EES SERIES REPORT NO. 12

PROGRESS OF ANALOG-HYBRID COMPUTATION

1966

ENGINEERING EXPERIMENT STATION
COLLEGE OF ENGINEERING
THE UNIVERSITY OF ARIZONA
TUCSON, ARIZONA

Granino Arthur Korn was born in Berlin, Germany, on May 7, 1922. He was graduated from Brown University, Providence, R. I., in 1942, and received the M.A. and Ph.D. degrees in physics from Columbia University, New York, N. Y., and Brown in 1943 and 1948, respectively.

After wartime service in the U. S. Navy, he was, successively, a Project Engineer with Sperry Gyroscope, Great Neck, N. Y., head of the analysis group at Curtiss-Wright/Columbus, a staff engineer with Lockheed/Burbank, and a consultant under his own name. Since 1957, he has been a Professor of Electrical Engineering at the University of Arizona, Tucson, where he teaches courses on electronic computers and random-process measurements; he is also a Director of Computer Research, Inc., Lafayette, Calif.

With his wife, Theresa M. Korn, he is the author of *Electronic Analog Computers* (1952 and 1956), *Mathematical Handbook for Scientists and Engineers* (1961), *Electronic Analog and Hybrid Computers* (1964), *Basic Tables for Electrical Engineering* (1965), *Random-Process Simulation and Measurements* (1966), and *Manual of Mathematics* (1966). He is co-editor-in chief of the McGraw-Hill *Computer Handbook* (1962) and the *Digital Computer User's Handbook* (1967) and has contributed articles to seven other engineering handbooks. He edits Mc-Graw-Hill's new series of short numerical tables and is chairman of the editorial board of *Simulation*. He has taught in Mexico and surveyed engineering education in Chile as a Consultant to the National Academy of Sciences.

Dr. Korn is a member of Sigma Xi, Simulation Councils, Inc., and the International Analog Computer Association.

Progress of Analog/Hybrid Computation

G. A. KORN

Abstract—Solid-state circuits and the integrated circuits succeeding them are improving analog-computing-element reliability by an order of magnitude, speed by two orders of magnitude, and still reducing system cost. The effects of integrated-circuit technology on the price, reliability, and sophistication of digital computers are even more pronounced, and the resulting system-design tradeoffs shape the future course of hybrid-computer techniques. This review article discusses the design of fast analog/hybrid computer systems, including recent developments in integrated operational amplifiers, electronic mode-control switches, digital attenuators, and new packaging techniques.

SURVEY

COMPUTERS are physical systems designed to implement the idealized relationships of mathematical models. Computers are conventionally classified as *analog computers* if the machine variables representing mathematical quantities are considered to vary continuously (voltages, shaft rotations), while a *digital computer* is a discrete-state system. The engineer's most noticeable impression of an analog computer is likely to be a fast, only moderately accurate differential-equation solver (differential analyzer) employing separate (parallel) analog computing elements to implement each mathematical relation; while the most important digital computers perform numerical operations accurately and sequentially and incorporate considerable decision-making ability and a large memory to permit stored-program control of computation and iterative operations. Advancing technology has evolved the familiar analog computer into a wide class of *hybrid-analog-digital computers* which may combine high analog computing speed with accurate digital operations, decision making, and memory.

Computers, analog, digital, or hybrid, have two principal applications. They may set up mathematical relations to produce desired unknown or otherwise desired quantities and/or functions (problem solving, simulation of engineering systems). The second (and at least as important) application of computing devices is as *system components* used for instrumentation, control, and display: here, the computer forces at least a portion of a reluctant physical system to act like a mathematical model and thus recreates it nearer to the heart's desire.

Up to 1960, the simulation of dynamic systems such as aircraft, missiles, space vehicles, and nuclear and chemical plants was accomplished almost entirely by means of electronic analog computers. The parallel computing elements of such machines permit high-speed real-time solution of differential equations, and establish a block-diagram-oriented, live mathematical model offering a good deal of intuitive insight into system operation. "Hands on" changes of system parameters, programs, and subsystems are readily possible with analog computers; solutions can be observed on multi-trace and xy recorders to provide the investigator with immediate feedback about the results of system changes. Analog-computer studies of large dynamic systems also show up significant disadvantages of this type of computation; the relatively low component accuracies permit long-term trajectory studies only if sophisticated mathematical perturbation techniques are applicable, and such techniques destroy much of the intuitive insight of direct

analog computing elements
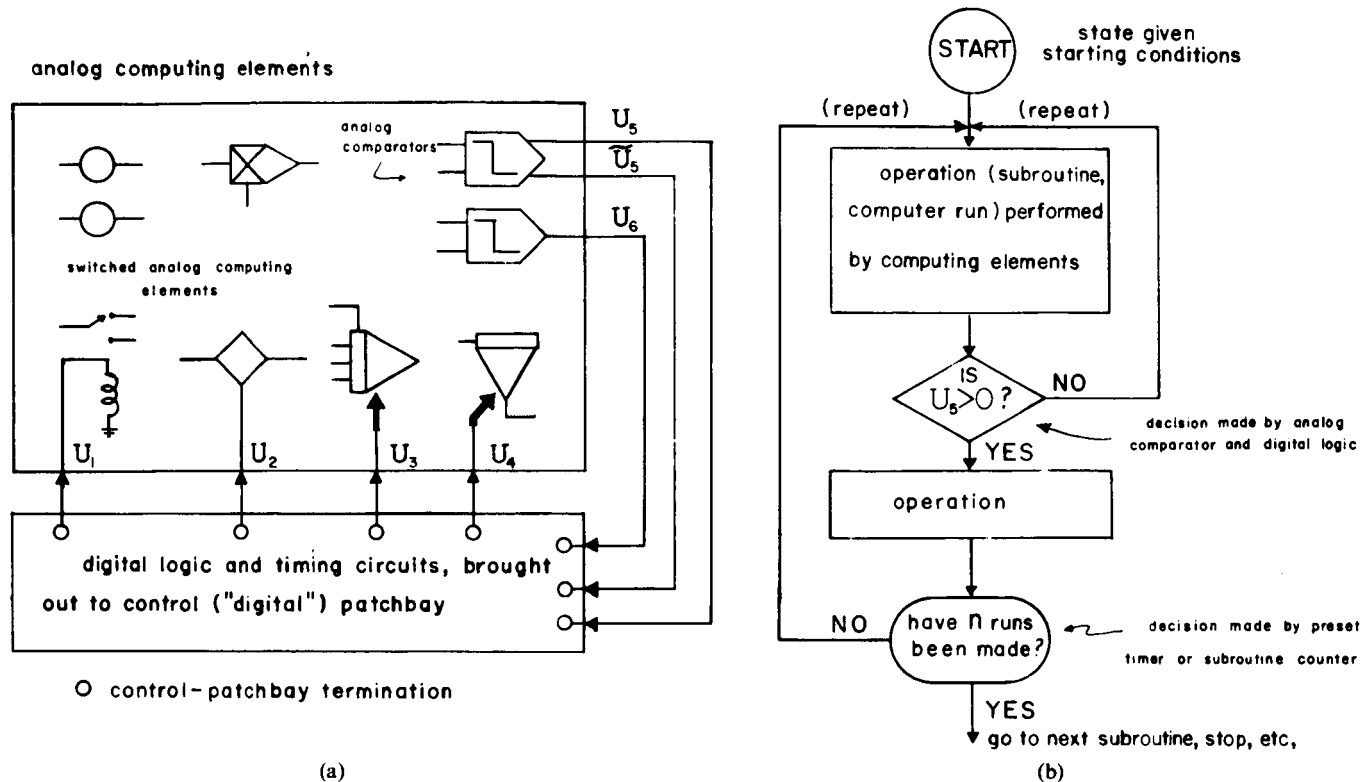

(a)

state given
starting conditions

(b)

Fig. 1. (a) Iterative-differential-analyzer operation. (b) Flow diagram. The principal operations are analog-computer (differential-analyzer) runs simulating a physical system, as ordered by electronic switches actuated by corresponding digital control variables U_i. Control-variable changes may be preset by timers and counters and/or "computed" by analog comparators and digital logic. Switches and comparators form the digital-analog and analog-digital interfaces. Typical applica-

tions are to iterative optimization of the simulated system and to statistical computations. A small or large general-purpose digital computer may be added and will communicate with the analog computer via A/D and D/A converters, and with the control logic via interrupt and sense lines and by digital commands fed into the control patchbay [19].

simulation. Moreover, scale factor estimation and problem checkout can become very cumbersome.

Improved simulation techniques combining analog and digital computation take two essential forms, which are often used in combination. An *iterative differential analyzer* is an electronic analog computer incorporating analog memory in the form of track-hold circuits and equipped with analog comparators, patchable digital logic elements, and comparator- or logic-operated *D/A* switches, and timing circuits (Fig. 1).

Such machines have, thus, a measure of the memory and decision-making ability of a digital computer; integrator mode control and, to some extent, analog-computer programs and system parameters can be changed by the digital logic which, in turn, takes account of analog-comparator decisions. This permits complex subroutine control, branching, simulation of digital computers, multiplexing of computing elements, and, above all, automatic iteration and system optimization. The iterative differential analyzer is at its best in its more recent ultra-high-speed versions permitting iteration of as many as 2000 complete system simulations per second under digital-logic control. Such fast iterative computations permit rapid computation of statistics (Monte Carlo methods), so that the system designer receives practically immediate feedback of the effects of system parameter changes *on statistics*, and automatic optimization of statistics becomes possible (Fig. 2) [2].

True *combined analog-digital simulation* with an analog computer (usually an iterative differential analyzer) joined to a general-purpose digital computer through a converter linkage [1] has permitted real-time simulation of large and very complicated systems. The digital computer is, in particular, employed for accurate and relatively slow trajectory integration, while the analog computer deals with the high-speed dynamics; in addition, the digital computer can conveniently store and/or compute complicated functions of several variables and, also, perform many housekeeping tasks such as potentiometer setup, static checking, and the computation of accurate check solutions. Combined simulation is often made mandatory by stringent combinations of speed and accuracy requirements. But it should be remembered that combined simulation joins not only the advantages of analog and digital computers but their disadvantages as well, for the simulation suffers both from the low precision of the analog computer and the sampled-data bandwidth limitation of the digital machine. In addition, large combined systems tend to be complex and very expensive, and the difficult and sophisticated programming required destroys much of the intuitive appeal of the simulation.

