interested both in the description of large electrical networks and also how computer programs can be written in such a way to achieve adequate computer analysis [7]. His first efforts led to the Program TAP [4]–[6], [9]. Because of the inevitable problems with a new software system, TAP was never released into the public domain. Nonetheless, TAP formed the basis for two very important early CAD

programs, ECAP1 from IBM and Norden [31] and the PREDICT program from a different division of IBM [30]. Both of these programs again suffered the usual fate of first-generation software systems—they were hard to use, were very unfriendly, had tendencies to nonconvergence in the solution of the equations, and so forth.

The successes and failures of the first programs led to new programs, in part because of the critical need to study problems associated with radiation effects in electronics for military systems. The PREDICT program led to SCEPTRE, which for almost 20 years has been used successfully, principally for aerospace needs [42]. Another new program based on the initial efforts was Malmberg's NET1 from the Los Alamos National Laboratory [40]. Both SCEPTRE and NET1 at that time used explicit-integration and/or predictor–corrector techniques in the solution of the integro-differential equations of the nonlinear systems of interest. These systems were usually very stiff types with very large numbers of widely varying eigenvalues at any equilibrium point of interest. Therefore, if the computer solution was possible at all, the time consumed in the solution was extremely large because of the very small time steps that were necessary to maintain stability of the numerical methods.

III. Implicit Integration, Nodal Analysis, and Early Nonlinear DC Programs

At least two independent paths of investigation arose in response to the time-constant problem. At Autonetics, now a division of Rockwell Corporation, a group of circuit and electronic device experts working with applied mathematicians used alternative approaches to those of the initial CAD programs. In retrospect, one can say that a kindergarten approach was employed. But I hasten to add that such an approach was not chosen because of naivity. In contrast to a state-variable formulation for electric equations, a conventional, datum-node analysis technique was used. Simple circuit schemes such as the inclusion of 1-Ω resistors and the use of Norton equivalents avoided problems with floating-voltage sources and the like. Inductances were either neglected or were treated separately. In addition, a simple implicit integration scheme, the Backward-Euler Method, was used. In implicit integration, the set of integro-differential equations become, for a time step at a time point, a set of static algebraic equations, i.e., a set of dc equations, the solution of which is straightforward [43]. The program which resulted from this effort and these decisions was TRAC. TRAC has had an effect comparable to that of TAP; it had a large effect on subsequent computer-aided circuit analysis [35].

Transportability was another early recognized difficulty. TRAC did not see wide usage except for special aerospace concerns involved in defense electronic contracts. One reason for the limited use was the fact that the code was not transportable to other machines; several key routines were written in the assembly language of a particular computer. Transportability to other machines, even of the same type, was difficult. The assembly language subroutines were rewritten in Fortran at Berkeley in 1968–1969. Subsequently, a new program in Fortran based upon the same choice of techniques and algoithms was written by Jenkins and Fan in 1969–1970. This work evolved into the TIME program at Motorola [34] and the SINC program at Berkeley.

At approximately the same time as the TRAC developments, Shichman at Bell Labs [64] proposed a second-order implicit-integration scheme that proved very effective and produced better performance relative to the revised TRAC. The program which he wrote as a vehicle to investigate this scheme was CIRPAC [68], [69]. The superiority of using the second-order integration scheme was clear. It was soon evident that Gear, working with large-scale systems and numerical integration techniques, had also evolved comparable backward integration schemes—variable-order, variable-timestep implicit integration routines [22]. Shichman and Gear's second-order routines were seen to be substantially the same. Although higher order implicit integration schemes can be used, the range of applicability is much smaller and their use has in circuit simulation been limited.

Howard at Berkeley, in 1967, before the above programs were available to him, encountered a problem with a seemingly simple, four-transistor, nonlinear circuit which needed to be solved with respect to not only the dc state but how it was effected by temperature. Hand analysis and evaluation proved exhaustive. Working in an interactive mode, on what would now be considered less powerful than a modern microcomputer, he wrote a program in terms of a set of nodal equations, modeling the transistors with a simple Ebers-Moll nonlinear model, linearizing the equations at the equilibrium point of interest and using Newton-Raphson and excursion-limiting techniques to iterate to a solution. The result was the BIAS program which was subsequently expanded and rewritten by McCalla in 1969 into the BIAS3 program [44]. This nonlinear dc program was released into the public domain. It had extensive use not only at Berkeley, but also at a number of locations on three continents. The experience with this simple program, as well as with TRAC and CIRPAC, had a major effect at Berkeley on the next generation of circuit simulators.

IV. The Sparse Tableau

In contrast to the developments at Berkeley in the late 1960's, which included a high component of engineering heuristics, an independent effort starting from a theoretical base took place at IBM Research. Hachtel and his colleagues at Yorktown were aware of the difficulty of network equation formulation as it applied to computer programming and developed an elegant, award-winning, complete description, called the Sparse-Tableau [25]. From the start, they neglected the usual, classical concentration on obtaining the most concise matrixes. Instead, they placed in evidence all element descriptions and connections in

elementary form, very suitable for programming, and the all-important aspect of the sparsity of the connection matrix.