By contrast, *the use of one of the increasingly inexpensive small (12- to 16-bit) general-purpose data processors as an—immeasurably improved—logic and control unit for a fast iterative differential analyzer has proved to be an extra-*

$$\overline{\langle e^2(t)\rangle_T} = \frac{1}{n}\sum_{k=1}^{n}\langle{}^k e^2(t)\rangle_T = F(\alpha_1,\alpha_2,\alpha_3)$$
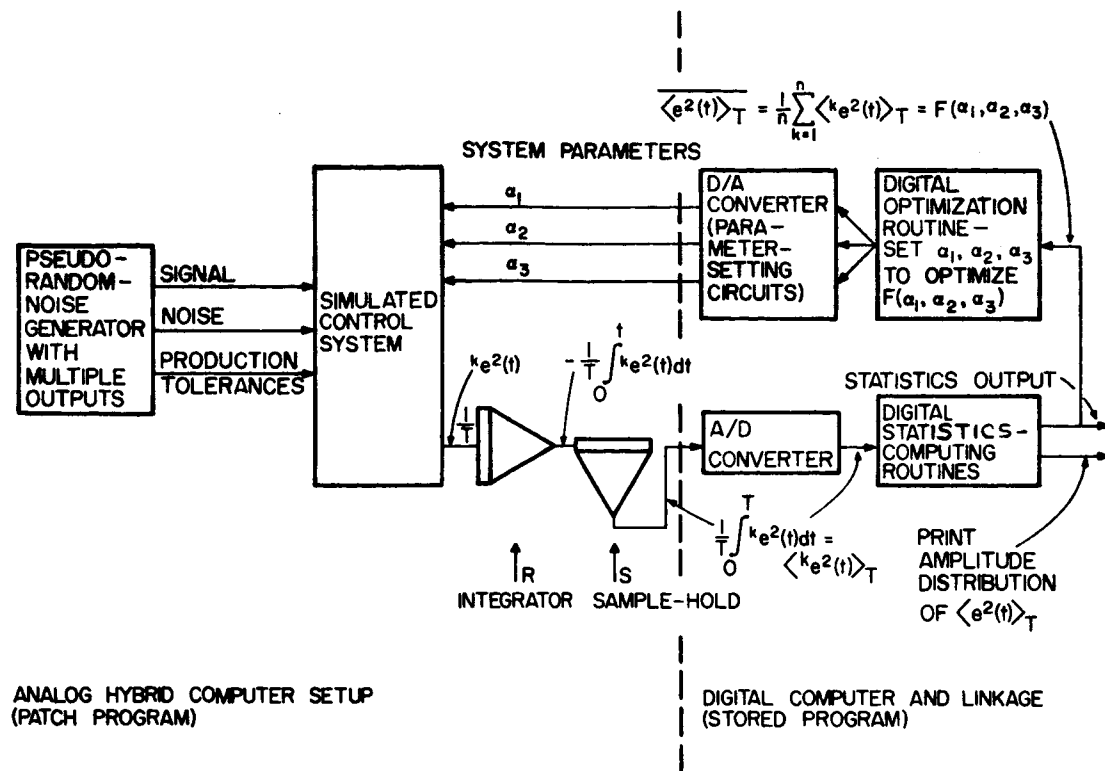
SYSTEM PARAMETERS

Fig. 2. This example illustrates the possibilities of combining a fast iterative analog computer with a small stored-program digital data processor. Analog computing elements produce successive samples of the squared-error time average

$$\langle{}^k e^2(t)\rangle_T = (1/T)\int_0^T {}^k e^2(t)\,dt$$

for a simulated control system with random input and noise. The digital computer finds successive values of the sample average

$$\overline{\langle{}^k e^2(t)\rangle_T} = (1/n)\sum_{k=1}^{n}\langle{}^k e^2(t)\rangle_T$$

for $n = 100$ to 10 000 computer runs and changes control-system parameters so as to optimize the sample average. Note that each analog-digital and digital-analog conversion and digital-computation sequence is required only once per analog-computer run (from G. A. Korn, "Hybrid-computer Monte-Carlo techniques," *Proc. IFIP Cong.*, New York, May 1965; see also [2]).

ordinarily successful combination of analog and digital computers. The small digital computer performs timing and mode-control functions, computes statistics, and optimizes parameter combinations; even with 2000 analog-computer runs per second, most digital-computer operations take place only once per analog-computer run, so that the digital computer can perform very sophisticated logic and statistical operations at a comfortable rate. At the same time, the small digital computer again performs housekeeping functions such as the setting of potentiometers (more recently replaced by digital attenuators), static checks, and various routine computing chores around the simulation laboratory.

The essential components of electronic analog computers are resistor-capacitor-diode networks and operational amplifiers; as electronic-analog/hybrid computation enters its third decade, a technical revolution has made the use of vacuum tubes in analog computers as wholly obsolete as it is in portable radios or in digital computers. The advent of monolithic integrated circuits—linear as well as digital— and new field-effect devices promises to decrease prices of

operational amplifiers by a factor of three while frequently eliminating the need for chopper stabilization and extending attainable bandwidths to tens of megacycles. At the same time, mean-time-between-failures has increased by at least an order of magnitude, while integrated-circuit miniaturization further reduces computer cost by eliminating much interunit wiring. The new technology is beginning to produce reliable, exceptionally flexible, very fast iterative differential analyzers in the medium-size 50 to 100 amplifier class; they are well equipped with modern nonlinear computing elements, digital logic, and all-electronic mode control. Somewhat smaller machines are beginning to make a pronounced impact on modern engineering education: as an example, the University of Arizona, Tucson, in the last quarter of 1966, had 29 analog computers distributed over at least six departments; all engineering students learn FORTRAN and analog-computer simulation in a special course integrated with their progress in advanced calculus and ordinary differential equations.

The development of analog and hybrid-analog/digital computation is intimately bound up with the corresponding

progress of digital computation. The integrated-circuit revolution is producing excellent flexible and fast digital computers whose price tags read below $20 000; at the same time, larger digital computers are conveniently time-shared by many users through multiple stations. Coupled with ever-increasing digital-computer speeds, both methods increasingly permit direct intercourse between man and digital computer, an advantage heretofore reserved for analog computers. While the cost of both small digital computers and time-shared large installations is continually decreasing, improved graphical displays and programming consoles, together with the freedom from analog scaling afforded by floating-point hardware, make digital simulation increasingly attractive. The replacement of many large analog and even hybrid-analog-digital simulations, such as three-dimensional flight simulations, by new all-digital simulators has already overtaken training-type flight simulators and is proceeding into the engineering departments of larger aircraft companies. The coming breakthrough of digital dynamic-system simulation will require design effort on more powerful human-factor engineered display consoles permitting instant interaction between operator and machine, improved numerical techniques, and multiple digital processing with the aid of special-purpose digital computing elements implementing time-consuming subroutines. The perfection of all these techniques awaits only the pleasure of the digital-computer manufacturers, for whom simulation represents only a relatively small fraction of the market. In the meantime, a number of new digital *block-diagram interpreter and compiler programs* permit simulation of an analog computer with a conventional digital computer.

A common feature of iterative-differential-analyzer operation, digital simulation, and combined analog-digital simulation is an ever-increasing need for more sophisticated mathematical techniques and, in consequence, for better-educated investigators. The use of computers for excessively complete simulation can rarely substitute for mathematical insight into a problem; automatic system optimization requires knowledge of an increasing body of mathematical theory; perturbation techniques permit the attainment of astonishing accuracies with simple analog computers; combined simulation requires a knowledge of sampled-data theory; and effective utilization of the computer consoles now within easy reach of every researcher or system designer will require much re-education of our modes of thinking in research and system design.

REQUIREMENTS FOR REALLY FAST ANALOG COMPUTATION

While "slow" analog computation at signal frequencies below about 100 Hz still constitutes the bulk of simulation work, at least the larger "slow" analog computers will tend to be displaced by new general-purpose digital computers as the latter improve. Even at present, modern hybrid-computer installations tend to replace every possible analog operation with digital routines to save maintenance and scaling; one might say that the analog-digital interface of existing hybrids is moving so that more and more of the

"slow" analog equipment is associated with crew-station simulation and other partial system tests. The most fruitful applications of really fast iterative analog computation (iterative rates >200 Hz, signal frequencies >2 kHz) is to *statistical applications* (Monte Carlo simulation), *to rapid system optimization, and to combinations of these (Fig. 2). The real advantage of fast computation is not the mass production of data (this could be done by a digital computer working at night), but the possibility of direct on-line-operator interaction with the results from, say, a sample of 1000 computer runs completed in a second or less.*

As a matter of competitive advertising, most commercially available analog/hybrid computers offer repetitive operation at 1000 computer runs per second, *but this is usually wholly misleading*. It is easy to supply computer-reset pulses at 1 kHz, but meaningful computation at such iteration rates requires, at least, component accuracies within 0.2 percent at signal frequencies of 5 to 10 kHz. It is instructive to investigate just what such a requirement means. Before one can even think of dealing with possible amplitude errors, it is necessary to control errors due to phase shift and switching time delays [1]. The effects of phase shift in amplifiers and passive networks and those of time delays caused by late operation of integrator and track-hold mode-control switches and analog comparators cause quite similar delay errors, as illustrated in Fig. 3. Given a signal waveform changing at the rate $2\pi a$ (typically a sinusoidal voltage $a \cos 2\pi ft$ of frequency f Hz), a timing error of Δt seconds, or an equivalent phase-shift error, $\Delta\phi$ radians cause the true percentage error

$$\varepsilon = 200\pi f \Delta t = 100 \Delta\phi. \tag{1}$$

The "magic equation" (1) spells the doom of high-frequency computation with most commercially made equipment. Consider the very lenient requirement of 0.2 percent error at a computing frequency of 5 kHz. This requires:

1) Phase-shift errors $\Delta\phi$ below 0.002 radians or about 0.12 degrees at 5 kHz. Even in a unity-gain phase inverter, this requires amplifiers with a zero-dB crossover frequency of 10 MHz with a 20 dB/octave rolloff, and perhaps 2 MHz for a less conservative rolloff. The situation becomes exactly ten times worse for a gain-of-10 phase inverter, not to speak of multipliers and function generators. Nevertheless, such requirements can be met with $\pm$ 10-V feedforward amplifiers (described further in the following) and clever equalization.