Program ASTAP was developed at IBM based on the sparse-tableau representation of the electrical network [32], [74]. In addition, very sophisticated methods were included to obtain compiled machine code for the network description in order to achieve the fast repeated analysis suitable for statistical analysis of the circuit as parameters and elements are changed. Variable-order, variable-step implicit integration schemes were employed, as well as suitable methods to linearize the nonlinear equations at an equilibrium point. ASTAP has come into worldwide use within the IBM Corporation. It has also been made available for a fee outside IBM, although its usage outside of IBM has not been extensive primarily because of the computer cost of simulations.

V. Modified-Nodal Analysis (MNA)

In 1969–1970, Rohrer initiated a graduate student project, with approximately ten students, into the investigation of circuit analysis using digital computers. Based on his work at Fairchild, as well as the results working with graduate research students, all aspects of the problem were studied: network formulation, linearization, integration techniques, sparse-matrix techniques, gaussian elimination and LU decomposition, pivoting techniques, etc. The results of this project, i.e., the collection of conclusions and coded routines, formed the CANCER program [47]. The major components of this program will be brought out shortly.

In parallel to the CANCER project was the development by McCalla of the SLIC program [33]. SLIC was based upon the BIAS program and included new linear circuit analysis developments and new nonlinear device modeling. A particular feature, now incorporated in most simulators, was the determination of the dc state of the circuit followed by the evaluation and printout of all bias dependent parameters of the transistors.

The interaction among the several circuit simulation projects involving SINC, CANCER, and SLIC at the same location was extremely fruitful. It should be noted that the programs incorporated floating voltage sources, inductances, etc., as side constraints. That is, the conventional nodal analysis procedure was modified.

The modified-nodal-analysis technique was developed in a formal sense simultaneously by Ho and his colleagues at IBM [28], [29]. This technique was included in the ICD program written in APL. ICD is an excellent interactive program for on-line simulation and circuit design [28], [29], [76].

The circuit simulators cited above (in those days they were sometimes called third-generation simulators) shared common features: a modified form of nodal analysis, modified to be able to take care of voltage sources, floating sources, and inductive elements; a first- or second-order backward integration technique; advantage was taken for memory considerations and computer run time speed of the sparsity of the matrix of the linearized elements values of the circuit; pivoting to maintain sparsity (usually a Markowitz [41] or Berry reordering [3], Newton-Raphson linearization modified with excursion limiting, and LU solution of the equations. Further, after very extensive use in instructional courses with hundreds of students and many thousands of accesses, a hardware description language, i.e., an input circuit language, evolved which was relatively "friendly" from the standpoint of the new user. Similarly, because batch operation was needed, and the fact that simple graphics were unavailable at that time, simple inexpensive means were used to achieve line-printer plots of the output response for easy evaluation of the simulated circuit performance, as well as circuit and device parameter values at a chosen operating or initial point. With these features, the programs could be considered not just analysis programs, but electrical circuit simulators. A "dry lab" had been achieved.

After very extensive use by undergraduate and graduate students in class-assigned problem sets, the CANCER program evolved into Nagel's SPICE1 program [48]. SPICE1, as well as SLIC and SINC, were placed into the public domain. All have benefited from the availability of these simulators. Very importantly, feedback from usage has been an important aid in the continual development and fine tuning of this type of program and for the increased applicability of these programs to new circuit applications and types.

Nagel, in his doctoral studies, studied thoroughly all of the experience of the prior ten years with this type of circuit simulator and investigated the possibilities of improvement in making different choices. In particular, he studied thoroughly the synergism aspects of proper choice of all of the components and techniques needed in a circuit simulator. In early 1975, SPICE2 emerged, which has become a worldwide CAD tool [49], [13].

It must be stressed that the early workers found that programs which achieve CAD are complex software systems in their own right. Although there has been a need for theoretical and practical investigations in all of the algorithmic aspects of a CAD design procedure, there has also been a need to concentrate on the overall software system. The set of "best" algorithms and techniques may not, and usually does not, lead to the best software system. Rather, one must determine the best set of algorithmic procedures in order to be able to achieve a program package that is optimal or at least well conditioned in itself. These choices were made in the popular, well-used programs ASTAP and SPICE2.

It is interesting to compare ASTAP and SPICE2. ASTAP has a slow setup time to achieve fast repeated analysis. SPICE2 achieves fast circuit input checking with the small analysis time penalty of MNA. The former is very advantageous for the repeated use in statistical analysis. Fast turnaround time at low cost is important for initial design evaluation since 80 percent of simulation runs fail at input time.

VI. Other Contributions

The circuit simulators described above, e.g., ASTAP and SPICE, do not of course constitute the only successful and important efforts in this field in the past two decades. As mentioned earlier, the author is concentrating on the programs with which he is most familiar and which are representative of the developments which have occurred. For completeness, however, mention must be made of the excellent work in circuit simulation done at the University of Illinois, Urbana, under Trick. Wing at Columbia University has also made many fundamental investigations and has led program development efforts. Director and his students at Florida and CMU have made major contributions, as have Engl at Aachen and DeMan at Leuven, in Europe. Spence's early work in interactive CAD must be mentioned. At Bell Labs, major results have been produced by Scharfetter, Dowell, Kozemchak, and Nagel. The ADVICE program is Nagel's latest version of SPICE.