2) The above phase-shift requirements extend to all passive networks, attenuators, and interconnection lines (if any) in the computer. Distributed capacitances make the 100-K resistors used in commercial analog computers act like veritable delay lines at high frequencies. While some of these ills can be cured with equalizing capacitors, probably the best solution is the use of 1-K to 10-K summing resistors and similarly low impedance levels in all attenuators. Solid-state computing elements can be mounted directly behind
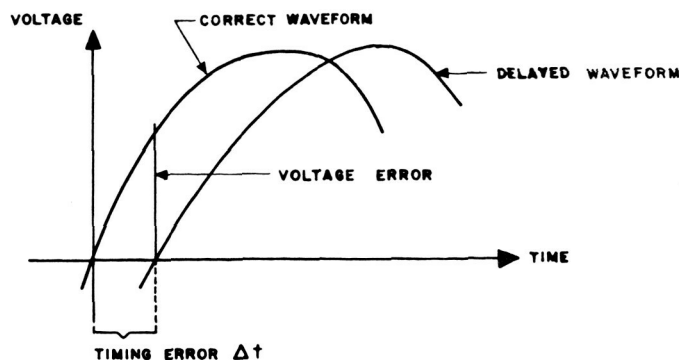
Fig. 3.   Timing errors Δt or phase errors $\Delta\phi = 2\pi f \Delta t$
are crucial in fast analog computation.

Fig. 4.   Operating with signal frequencies as large as 60 kHz, the digitally controlled ASTRAC II here displays the complete input-output cross-correlation function of a fast-time simulated control system with random input once every second. The operator can immediately assess effects of system-parameter changes on statistics computed from hundreds of analog-computer runs.

the analog-computer patchbay without any interconnecting lines.

3) A glance at the "magic equation" (1) shows clearly that 0.2 percent accuracy at 5 kHz requires all integrator mode-control switches to initiate computation within 65 ns of one another. Such *differential* switching times for integrators and track-hold circuits are possible only if the total switching time is below about 200 ns.

4) If an analog comparator or D/A switch must respond in the course of a computer run, its *total* (not differential) response time must be within 65 ns. Fortunately, most fast iterative computation requires comparators to respond to track-hold outputs only, so that timing does not cause errors.

A frequently redeeming feature is the fact that the highest signal-frequency components constitute only, say, 10 percent of the total analog signal, so that the total percentage error is reduced. In any case, (1) indicates that some commercially available analog computers can compute with fair accuracy at repetition rates of the order of 50 to 200 Hz. The ASTRAC II and LOCUST iterative analyzers (Fig. 4) developed at the University of Arizona under NASA and U. S. Air Force sponsorship do permit 1 kHz iteration; unity-gain-inverter phase shift is below 0.1 deg at 10 kHz, with integrator differential switching times below 20 ns, and comparator response time below 80 ns.

SOME HYBRID-COMPUTER SYSTEM CONSIDERATIONS

Since their inception about a decade ago, combined analog-digital computer systems have developed from true monstrosities into reasonably finished hardware; software and programming know-how are still lagging component
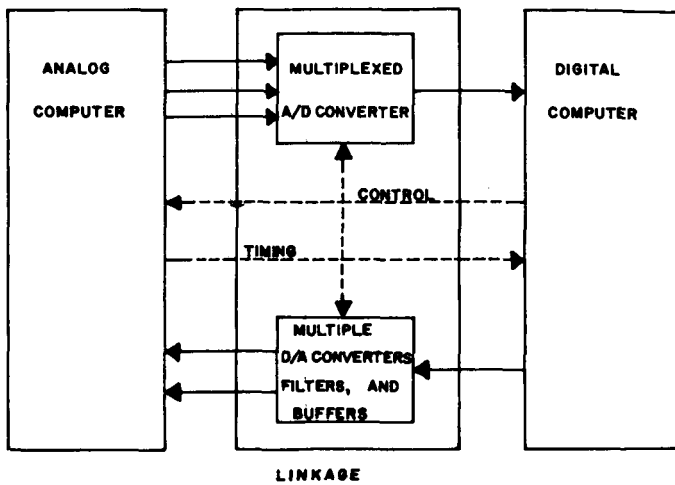
LINKAGE

Fig. 5. Simplified hybrid-computer loop.

developments. In addition to the usual computing errors inherent in analog and digital computers as such, the hybrid-computer system (Fig. 5) adds some errors of its own; these error sources are nicely discussed in [3]–[7]. Briefly, the errors due to hybrid-computer operation as such are:

1) *Conversion errors* due to poor A/D and D/A converters.

2) *Quantization errors* due to the finite number of digital bits; with the usual 13- to 14-bit systems, these are rarely serious and can often be further reduced through the application of dither [2]. Reference [6] presents a sophisticated theory for quantization-error estimation, but the best estimates are obtained by digital-computer test routines which simply reduce the number of bits used for a test run.

3) *Errors due to digital processing and conversion time delays.* While elaborate studies of such errors by means of Z-transform analysis are possible at least for linear systems [4], [5], we can simply consider the entire digital computer and linkage (Fig. 5) as an analog computing element having a time-delay error Δt; then the "magic equation" (1) estimates the component error, which in each hybrid-computer setup affects the solution very much like a phase-shift error $\Delta\phi = 2\pi f \Delta t$ for sinusoidal components of frequency f.

4) *Skewing errors* due to nonsimultaneous analog-digital and/or digital-analog conversion of different computer variables.

To minimize the effects of the various timing errors, one makes the processing delay Δt constant through suitable interruptions of the digital computation. Partial compensation of the time-delay error is then obtained with the aid of ordinary lead networks ahead of the analog-to-digital converter, or by the more sophisticated method of predictive extrapolation, either in the digital computer or with extrapolating digital-to-analog converters [1], [5], [7]. The processing delay Δt is essentially reduced to the time taken for the digital computations on each analog sample if the

digital computer is equipped with a direct memory-access channel. With this option, which should be included in every digital computer used for hybrid computation, it will usually be found that the skewing errors are small compared to the time-delay errors.

ANALOG-SYSTEM COMPONENTS

We turn next to the design of the analog-computer system. ± 100-V analog computing elements may be preferred for optimal accuracy in "slow" simulation, because static accuracy in nonlinear computing elements is less expensive to obtain and maintain in a ± 100-V system. Note, however, that static errors within 0.02 percent of half-scale *can* be maintained with ± 10-V summers and multipliers as well as with ± 100-V components [7]. ± 10-V amplifiers can be faster and less costly than ± 100-V amplifiers; but the main advantage of ± 10-V systems is the possibility of utilizing lower computing impedances to improve bandwidth, integrator drift, and computer noise. Miniaturization, which brings further bandwidth advantages, is also managed more easily at ± 10-V because of the reduced heat-dissipation requirements.

The most accurate "slow" analog computers still employ wirewound resistors, but these are giving way to new precision metal-film resistors, at least wherever bandwidth is a consideration. Carbon-film resistors may be regarded as obsolete for analog-computer applications.

In the most accurate analog computers, capacitor dielectric absorption, calibration, electroelastic, thermal, and hysteresis effects are critical accuracy-limiting factors; they cause component errors of the order of 0.02 percent [1]. Fine-grain polystyrene film is still the preferred dielectric. Compression trimming of plastic-film capacitors, which implies the possibility of mechanical deformation of foil and dielectric, is apt to cause more trouble than it cures. A significant step forward was the 1965 development of a simple equalizer circuit [8] capable of reducing dielectric-absorption effects to below 0.005 percent. An important consequence is the possibility of a more favorable compromise between capacitor dielectric absorption and the temperature coefficient of capacitance.

The hybrid-computer era has seen relatively little improvement in the design of analog multipliers and function generators, although it is now possible to compensate both special-purpose and general-purpose diode function generators accurately for diode-breakpoint drift with temperature by including matched diodes in the bias supplies [7]. Diode quarter-square multipliers have almost entirely displaced servos and time-division multipliers and have attained component accuracies of better than 0.05 percent of half-scale; special wide-band multipliers have errors as low as 0.5 percent at 10 to 50 kHz [2]. Accurate diode quarter-square multipliers are expensive and have obvious disadvantages due to their inherent piecewise-linear approximation of the product. Some renewed investigation of time-division multipliers employing new types of solid-state switches may well be in order.

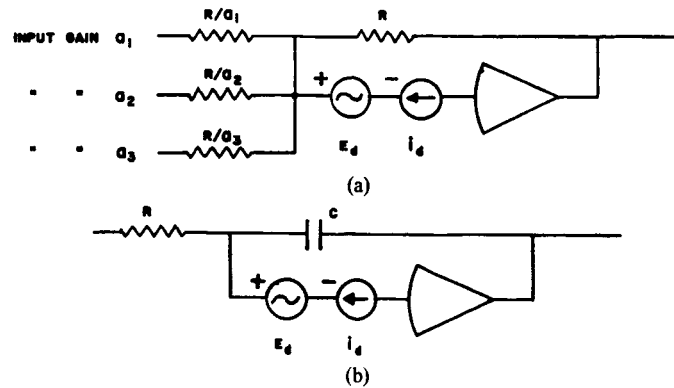Where bandwidth considerations permit, nonlinear op-

Fig. 6. (a) Summing amplifier. (b) Electronic integrator.
Both show offset voltage and current sources.

TABLE I

Typical Drift Errors in General-Purpose Analog Computers (1966)

(*Maintained at* 25 ± 5 *deg* C, *with regulated power supplies* (2))

	Integrated Circuit ± 10 V, no chopper $R = 10$ K	± 10 V FET input, no chopper $R = 10$ K to 100 K	High-quality chopper-stabilized amplifier (FET chopper)	
			± 10 V $R = 10$ K to 100 K	± 100 V $R = 100$ K to 1 M
E_d	$\pm 30 \,\mu$V	$\pm 60 \,\mu$V	$\pm 10 \,\mu$V	$\pm 40 \,\mu$V
i_d	± 30 nA	negligible	negligible	negligible
e_{sum} $(a_1 + a_2 + a_3 = 1)$	$\pm 360 \,\mu$V	$\pm 120 \,\mu$V	$\pm 20 \,\mu$V	$\pm 80 \,\mu$V
e_{sum} $(a_1 + a_2 + a_3 = 21)$	$\pm 960 \,\mu$V	± 1.3 mV	$\pm 220 \,\mu$V	$\pm 880 \,\mu$V
e_{int} (typical computer run, $t/RC = 50$)	± 16.7 mV	± 3 mV	$\pm 500 \,\mu$V	± 2 mV
e_{int} (long integration, $t/RC = 1000$)	± 0.33 V	± 60 mV	± 10 mV	± 40 mV

erations are performed digitally, or by multiplying D/A converters. Even after 20 years of development, setup procedures for diode function generators are still cumbersome; a number of manufacturers have developed card- and patchboard-programmed diode function generators to simplify function storage. The problem of generating functions of two or more variables is as bad as ever [1]. With small general-purpose digital computers available at prices as low as $10 000, the best solution for function generation, at least in "slow" analog simulation, appears to be to utilize such a small digital computer for the generation of between five and ten analog functions: at a price of $10 000 for the digital computer, plus $5000 for a simple linkage, the cost per function is still comparable to that of commercially available card-programmed function generators [9].

Low-Drift dc Amplifers

Solid-state dc amplifiers can have substantially lower drift than vacuum-tube amplifiers. Dc amplifier drift is specified in terms of the equivalent voltage and current offset after a given time interval, power-supply-voltage change, or temperature change. Referring to Fig. 6, the equivalent-input-offset voltage E_d and current i_d are, respectively, defined as minus the input voltage and current needed to return the open-loop output to zero (balance condition) with a grounded voltage source and a current source at the summing point. In the summer and integrator circuits of Fig. 6,

the output-voltage errors due to drift, assuming high amplifier gain, are [1]

$$e_{sum} = -(a_1 + a_2 + a_3 + 1)E_d - Ri_d \quad \text{(summer)}$$

$$e_{int} = -\frac{t}{RC}(E_d + Ri_d) \qquad \text{(integrator)} \quad (2)$$

where t is the computing time. Table I illustrates drift-error effects for typical computer dc amplifiers operated over normal room-temperature ranges.

The design of high-quality solid-state dc amplifiers, both discrete-component and integrated-circuit types, has reached a high state of development. Figure 7(b) illustrates an amplifier circuit typical of recent integrated-circuit design. It is worthwhile to list the circuit features used to obtain low drift in such amplifiers.

A *differential input stage* with a current-source transistor in the emitter return ties the temperature dependence of the first-stage output to the *difference* of two transistor V_{BE}'s; transistor matching for V_{BE} and β can reduce voltage drift from 2.5 mV/deg C to as little as 5 μV/deg C [1]. Matching is especially favorable in monolithic integrated circuits; differential heating of the two input transistors must be minimized by low values of the collector current ($< 200 \,\mu$A). The first-stage gain should be sufficiently high (at least 10) to make drift effects in later stages negligible. A roughly matched second differential-amplifier stage may

β, V_{BE}, OR I_{CBO}
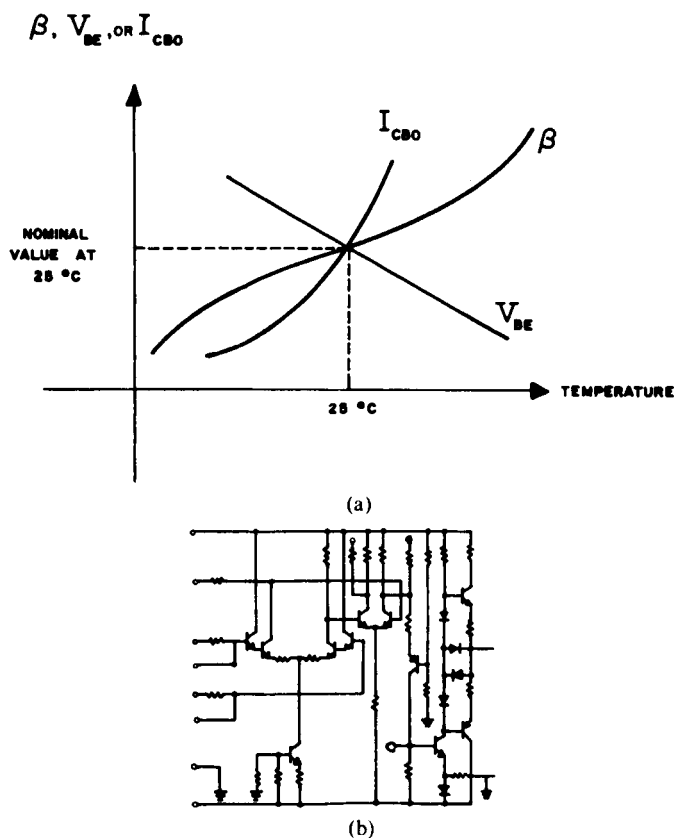

(a)

(b)

Fig. 7.   (a) Temperature dependence of transistor parameters. (b) Typical integrated-circuit dc amplifier (similar to Westinghouse Type 174Q).

follow the first for more perfect common-mode rejection, or common-mode feedback from a succeeding stage may be used [1].

Differential-amplifier voltage drift can, we note in passing, be reduced below 0.5 μV/deg C if the V_{BE} temperature coefficients are matched through individual collector-current adjustments [1], but this has not been applied in production.

Current drift, which may be thought of as a change in input-transistor bias-current requirements, is due to changes in i_c/β and in the leakage current I_{CBO}. The latter is negligible for silicon transistors at room temperatures; at higher temperatures, the I_{CBO} change will, in fact, *counteract* the i_c/β change with temperature. It is possible to compensate for the current drift with the aid of temperature-dependent current sources comprising thermistors, diodes, or transistors, or to counteract changes in β by corresponding temperature-dependent changes in i_c [1]; but such compensation schemes work best over limited temperature ranges (perhaps 10 to 20°C) because of the nonlinear temperature dependence of β [Fig. 7(a)]. In general, *the best method for reducing current drift is to use high-β transistors and to keep the first-stage collector currents below 100 to 200 μA*. This reduces i_c/β in toto and, hence, also its changes with temperature. In Fig. 7(b), the input current has been further reduced through the inclusion of common-collector input transistors, which roughly divide the bias-current re-

quirement by another factor of β. It is thus possible to reduce the *total* input current well below 100 nA between -50 deg C and 180 deg C. The differential-amplifier connection as such can reduce current drift only if both input transistors are connected to similar source impedances; this is rarely the case in general-purpose computers.

In dc amplifiers requiring very low input currents and high input impedances, *junction FET's may replace the input transistors* to reduce input currents to the order of picoampères; MOSFET's are not recommended for dc-amplifier input stages, since they may tend to change their properties with time. The voltage drift of FET dc-amplifier stages can be reduced through selection of an optimal operating point [12] and by matching of differential-amplifier stages. The best FET amplifiers now have about the same voltage drift as high-quality bipolar transistor stages (3 to 10 μV/deg C; also see Table I).

IMPROVED ELECTRONIC CHOPPERS

In accurate general-purpose analog computers, the possibility of open-loop integration of offset voltages and currents still makes dc amplifier chopper stabilization [1] mandatory. MOSFET choppers operated at frequencies of 100 to 1000 Hz have long-term input voltage offsets below 20 μV and negligible current offset over a wide range of temperatures. Sinusoidal drive to the input chopper reduces switching spikes. FET choppers permit low-voltage drive at higher frequencies than neon-bulb/photoresistor choppers and are also more reliable.

INTEGRATED-CIRCUIT CONSIDERATIONS

Special integrated-circuit considerations in amplifier design require that voltage levels depend on resistance *ratios* rather than resistance *values*. Large resistance values should be avoided whenever possible, extra transistors can be added just about as easily as resistors, and transistor matching and thermal coupling must be generally excellent. While earlier types of monolithic integrated circuits had only NPN transistors, complementary transistors are readily included in more recent designs, a very important advantage. Breakdown voltages have been greatly improved recently, so that ± 10-V operation is possible without much adverse effect on bandwidth. New ceramic insulation [23] can be expected to improve monolithic dc amplifiers still further.

THE STRIKING ADVANTAGES OF FEEDFORWARD AMPLIFIERS

Feedforward amplifiers (Fig. 8) pass high-frequency-signal components through a high-pass filter directly to a wideband output stage, while cascading a high-gain dc amplifier with the output stage for lower frequencies; the well-known chopper stabilizers for dc amplifiers are special cases of feedforward circuits [1].

The now widely used ± 10-V *high-gain class AB complementary output stage* of Fig. 8 can have a gain as high as 2000 at 20 kc with unity gain at 25 to 100 MHz, depending on transistors and load. It has relatively low quiescent current and is especially well suited to feedforward operation,
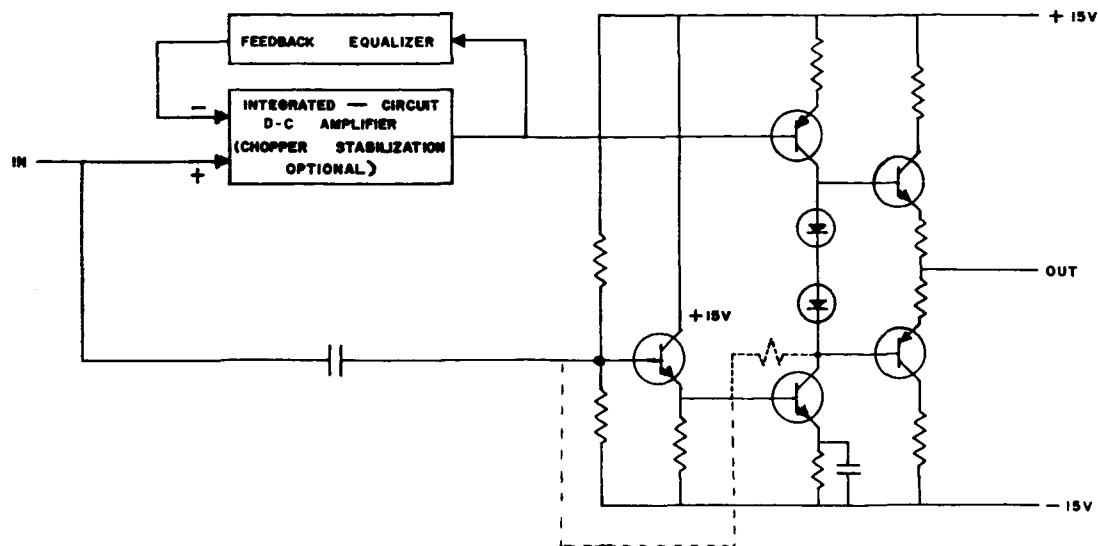
Fig. 8.   Feedforward dc amplifier employing a fast class AB output stage and an integrated-circuit dc amplifier
(LOCUST computer, University of Arizona).

since high- and low-frequency signals are added without the usual mixing loss [1]. The combination of such a feedforward output stage with a modern low-cost integrated-circuit input amplifier yields a number of remarkable advantages:

1) *The amplifier high-frequency response is essentially that of a single wide-band stage followed by an emitter follower*, and no shunt rolloff networks are required for stabilization. As a result, such amplifiers can produce full ± 10 V, 30 mA output all the way up to 2 MHz.