The work at Autonetics (now Rockwell) did not stop with the TRAC program. There has been a continual effort there, particularly by Bastian, which led to the SYSCAP programs. Similarly, the Air Force has continued to sponsor investigations and developments with SCEPTRE. In Japan, many significant developments in circuit simulation have been made, particularly in industrial concerns [71].

VII. Large-Scale Circuits

Initially, the SPICE-type programs were thought to be limited to a maximum circuit size of a few hundred elements and devices. It was believed that simulation of larger circuits would limit the patience of the designer with respect to computer run time and exhaust his available financial resources. It was believed that the designer of a large system would always partition the circuit into manageable segments, and this has been certainly true. Nonetheless, for the large-scale circuits and systems of interest, particularly VLSI circuits, the interaction between the partitions has almost always been significant. One can not always assume that all parasitic aspects and significant delays have been included or estimated correctly. Thus, it has been necessary for design and evaluation purposes to simulate larger and larger circuits. This has been particularly true of circuits consisting of large regular arrays such as RAM and ROM memory circuits.

Two simultaneous developments in the past have aided the above task. First, faster, larger mainframe computers have become available and less costly; thus, cost-effective simulation of large circuits is possible. Second, and as brought out below, developments have been made with the circuit simulators themselves which have provided significant speed-up, i.e., less run time. It should also be mentioned that the interaction problems in LSI and VLSI circuits, as mentioned above, have been so severe that it has been essential in spite of the cost to make large numbers of long simulation runs, of the order of days of mainframe computer time, to ascertain the cause of critical-path delays. From every major electronics company has emanated stories concerning the success of circuit simulation in isolating significant problems. This has led to the use of circuit simulators with dedicated computers for the analysis of very large circuits.

Many workers and many schemes have been involved in the improvement in simulation speed. Epler, at TI, has noted that a factor of five improvement can be achieved over an early release of the SPICE program by "tuning" the software to the system [20]. McCalla, Nagel, and Cohen, in their work over ten years with SPICE and SLIC, have also observed this kind of improvement by tuning. Their use of compiled machine code for central portions of the program also decreased run time. Newton, in the mid-1970's, noted, in developing a special interactive circuit simulator for an early desktop computer, that he could achieve large ratios of improvement by very carefully noting capabilities of the computer [50]. Starting from an initial straightforward use of high-level language, he was able to achieve a factor of 20 improvement in speed. This type of improvement, although not as large, has also been observed recently with interactive circuit simulators used on presently available desktop computers [24].

An early speed improvement technique that proved very effective has been called the bypass scheme. In bypass, the terminal voltages of all electronic devices are monitored from time-point to time-point. If there has been little change, the evaluation of the device is bypassed. Since for circuits of 100 transistors or less the device evaluation time consumes 80 percent of the analysis time, the improvement can be significant in many IC simulations.

In addition to the (heuristic) bypass scheme, theoretical efforts have also contributed significant speed-up. Branin, following along the lines of the early work of Kron, and which started Branin in our CAD activities, introduced branch-tearing techniques to circuit simulation [4], [37]. Other significant work has involved node-tearing techniques and latency [60], [65], [75]. The technical aspects of these developments are well reviewed in the simulation summary of October 1981 [26].

VIII. Dedicated and Vector Computer Investigations

Investigations to improve circuit simulation speed have also centered on the computer itself. Since different machines given different limitations or lead to different choices of the algoithms for simulation, tailor-made, nonportable programs can be developed. For a particular dedicated computer, Program SPUDS by Cohen was the result of an investigation which started with the simulation results using a high-level language [14]. He then investigated what speed-ups can be obtained with assembly language routines. Finally, he developed special-purpose micro-coded instructions from the available machine repertoire that were particularly suited to the tasks of circuit simulation. For the new sparse-matrix solution instructions, speed improvements of 20 were obtained. However, overall simulation performance with Program SPUDS for reasonable size circuits was only a factor of three improvement in relation

to a tuned simulator written in a high-level language using adequate compilers.

Other developments have utilized vector computers [27]. Most large circuits make highly repetitive use of a few basic circuits and the use of a small number of distinct electronic device models. Thus, it is conceivable that a vector processor can be employed where these individual subscircuits and device types can be processed in parallel in the vector mode. The CLASSIE Program, by Vladimirescu, is one result of such an investigation [72], [73]. Due to typical circuit sparseness and the gather–scatter problem [10], vector schemes do not provide too large improvements in simulation speed. Approximately up to a factor of ten improvement over SPICE2 in circuit simulation speed can be obtained for highly repetitive large circuits.

In other recent efforts, circuit simulator SPICE has been used with array processors. Again, by exploiting the vector mode of analysis, factors of five in improved performance can be obtained over that performance obtained on a super-minicomputer [15], [67].

In major new investigations, special-purpose routines necessary in circuit simulation are being designed into hardware (ICs). In these developments, the nonportability aspect is taken to the limit, and the "micro code" of the computer is being optimized for the task at hand.