2) *The feedforward combination completely avoids the usual compromise between high-frequency response and low-drift/low-noise design of the dc-amplifier input stage.* If this fact were more generally realized, the current drift and low-frequency-flicker noise of many general-purpose integrated-circuit dc amplifiers could be improved by at least a factor of three by the simple expedient of reducing their input-stage collector current; the feedforward connection would take care of the frequency response.

The circuit of Fig. 8, intended for the University of Arizona's LOCUST iterative differential analyzer, combines a commercially available, low-cost, low-drift integrated-circuit amplifier with the discrete-circuit output stage to produce a high-quality, wide-band amplifier at very low cost. A chopper-stabilizer stage can be added to permit more accurate low-frequency integration.

A very similar circuit also permits the design of inexpensive ± 100-V, 25-mA dc amplifiers: one merely replaces the class-AB complementary output stage shown in Fig. 8 by an NPN/NPN 100-V totem-pole output stage having a gain of approximately 12.

IMPROVEMENTS IN ELECTRONIC MODE-CONTROL SWITCHING

The computing errors caused by electronic mode-control switching for integrators and track-hold circuits are dis-

cussed in detail in [1], [7], [16], and [17]. In addition to the high switching speeds already discussed, high-quality electronic switches must have high back resistance, low forward resistance, minimum voltage and current offset, and low spike-causing capacitance between logic (control) and signal circuits. Figures 9 and 10 illustrate two modern electronic switching schemes for integrator control; others are described in [1]. In particular, the circuit of Fig. 9 employs 200 to 800 ns FET switches inside feedback loops to avoid FET forward-resistance effects; it practically duplicates the static accuracy of relay switching. The circuit also permits fast integrator resetting: in RESET, the integrating capacitor charges through a low-impedance path. On the other hand, the requirement for a separate holding capacitor (in reality, one capacitor for each time scale provided), as well as multiple switching, makes this circuit quite cumbersome. Thr finite forward resistance r_F of the bipolar transistor grounding the holding capacitor causes the stored value to be in error by 200 $\pi f r_F C$ percent for sinusoids of frequency f [1]. r_F, which is of the order of 10 ohms, will also cause some feedthrough from the integrator inputs in RESET. This mode-control circuit is, therefore, best suited for relatively slow, high-impedance computers.

The switched-amplifier method [Fig. 10(a), (b)] used in the University of Arizona's ASTRAC II, APE II, and LOCUST computers employs a switched feedback amplifier whose very low output impedance (less than 0.2 ohms in ASTRAC II) practically shortcircuits the effect of the integrator inputs so that they need not be opened by a separate switch in RESET. No HOLD mode is provided; all holding is done with separate track-hold circuits, which greatly simplifies digital control and improves frequency response, but may require additional equipment. Since the high-current switch is never saturated, switches of this type are the fastest integrator mode-control switches available (total switching time 40 to 80 ns, differential switching time less than 20 ns). The ASTRAC-II circuit is described in
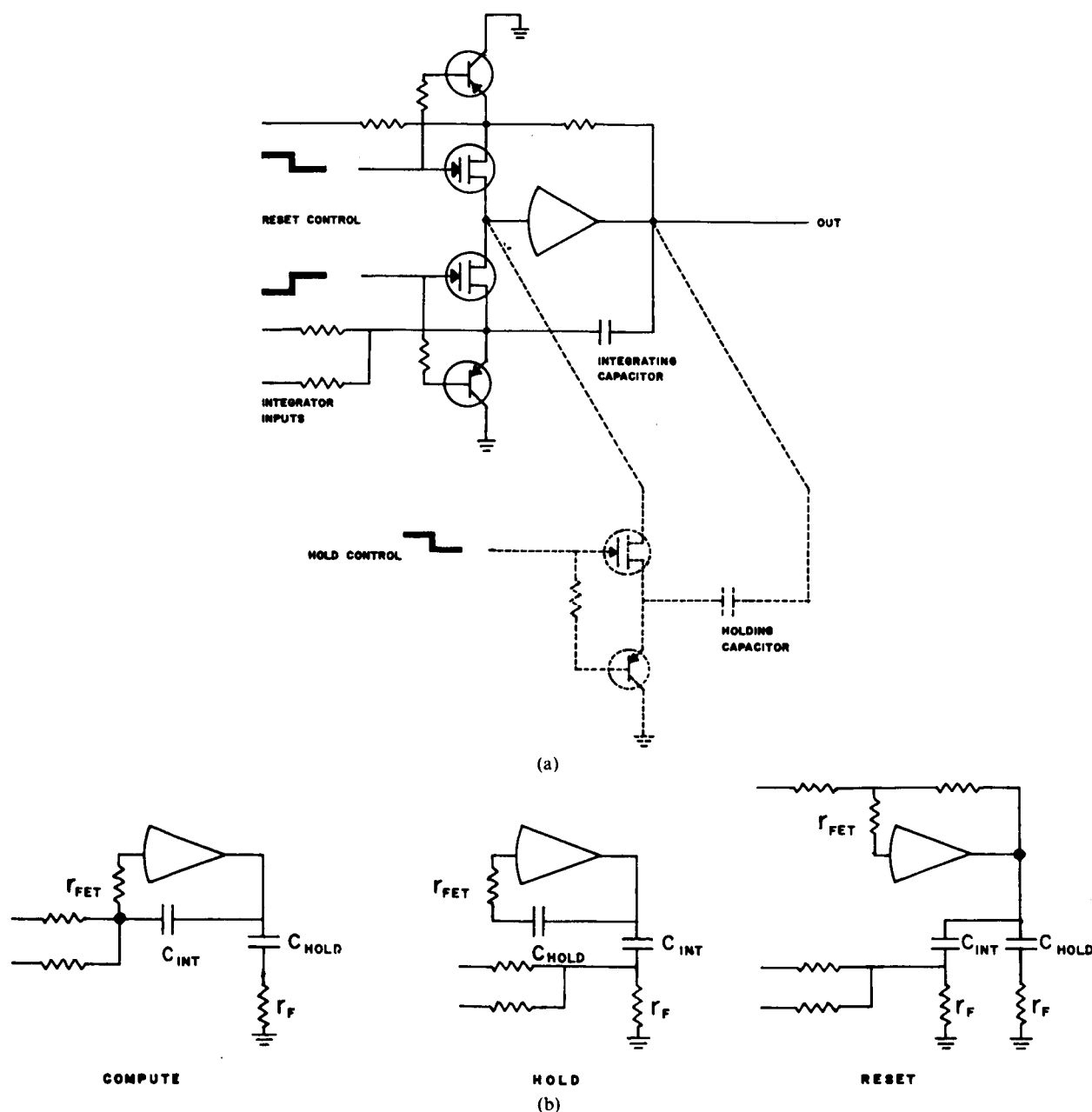
Fig. 9.   FET mode-control circuit with optional HOLD circuit indicated (a) in dash lines and (b) equivalent circuits.

detail in [17]; Fig. 10(c) shows a developmental switch with a low-drift FET input stage.

DIGITAL ATTENUATORS FOR COEFFICIENT SETTING

Digital attenuators for coefficient setting are simplified D/A multipliers designed to switch only in the computer RESET (or POTSET) and HOLD modes. Digital attenuators are set up either by keyboard or, more frequently, by digital computer commands. They offer higher setting speeds and, hopefully, greater reliability than servo-set coefficient-potentiometer systems.

The earliest commercially available system set the last few bits of its digital attenuators through a feedback loop similar to a servo-set-potentiometer system. This permitted the use of low-offset FET switches without special compensation for the FET resistances. It seems simpler, however, to employ open-loop setting of bistable reed relays, which give each attenuator a nondestructive memory and use no control current during computation. After each attenuator is set, its output for a constant reference input is readily read out into a digital voltmeter or the digital computer, so that visual and/or automatic checking is still achieved. With the computer reference voltage turned off while attenuators are being set, relay contacts never interrupt current; they are operated well below their ratings and should last a long time. For any reasonable number of attenuators, relay switching is *not* more time consuming than electronic switching: in the University of Arizona's
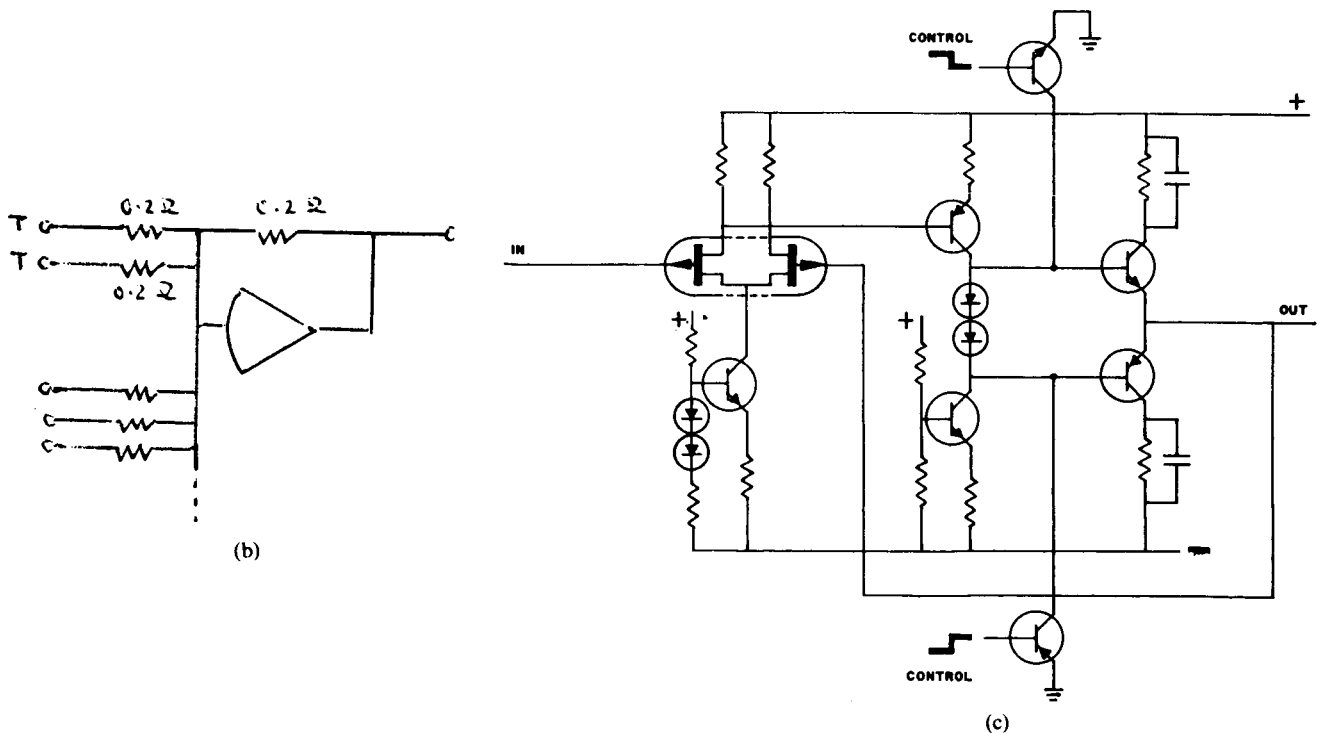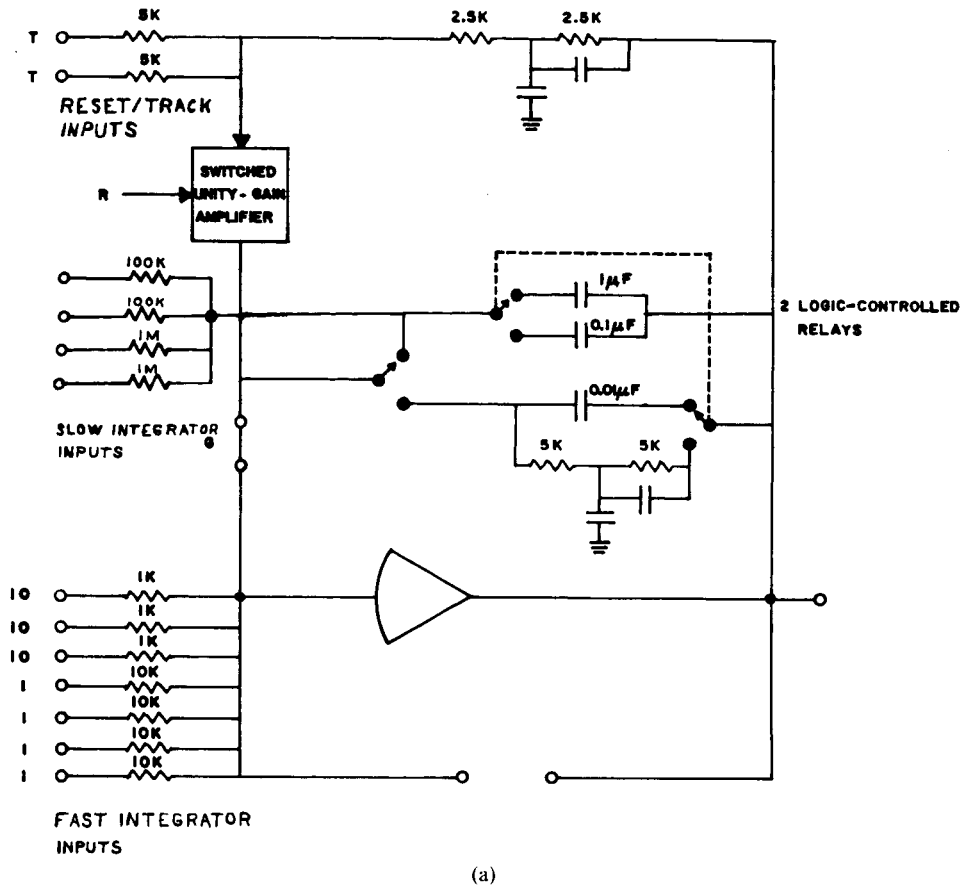
(a)

(b)

(c)

Fig. 10. (a) LOCUST summer/integrator employing switched-amplifier mode control to obtain differential switching times below 20 ns (see also [17]). Separate low- and high-impedance summing networks are used for fast and slow computation, so that three logic-switched capacitors yield five different time-scale ranges with integrator input gains all the way from 10^5 s^{-1} to 1 s^{-1}. The fourth state of the two-relay combina-tion produces a low-impedance summer without any need for bottle plugs. The circuit employs a 30 MHz, ± 10-V amplifier similar to that of Fig. 8. (b) Simplified equivalent circuit of a switched-amplifier mode-control system in RESET or TRACK. (c) Switched FET-input ampli-fier proposed for the LOCUST system.

new digital-attenuator system, C. Pracht introduced the notion of making all relay driver bistable devices (transistor flip-flops) [18], so that the digital control system can go on to set the entire attenuator system while the first few relays are still settling. All attenuators are thus set in little more than the time required to set a single relay. The system is, in fact, so fast that it becomes possible to combine the relay-driving flip-flops into shift registers with *serial* input: this greatly simplifies the digital addressing of individual attenuators (one control line per attenuator, instead of 14 to 16).

Another question to be resolved is whether digital attenuators should employ binary or BCD weighting. The latter favors the use of manual keyboards, while binary weighting offers equipment savings and is more natural for most digital computers. The University of Arizona's system employs binary weighting and has replaced the coefficient-setting keyboard by a digit switch presetting a BCD counter; the latter counts the preset number of clock pulses into a binary counter/shift register, which can also accept parallel input from the digital-computer output bus and controls attenuator settings [18]. All manual settings are read back at once on the computer digital voltmeter.

The reader may have begun to note that the design of a practical digital-attenuator system is a rather sophisticated problem. Figure 11 illustrates some attenuator-network designs; many combinations are possible. Figure 11(a) shows a simple binary-weighted summing network employing SPDT (form C) relays to prevent capacitive feedthrough. Since the required 2^{14} to 1 resistance range is too large, Fig. 11(b) implements at least the low-current branches with network transfer impedances; this circuit also permits the use of the somewhat cheaper SPST (form A or B) relays. Both Fig. 11(a) and 11(b) require summing-junction patching. The circuit of Fig. 11(c) permits patching to summing amplifier or integrator gain-of-10 or gain-of-1 inputs if the latter are loaded as shown. The simple ladder network of Fig. 11(d) can be similarly patched and permits especially simple construction since an entire set of similar metal-film resistors can be deposited in a single operation.

The choice of resistance values in Fig. 11(a) to (d) must satisfy the following requirements.

1) The load on the driving amplifier must not be larger than that of a comparable coefficient potentiometer.

2) The summing-junction-to-ground impedance of the attenuator has a lower limit determined by bandwidth specifications (i.e., the attenuator must not kill the operational amplifier feedback).

3) The attenuator must not attenuate high frequencies unduly; with suitable lead capacitors, digital attenuators, which can be located directly behind the computer patchbay, should create less phase shift than coefficient potentiometers.

Note that even with the best possible resistance values, the attenuator circuits of Fig. 11(c) and (d) cannot produce a
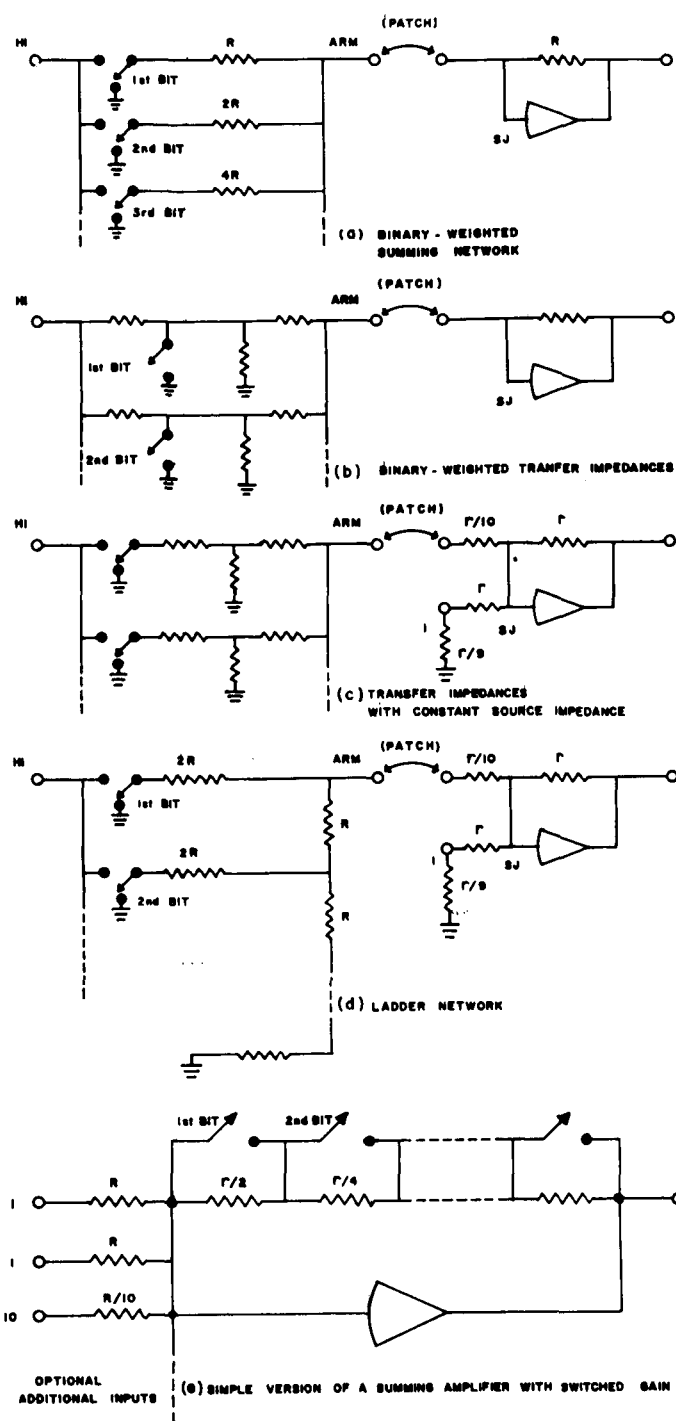

Fig. 11.   Digital-attenuator design.

full gain of 1 or 10 when patched into the corresponding amplifier inputs. Maximum coefficients of the order of 0.5 and 5, respectively, seem practical.