IX. Timing and Relaxation-Based Simulation

Timing Simulation is circuit simulation in which approximations are purposely introduced with relaxed accuracy to achieve greatly improved simulation speed. In a sense, timing simulation has been with us from the beginning of CAD. Individual workers have programmed rudimentary circuit analysis schemes to solve a problem at hand. Often times, these seemingly crude programs were sufficient for significant developments within IC companies. A systematic approach to this topic, however, was started approximately ten years ago by Gummel and his coworkers at Bell Laboratories, in the process of developing an integrated set of software tools (one of the first) for the electrical and physical design and verification of an integrated circuit design methodology. His program MOTIS was a significant development which led to a continued development in other places on many continents [11].

In the initial MOTIS-type programs, internal feedback of devices, although present, is assumed negligible. Thus, a relaxation-based matrix solution technique, such as the Gauss-Jacobi or the Gauss-Seidel method, can be used to achieve very fast circuit simulation. The key to the simplification is the taking of only one step of the relaxation iteration at each time point. The simplification provides accurate results if feedback is negligible. Of course, the assumptions may not hold well, and the simulation results may be inaccurate. It has been necessary, then, in timing simulation when greater accuracy is required to introduce methods to achieve more accurate simulation. A possible procedure at this stage is to move to a conventional circuit simulator. But the different I/O requirements have always been a major problem. Different programs developed by different groups or individuals evolve with nonidentical input and output formats. Input file preparation or translation is awkward and very time-consuming and cost-producing. The "standards" which have evolved by common usage are often not general enough to satisfy the needs for new programs or applications.

The study of numerical limitations of timing analysis has led to new types of accurate and fast circuit simulation. The relaxation iteration can be continued to convergence either in space (over the circuit nodes) or in time to achieve as accurate a solution as that provided by conventional circuit simulation employing the same electronic models [55]. The major advantage of relaxation-based approaches is the ability to exploit time sparsity, using the event-driven selective trace techniques first developed for logic simulators. For large circuits, over 90 percent of the transistors may not require re-solution at a time point. These new iterated timing analysis programs can be predicted to become the circuit simulators of the near future.

Even with recent improvements in the speed of circuit simulators, it is generally impractical to simulate entire VLSI circuits at the electrical level. Critical timing paths, usually containing less than 5000 transistors, are identified by the circuit designer and fed to a simulator. Unfortunately, errors can occur when the designer chooses the wrong timing path for simulation or ignores a complex interaction between a number of paths. Since the 1960's, digital logic designers have been using "timing verification" programs [36], [46], [45] to identify critical paths in complex logic systems such as main-frame computers. These programs assign a delay to each logic gate and perform a critical-path analysis of the network to identify potential problems. Recently, these techniques have been applied successfully at the circuit level as well [56]. Using simple RC models for rise and fall times at each circuit node, these timing verifiers can predict critical path delays to within 10 percent for constrained IC design styles. Once the critical paths have been identified, the circuits on the critical path are prepared for input to a circuit simulation program to check the timing predictions of the timing verifier. Initial results with timing verification at the circuit level are promising and timing verifiers will certainly become an important addition to the CAD toolbox.

Shortly after the development of timing simulation, mixed-mode or hybrid simulation emerged. In mixed-mode simulation, a portion of a large circuit is simulated, say in electrical activity, while simultaneously another portion of the circuit is being simulated in a different mode, say logic. For example, it is possible to have the majority of a large circuit being simulated in terms of logic levels while an interacting and related subportion of the complete circuit is being simulated in a timing or circuit simulation mode. With respect to logic and timing simulation, significant developments have occurred. In Leuven, DeMan and his coworkers [17], [18], and at Berkeley, Newton and his coworkers [51]–[53], have developed the DIANA and SPL-

ICE Programs, respectively. Of course, other levels and groupings of simulation can also be achieved. At CMU, Director and his students have developed the Program SAMSON [61], [62]. It is possible for two-dimensional device simulation to occur simultaneously with the circuit simulation of the connection of the devices [19]. In Europe, North America, and Japan, significant activities of mixed-mode simulation are in progress at most centers of CAD activity. These mixed-mode simulators are just coming into use in VLSI design.

X. Optimization

As brought out earlier, the application of the first circuit-analysis programs was for repeated analysis under control of another program to choose the best circuit element values to achieve overall specifications, i.e., an optimization package evolved. The initial work concerned the design of electrical filters, in the sinusoidal steady state. Work in computer optimization has continued through the years and significant results and conclusions have been achieved in the problem areas of design centering, tolerance optimization, and parasitic inclusion. In addition to the passive filter problem, design solutions for which have existed for some time, optimization programs have been modified and expanded to achieve the optimum design of new and important active filtering techniques.

Of course, there is a limit to what can be accomplished directly in the frequency domain. In spite of transform techniques (useful for linear circuits), optimization, tolerancing, and design-centering techniques are needed in the time domain directly and, of course, for nonlinear circuits. There is still a critical need for optimization efforts in the time domain in spite of the significant results to this point. The optimized design of nonlinear transient circuits is still very much an open question. Fortunately, because of the nature of binary, digital circuits in which the use of and design for two static states stops the propagation of errors throughout a circuit, effective design up to the present can be achieved solely on the basis of the proper inclusion of parasitics to estimate the significant timing paths.