To improve the network compromises, some digital-attenuator systems associate an amplifier with each digital attenuator. If the added cost of such an amplifier is accepted, it would seem desirable to provide it with summing inputs and to switch the feedback network rather than the input network, so that amplifier offset and noise are not amplified together with the signal [Fig. 11(e)].
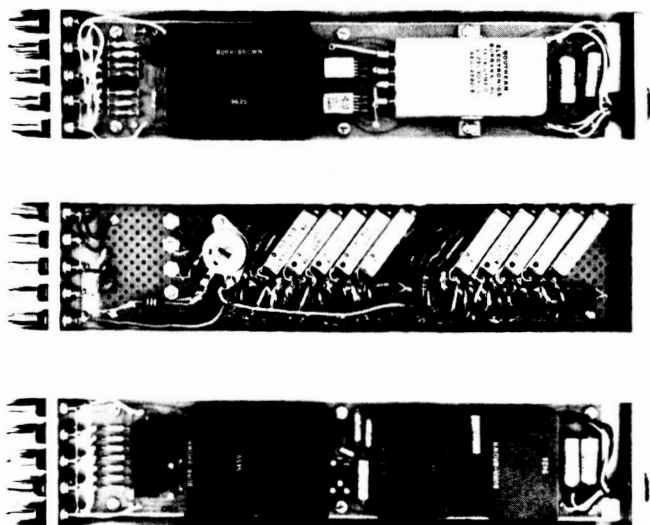
Fig. 12. These ASTRAC II ± 10-V computing elements plug directly into the rear of the shielded analog-computer patchbay (University of Arizona).

New Packaging Techniques

To improve maintainability and convenient system expansion as well as computing bandwidth, analog computing elements ought to be plugged directly into the rear of a shielded analog-computer patchbay whenever it is at all possible. With reasonably designed transistor and integrated-circuit packages, such patchbay mounting of all analog computing elements is possible for ± 100-V as well as for ± 10-V computers, although the ± 100-V machine will require careful management of forced-air cooling. In the ± 10-V ASTRAC II computer shown in Fig. 4, the *only* analog computing elements not located directly in the patchbay are the coefficient-setting potentiometers, which must necessarily be situated on the front panel. Digital attenuators, however, can be plugged into the patchbay, so that the last remaining analog signal wiring is eliminated. If the top and/or bottom rows of patchbay terminations are used for attenuator connections, the attenuator assemblies can be located above or below the analog patchbay, so that they do not take up much of the valuable space behind the analog patchbay.

In the ASTRAC II and LOCUST computers, amplifiers, electronic switches, multipliers, etc., are mounted in individual shielded boxes plugged into the rear of the patchbay (Fig. 12). Figure 13 illustrates the construction of a small educational iterative differential analyzer at much lower cost. Here, simple printed-circuit cards having a shielded ground plane on one side replace the shielded amplifier boxes. Sets of such cards bearing integrated-circuit analog and digital computing elements are plugged directly into the rear of the analog and digital patchbays. Note that the amplifier overload lights and balance controls are mounted on the front of the amplifier cards, so as to be accessible from the front panel. There is no wiring to the cards at all; even power supply wiring is supplied by spare patchbay terminations.
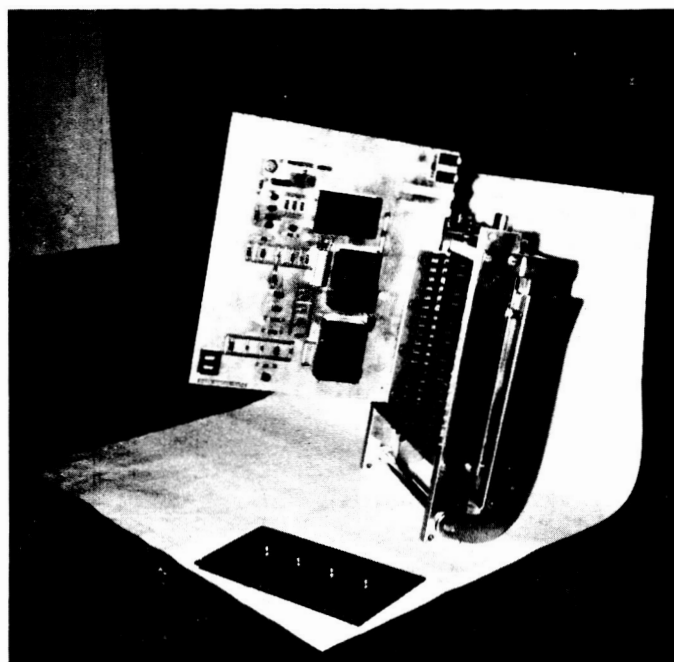

Fig. 13. APE II amplifier card plugged into the patchbay (here removed from the computer), and an integrated-circuit logic card.

In any fast hybrid-computer system, grounding and shielding systems designed to minimize crosstalk and digital-noise effects become a first-class design problem, which must be considered from the start. Reference [22] summarizes the principal design considerations.

Digital Control and Statistical Measurements

Integrated-circuit logic permits accurate timing and flexible control of iterative-analog-computer subroutines at very low cost. The ASTRAC II/LOCUST digital control system illustrated in Fig. 14 counts digital-clock pulses to produce not only analog-computer reset pulses but, also, precisely timed sampling pulse trains used to read out solution values and to compile random-process statistics. A crystal-controlled clock provides pulses at switch-selected rates of 1000, 500, 250, 100, 50, 25, and 10 kHz. A three-decade decimal counter (C_1) then produces a computer-reset pulse (R pulse) after 1000 clock pulses, so that computer runs or data samples can be generated at rates of 1000, 500, 250, 100, 50, 25, and 10 Hz; an external clock can also be used. Counters C_1 and C_2 are preset to produce readout sampling pulses S_1, S_{1D}, S_2, S_{2D} at digit-switch-selected multiples of 1/1000 computer run to permit flexible control of integrators, track-hold circuits, and track-hold pairs [1], [2], [19]. The logic-switched integrator capacitors are normally slaved to the repetition-rate switch, so that time-scale changes are automatic. C_3 is a subroutine counter counting computer runs, comparator responses, or other events.

Additional simple logic circuits produce automatic progressive delay of the sampling pulses S_1, S_{1D} after a preset number of n computer runs. This "scan mode" permits sampling readout of repetitive computer solutions into slow recorders and digital computers and is also used for
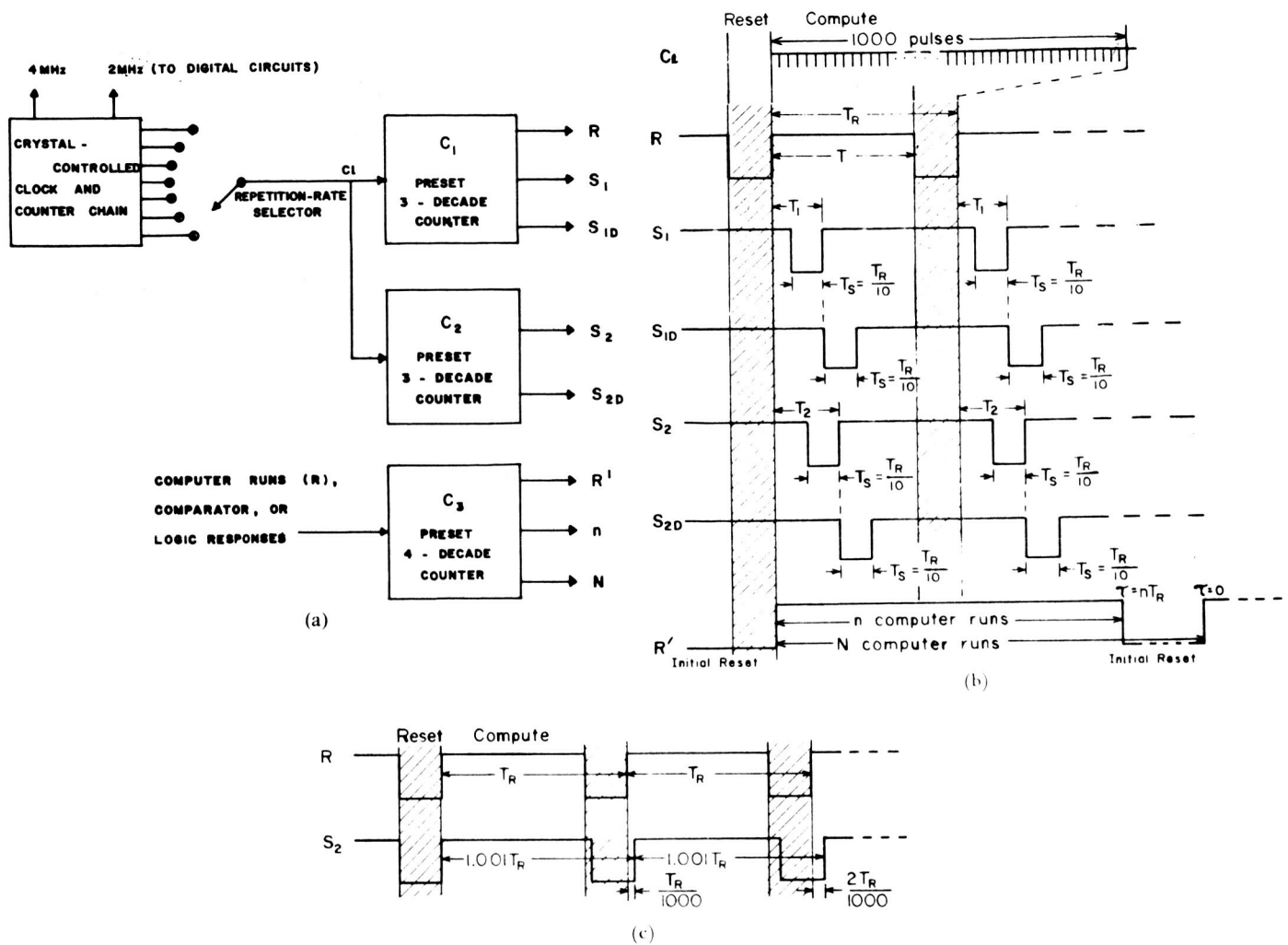
(a)