An excellent recent review paper on circuit-design optimization points out how, in the past few years, significantly improved performance has been obtained in the design of large systems because of the improved performance of the simulation packages on the one hand and, even more importantly, the development of new techniques to specify the performance and cost functions to be optimized [8]. Performance results from these automated designs give a significant improvement over what has been able to be achieved by skilled IC designers. Another review paper oriented toward general systems is also available for the reader interested in the latest status on techniques [59].

XI. Summary

With the availability of the digital computer, whether relay or electronic, circuit and system investigators and designers have utilized these computers, and the programs which run on them, as tools to achieve circuit analysis and design. Circuit analysis with digital computers has been a steady activity throughout the last thirty years, and important developments have been accomplished. Although adequate circuit simulators are now available, still as needs have arisen and been identified, investigations have continued; new programs have been written, and new results are being achieved. Even at this point with the very capable tools available, one can see the emergence of iterative timing analysis, a new class of tool, which will aid the design of large-scale circuits and systems.

Circuit-design optimization was the driving force for the initial circuit-analysis programs. This topic remains a critical one in CAS to achieve optimum design performance, manufacturability, and tolerancing, particularly for nonlinear time-dependent circuits.

It is clear, as one reviews the continued development of computer aids for circuit and system design, there has been a significant effort, both of those working in theory and in computer program development. CAD is now one of the most significant and meaningful areas for those of us identified with the CAS Society. This aspect is recognized in many ways. First, the CAS Society has formed a new journal, the IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, now in its third year of publication. Further, one of the standing committees of the CAS is CANDE, concerned with all aspects of CAD. The CANDE committee sponsors an important workshop each year, and its members take an active role in organizing sessions at the annual ISCAS and at new conferences such as the 1982 ICCC and the 1983 ICCAD. It can be anticipated that the concentration of effort within the CAS Society on CAD, and the parade of new results, will continue into the foreseeable future.

Acknowledgment

Because of the very wide range of CAD activities by the members of the CAS Society, and because of the all-encompassing nature of CAD, it has been necessary in writing this paper to make several difficult choices. Many major aspects of CAD could have been stressed in place of those chosen. Even though some topics are of prime interest to other IEEE societies and, in a sense, are covered in the publications of these societies, still, many members of the CAS Society would have preferred a concentration on these topics. They, naturally, will be distressed. Even within the thread of CAD activity concerning computer-aided circuit analysis, it has been necessary to make difficult and awkward choices concerning what contributions and what decisions to stress. Shortly after this thread was chosen, a next major decision concerned a restriction to the CAD most important for large-scale circuits. This, then, did not permit me to mention many important topics of simulation such as pole-zero techniques, sensitivity and sensitivity calculation, the adjoint technique, the steady-state simulation of circuits including oscillators, etc. The omission of these topics within computer-aided circuit analysis will distress other CAS Society members. But in order to keep the paper in manageable size and topic for an historical

review, these decisions have been painfully made. These have been solely my responsibility. But I do wish to acknowledge the significant help that I received from the reviews and criticisms by I. E. Getreu, M. J. McCalla, A. R. Newton, and A. Sangiovanni-Vincentelli. I am pleased to acknowledge the support that has been given our CAD group at Berkeley by many agencies and firms. In particular, I wish to mention the financial support given over the years by the Army Research Office, lately under Grant DAAG29-81-K-0021, the National Science Foundation, recently under Grant ECS-8310442, and the Semiconductor Research Corporation. Industrial firms that have provided both technical cooperation and financial support include Bell Labs, Digital Equipment Company, Hewlett-Packard, Tektronix, and Texas Instruments. And finally, I acknowledge the tremendous contributions that have been made by the many research students and researchers in the many locations where circuit simulation has been developed. In many of these academic and industrial locations, users of our circuit simulation programs have been the willing or unwitting guinea pigs to use and to improve these software products.

REFERENCES

[1] M. R. Aaron, "The use of least squares in system design," *IRE Trans. Circuit Theory*, vol. CT-3, pp. 224–231, 1956.
[2] T. R. Bashkow, "The *A* matrix, new network description," *Trans. IRE Trans. Circuit Theory*, vol. CT-4, pp. 117–119, 1957.
[3] R. D. Berry, "An optimal ordering of electronic circuit equations or a sparse matrix solution," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 40–50, Jan. 1971.
[4] F. H. Branin, Jr., "The relation between Kron's method and the classical methods of network analysis," in *IRE WESCON Conv. Rec.*, pt. 2, pp. 3–28, 1959. Also available as Tech. Rep. 00.686, IBM Development Laboratory, Poughkeepsie, NY, 1959.
[5] F. H. Branin, Jr., "DC analysis portion of PETAP-A program for analyzing transistor switching circuits," IBM Development Lab., Poughkeepsie, NY, Tech. Rep. 00.701, 1960.
[6] F. H. Branin, Jr., "DC and transient analysis of networks using a digital computer," in *IRE Int. Conv. Rec.*, pt. 2, pp. 236–256, Mar. 1962.
[7] F. H. Branin, Jr., "Computer methods of network analysis," *Proc. IEEE*, vol. 5, pp. 1787–1801, Nov. 1967.
[8] R. K. Brayton, G. D. Hachtel, and A. L. Sangiovanni-Vincentelli, "A survey of optimization techniques for integrated-circuit design," *Proc. IEEE*, vol. 69, no. 10, pp. 1334–1361, Oct. 1981.
[9] N. G. Brooks and H. S. Long, "A program for computing the transient response of transistor switching circuits—PETAP," IBM Development Lab., Poughkeepsie, NY, Tech. Rep. 00.700, 1959.
[10] D. A. Calahan, "Multi-level vectorized sparse solution of LSI circuits," in *Proc. IEEE Int. Conf. Circuits and Computers*, New York, Oct. 1980.
[11] B. R. Chawla, H. K. Gummel, and P. Kozak, "MOTIS—An MOS timing simulator," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 901–909, Dec. 1975.
[12] K. A. Chen, M. Feuer, K. H. Khokhani, N. Nan, and S. Schmidt, "The chip layout problem: An automatic wiring procedure," in *Design Automation Conf. Proc.*, pp. 298–302, June 1977.
[13] E. Cohen, "Program reference for SPICE2," Univ. of California, Berkeley, ERL Memo ERL-M592, June 1976.
[14] E. Cohen, "Performance limits of integrated circuit simulation on a dedicated minicomputer system," Univ. of California, Berkeley, ERL Memo UCB/ERL M81/29, May 1981.
[15] C. Cole, "Attaching an array processor in the UNIX environment," Univ. of California, Berkeley, ERL Memo UCB/ERL M83/23, April 1983.
[16] H. DeMan, "Computer aided design for integrated circuits: Trying to bridge the gap," *IEEE J. Solid-State Circuits*, vol. SC-14, pp. 613–621, June 1979.
[17] H. DeMan *et. al.*, "DIANA: Mixed mode simulator with a hardware description language for hierarchical design of VLSI," in *IEEE ICCC'80 Conf. Proc.*, Rye, NY, pp. 356–360, Oct. 1980.
[18] H. DeMan, G. Arnout, and P. Reyneart, "Mixed mode circuit simulation techniques and their implementation in DIANA," in *Computer Design Aids for VLSI Circuits*, P. Antognetti, D. O. Pederson, and H. DeMan, Eds. Groningen, the Netherlands: Sijthoff & Noordhoff, 1980, pp. 113–174.
[19] W. L. Engl, R. Laur, and H. K. Dirks, "Medusa—A simulator for modular circuits," *IEEE Trans. Computer-Aided Design of ICAS*, vol. CAD-1, no. 2, pp. 85–93, Apr. 1982.
[20] B. Epler, Texas Instruments Co., private communication, 1982.
[21] A. Feller, A. Smith, P. Ramondetta, and T. Lombardi, "High speed CMOS/SOS using standard cells," RCA Engineer, pp. 36–40, Dec. 1974/Jan. 1975.
[22] C. W. Gear, "Numerical integration of stiff ordinary differential equations," Dept. Comput. Sci., Univ. of Illinois, Urbana, Rep. 221, 1967.
[23] R. S. Graham, "Relay computer for network analysis," *Bell Labs. Rec.*, vol. 31, pp. 152–157, Apr. 1953.
[24] R. S. Gyurcsik, D. Cheung, F. Ma, T. Yee, and D. O. Pederson, "BIAS-B/P: circuit simulators for desktop computers," in *IEEE Proc. 1983 Custom Integrated Circuits Conf.*, pp. 354–358, May 1983.
[25] G. D. Hachtel, R. K. Brayton, and F. G. Gustavson, "The sparse tableau approach to network analysis and design," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 101–113, Jan. 1971.
[26] G. D. Hachtel and A. L. Sangiovanni-Vincentelli, "A survey of third-generation simulation techniques," *Proc. IEEE*, vol. 69, pp. 1264–1280, Oct. 1981.
[27] S. D. Hamm and S. R. Beckerich, "VAMOS: Circuit simulation program for a vector computer," in *Proc. ICCAD Conf.*, Sept. 1983.
[28] C. W. Ho, A. E. Ruehli, P. A. Brennan, and D. A. Zein, "Interactive circuit analysis and design using APL," in *Proc. 1975 ISCAS*, pp. 216–219.
[29] C. Ho, A. E. Ruehli, and P. A. Brennan, "The modified nodal approach to network analysis," *IEEE Trans. Circuits Syst.*, vol. CAS-25, pp. 504–509, June 1975.
[30] "Automated digital computer program for determining responses of electronic systems to transient nuclear radiation (PREDICT)," IBM Space Guidance Center, Oswego, NY, IBM File 64-521-5, July 1964.
[31] "1620 electronic circuit analysis program [ECAP]," ['1620-EE-02X] User's Manual, IBM Application Program File H20-0170-1, 1965.
[32] "ASTAP—Advanced statistical analysis program," IBM Program Product Document SH20-1118-0, IBM Data Processing Div., White Plains, NY, 1973.
[33] T. E. Idleman, F. S. Jenkins, W. J. McCalla, and D. O. Pederson, "SLIC-A simulator for linear integrated circuits," *IEEE J. Solid-State Circuits*, vol. SC-6, pp. 188–204, Aug. 1971.
[34] F. S. Jenkins and S. P. Fan, "TIME—A nonlinear dc and time-domain circuit simulation program," *IEEE J. Solid State Circuits*, vol. SC-6, pp. 188–192, Aug. 1971.
[35] E. D. Johnson, C. T. Kleiner, L. R. McMurray, E. L. Steele, and F. A. Vassallo, "Transient radiation analysis by computer program (TRAC)," Autonetics Div., North American Rockwell Corp., Anaheim, CA, Tech. Rep. issued by Harry Diamond Labs., June 1968.
[36] T. I. Kirkpatrick and N. R. Clark, "PERT as an aid to logic design," *IBM J. Res. Develop.*, pp. 135–141, Mar. 1966.
[37] G. Kron, "A set of principles to interconnect the solutions of physical systems," *J. Appl. Phys.*, vol. 24, pp. 965–980, 1953.
[38] E. S. Kuh and R. A. Rohrer, "The state-variable approach to network analysis," *Proc. IEEE*, vol. 53, pp. 672–686, July 1965.
[39] W. Lattin, "VLSI design methodology: The problem of the 80's for microprocessor design," in *Proc. 16th Design Automation Conf.*, (San Diego, CA), pp. 548–549, June 1979.
[40] A. F. Malmberg, F. L. Cornell, and F. N. Hofer, "NET-1 network analysis program," Los Alamos Scientific Lab., Los Alamos, NM, Tech. Rep. LA-3119, 7090/94 Version, Aug. 1964.
[41] H. M. Markowitz, "The elimination form of the inverse and its application to linear programming," *Management Sci.*, vol. 3, pp. 255–269, Apr. 1957.
[42] H. W. Mathers, S. R. Sedore, and J. R. Seuts, "Automated digital computer program for determining responses of electronic circuits to transient nuclear radiation (SCEPTRE)," IBM Space Guidance Center, Oswego, NY, IBM File 66-928-611, Feb. 1967.
[43] W. J. McCalla and D. O. Pederson, "Elements of computer-aided analysis," *IEEE Trans. Circuit Theory*, vol. CT-18, pp. 14–26, Jan. 1971.
[44] W. J. McCalla and W. G. Howard, Jr., "BIAS-3: A program for the nonlinear dc analysis of bipolar transistor circuits," *IEEE J. Solid-State Circuits*, vol. SC-6, no. 1, pp. 14–19, Feb. 1971.
[45] J. McClure, "Frequency domain simulation of very large digital systems," in *Proc. IEEE Int. Symp. Circuits and Systems*, Houston, TX, pp. 424–430, Apr., 1980.
[46] T. M. McWilliams, "Verification of timing constraints in large digital systems," in *Proc. IEEE Int. Symp. Circuits and Systems*, Houston, TX, pp. 415–423, Apr., 1980.
[47] L. Nagel and R. A. Rohrer, "Computer analysis of nonlinear circuits, excluding radiation (CANCER)," *IEEE J. Solid-State Circuits*, vol. SC-6, pp. 166–182, Aug. 1971.
[48] L. W. Nagel and D. O. Pederson, "Simulation program with in-

tegrated circuit emphasis," in *Proc. 16th Midwest Symp. Circ. Theory*, Waterloo, Canada, Apr. 1973.
[49] L. Nagel, "SPICE2: A computer program to simulate semiconductor circuits," Univ. of California, Berkeley, ERL Memo UCB/ERL M75/520, May 1975.
[50] A. R. Newton and G. L. Taylor, "BIASL.25 An MOS circuit simulation program for a programmable calculator," in *Tenth Annual Asilomar Conf. Circuits, Systems, and Computers*, pp. 280–283, Nov. 1976.
[51] A. R. Newton, "The simulation of large-scale integrated circuits," Univ. of California, Berkeley, ERL Memo No. ERL-M78/52, July 1978.
[52] ____, "Techniques for the simulation of large-scale integrated circuits," IEEE Trans. *Circuits Syst.*, vol. CAS-26, pp. 741–749, Sept. 1979.
[53] A. R. Newton, "Timing, logic, and mixed mode simulation for large MOS integrated circuits," in *Computer Design Aids for VLSI*, P. Antognetti, D. O. Pederson, and H. DeMan, Eds. Groningen, The Netherlands: Sijthoff & Noordhoff, 1980, pp. 175–240.
[54] A. R. Newton, "A survey of computer aids for VLSI layout," in *1982 Symp. on VLSI Technology*, no. 82, pp. 72–75, Sept. 1982.
[55] A. R. Newton and A. L. Sangiovanni-Vincentelli, "Relaxation based electrical simulation," *IEEE Trans. Electron Devices*, vol. ED-30, Sept. 1983, and *IEEE Trans. Computer-Aided Design of ICAS*, to be published.
[56] J. K. Ousterhout, "CRYSTAL: A timing analyzer for *n*MOS VLSI circuits," in *3rd Cal. Tech. Conf. on VLSI*, pp. 57–70, 1983.
[57] G. Persky, D. N. Deutsch, and D. G. Schweikert, "LTX—A system for the directed automation design of LSI circuits, in *Proc. 13th Design Automation Conf.*, 1976.
[58] G. Persky, D. N. Deutsch, and D. G. Schweikert, "LTX—A minicomputer-based system for automatic LSI layout," *J. Design Automation and Fault-Tolerant Computing*, vol. 1, no. 3, pp. 217–255, May 1977.
[59] E. Polak, "Semi-infinite optimization in engineering design," in *Lecture Notes in Economics and Mathematical Systems*, vol. 215, *Semi-Infinite Programming and Applications*, A. V. Fiacco and K. O. Kortanek, Eds., Berlin, New York, Tokyo: Springer-Verlag, 1983.
[60] N. B. Rabbat, A. L. Sangiovanni-Vincentelli, and H. Y. Hsieh, "A multilevel newton algorithm with macromodeling and latency for analysis of large-scale nonlinear networks in the time domain," *IEEE Trans. Circuits Systs.*, vol. CAS-26, Sept. 1979.
[61] K. A. Sakallah and S. W. Director, "An event driven approach for mixed gate and circuit level simulation," in *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 1194–1197, May 1982.
[62] K. A. Sakallah and S. W. Director, "SAMSON: A mixed circuit logic level simulator," to be published.
[63] R. A. Saleh, J. E. Kleckner, and A. R. Newton, "Iterated timing analysis in SPLICE1," in *Proc. ICCAD Conf.*, Sept. 1983.
[64] I. W. Sandberg and H. Shichman, "Numerical integration on systems of stiff nonlinear differential equations," *Bell Syst. Tech. J.*, vol. 47, pp. 511–527, Apr. 1968.
[65] A. L. Sangiovanni-Vincentelli, L-K. Chen, and L. O. Chua, "A new tearing approach—Node-tearing nodal analysis," in *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 143–147, Apr. 1977.
[66] A. L. Sangiovanni-Vincentelli, "Circuit simulation," in *Computer Design Aids for VLSI Circuits*, P. Antognetti, D. O. Pederson, and H. DeMan, Eds. The Netherlands: Sijthoff & Noordhoff, 1980, pp. 19–112.
[67] L. J. Shanbeck and R. S. Norin, "QSPICE: An application of array processors to CAD simulation of IC circuits," in *Proc. IC-CAD Conf.*, Sept. 1983.
[68] H. Shichman, "Computation of dc solutions for bipolar transistor networks," *IEEE Trans. Circuit Theory*, vol. CT-16, pp. 460–466, Nov. 1969.
[69] ____, "The integration system of a nonlinear network-analysis program," *IEEE Trans. Circuit Theory*, vol. CT-17, pp. 378–386, Aug. 1970.
[70] J. Soukup, "Circuit Layout," *Proc. IEEE*, vol. 69, pp. 1281–1304, Oct. 1981.
[71] T. Sudo, T. Ohtsuki, and S. Goto, "CAD systems for VLSI in Japan," *Proc. IEEE*, vol. 71, Jan. 1983.
[72] A. Vladimirescu and D. O. Pederson, "A computer program for the analysis of LSI circuits," in *Proc. IEEE Int. Symp. on Circuits and Systems*, Chicago, IL, Apr. 1981.
[73] A. Vladimirescu, "LSI circuit simulation on vector computers," Univ. of California, Berkeley, ERL Memo, UCB/ERL M82/75, Oct. 1982.
[74] W. T. Weeks *et al.*, "Algorithms for ASTAP—A network analysis program," *IEEE Trans. Circuit Theory*, vol. CT-20, pp. 628–634, Nov. 1973.
[75] P. Yang, "An investigation of ordering, tearing, and latency algorithms for the time-domain simulation of large circuits," Univ. of Illinois, Urbana, Tech. Rep. R-891, Aug. 1980.
[76] D. A. Zein, C. W. Ho, and A. J. Gruodis, "A new interactive circuit design program," in *Proc. IEEE Int. Symp. Circuits and Systems*, Houston, TX, pp. 913–917, 1980.

Donald O. Pederson (S'49–A'51–M'56–F'64) received the B.S. degree from North Dakota State University in 1948, and the M.S. and Ph.D. degrees from Stanford University in 1949 and 1951, respectively.

From 1951 to 1953 he was a Research Associate in the Electronics Research Laboratory, Stanford University. He worked at Bell Telephone Laboratories, Inc. and was also a lecturer at New Jersey Institute of Technology from 1953 to 1955. In 1955 he joined the University of California, Berkeley. He is now a Professor in the Department of Electrical Engineering and Computer Sciences, engaged in research in integrated circuits and computer-aided integrated circuit design, and is the Chairman of the Department. From 1960–1964 he was Director of the Electronics Research Laboratory.

Dr. Pederson is a member of Sigma Xi and Eta Kappa Nu. In 1964 he was a Guggenheim Fellow. He was the recipient of the 1969 IEEE Education Medal. In 1974 he was elected to the National Academy of Engineering. He was awarded an honorary Doctor of Applied Science by the Katholieke Universiteit, Leuven, Belgium, in 1979. In 1982 he was elected to the National Academy of Sciences.