(b)

(c)

Fig. 14.   (a) ASTRAC II/LOCUST digital-control system. (b), (c) Timing scheme. The reset period ($R = 1$) lasts for 100 clock pulses once every 1000 clock pulses for repetitive operation; manually controlled real-time operation is also possible. Sampling pulses S_1, S_2, S_{1D}, S_{2D} occur at digit-switch-selected times after the start of each computer run. Sampling pulses can also be made to advance after a programmed number of $n = 1$ to 100 000 computer runs (shown in Fig. 14(c) for $n = 1$). This permits automatic plotting of repetitive analog-computer solutions and correlation functions (see also [1], [2], and [19]). All these clock functions can be implemented with only two integrated-circuit cards bearing multiple preset counters.
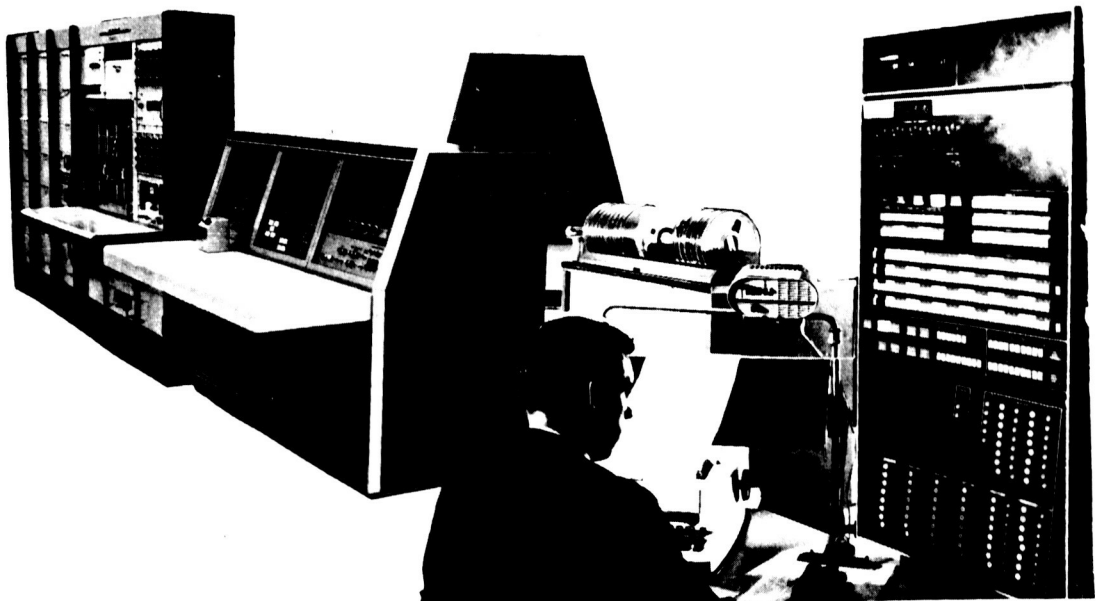

Fig. 15.   Analog computer, space-vehicle mockup, and digital computer are the essential components of a complex launch-system simulation. this composite photograph shows only a small portion of the computing equipment committed to a large simulation (Boeing Huntsville Simulation Center).

automatic plotting of correlation functions. Such statistical measurements, an especially fruitful application of hybrid computers, are discussed in detail in [2] and [21].

REFERENCES

[1] G. A. Korn and T. M. Korn, *Electronic Analog and Hybrid Computers.* New York: McGraw-Hill, 1964.

[2] G. A. Korn, *Random-process Simulation and Measurements.* New York: McGraw-Hill, 1966.

[3] J. McLeod, *Simulation: Dynamic Modelling of Things and Ideas.* New York: McGraw-Hill, to be published, 1967.

[4] W. J. Karplus, "Error analysis of hybrid-computer systems," *Simulation,* vol. 6, pp. 120–136, February 1966.

[5] E. G. Gilbert, "Dynamic-error analysis of digital and combined analog-digital computer systems," *Simulation,* vol. 6, pp. 241–257, April 1966.

[6] J. Vidal and W. J. Karplus, "Characterization and compensation of quantization errors in hybrid-computer systems," *1965 IEEE Internat'l Conv. Rec.,* pt. 3, pp. 236–241.

[7] W. Giloi, A. Kley, and G. Meyer-Brötz, "Arbeiten über Analog-Rechentechnik, *Telefunken Ztg.,* vol. 39, no. 1, 1966.

[8] C. H. Single, *U. S. Patent Application,* 1965.

[9] D. M. Tutelman, "A hybrid-computer technique for nonlinear-function generation," *Simulation,* vol. 6, pp. 308–322, May 1966.

[10] G. Meyer-Brötz and A. Kley, "The design of d-c amplifiers with high common-mode rejection," *Electron. Engrg.,* February 1966.

[11] ——, Zur Verbesserung der Bandbreite gegengekoppelter Verstärker durch Vorwartskoppelung, *Arch. Elek. Übertragung.,* vol. 19, p. 546, 1965.

[12] J. R. Miller et al., *Communications Handbook,* pt. 1, Texas Instruments, Inc., Dallas, Tex., 1965.

[13] T. B. Bergersen, "Field effect transistors in chopper and analog switching circuits," Motorola Semiconductor Products, Inc., Phoenix, Ariz., Application Note 220, 1965.

[14] R. J. Widlar, "A monolithic operational amplifier," Fairchild Semiconductor, Mountain View, Calif., Appl. Bull. 105/2, 1965.

[15] S. T. Robertson and J. J. Wisseman, "High-performance integrated operational amplifiers," Motorola Semiconductor Products, Inc., Phoenix, Ariz., Application Note 204, 1965.

[16] G. A. Korn, "Performance of operational amplifiers with electronic mode switching," *IEEE Trans. on Electronic Computers,* vol. EC-12, pp. 310–312, June 1963.

[17] H. R. Eckes, "A fast mode-control switch for iterative differential analyzers," *IEEE Trans. on Electronic Computers,* vol. EC-14, pp. 946–950, December 1965.

[18] C. Pracht, M.S. thesis. Dept. of Elec. Engrg., University of Arizona, 1966, in preparation.

[19] H. R. Eckes, and G. A. Korn, "Digital program control for iterative differential analyzers," *Simulation,* vol. 2, pp. 33–41, February 1964; see also [1].

[20] G. A. Korn, "Hybrid-computer techniques for measuring statistics from quantized data," *Simulation,* vol. 4, pp. 229–239, April 1965; see also [2].

[21] ——, "Hybrid-computer Monte-Carlo techniques," *Simulation,* vol. 5, pp. 234–245, October 1965; see also [2].

[22] ——, "Reduction of digital noise in hybrid analog-digital computers," *Simulation,* vol. 6, pp. 151–152, March 1966.

[23] M. Leeds, "Better isolation boosts IC performance," *Electronic Design,* vol. 14, pp. 17–19, September 1966.

Measurement of Phase Shift

In analog computer circuitry, the need for accurate phase measurements often arises. Inverting amplifiers contribute significant phase shifts at frequencies well below the 3-db frequency. These phase shifts cause large errors in a problem solution. Conventional phase measuring equipment does not operate at the higher frequencies where high-speed computers now operate. A simple procedure has been developed which utilizes a dual beam oscilloscope, or any indicating device with a differential input, to measure the phase shift between two sinusoids.

Consider the difference of two sinusoids whose phase shift is to be determined.

$$E(t) = a \sin \omega t - b \sin (\omega t + \theta). \quad (1)$$

By simple trigonometric manipulation, $E(t)$ may be expressed as follows:

$$E(t) = [a^2 + b^2 - 2ab \cos\theta]^{1/2}[\sin (\omega t + \alpha)] \quad (2)$$

where

$$\alpha = \tan^{-1} \frac{b \sin \theta}{a - b \cos \theta}.$$

Assume that the amplitude of the second sinusoid is adjusted until the amplitude

of the difference signal is a minimum. This condition occurs when

$$b = a \cos \theta, \quad (3)$$

as simple differentiation of the amplitude of $E(t)$ will show. Under this minimum condition the difference signal is given by

$$E(t) = a \sin \theta \sin (\omega t + \alpha). \quad (4)$$

To determine the unknown phase difference θ between a $\sin \omega t$ and $b \sin (\omega t + \theta)$, (e.g., amplifier input and output!), *apply these voltages to the differential input of a Tektronix or similar oscilloscope, and vary the second input gain so as to minimize the difference signal seen on the screen. The ratio of the difference amplitude to a equals sin θ, the sine of the desired phase angle.*

It is still necessary to determine the quadrant in which θ is to be found. Unless this is known *a priori,* it becomes necessary to either view the waveforms on a dual-trace scope or examine the zero crossings in a logical fashion to determine the qudarant. In addition, the sensitivity of the null is quite poor for angles close to multiples of 90°. For such angles, it would be necessary to compare the signal whose phase angle is to be determined with cos ωt, rather than sin ωt.

HOWARD HANDLER
Dept. of Elec. Engrg.
University of Arizona
Tuscon, Ariz.

Acoustical simulation of a randomly fading communication channel

by A. M. MANDERS, *University of Florida*
and J. V. WAIT, *University of Arizona*

JOHN V. WAIT received his BSEE from the University of Iowa in 1955, MSEE from the University of New Mexico in 1959, and PhD from the University of Arizona in 1963. He has worked in the general field of computing and signal processing systems for a number of years, and has published papers on hybrid-code differential analyzers, binary correlators, and hybrid computer circuit design. More recently he has held teaching positions at the University of California, Santa Barbara, and the University of Florida GENESYS facility at Cape Kennedy. In September, 1966 he returned to the high, dry lands of the Southwest, where he will again be associated with Granino Korn's Analog/Hybrid Computer Laboratory at the University of Arizona.

ARNFINN MOE MANDERS was born in Oslo, Norway in 1932. He came to the United States in 1955 after completing his studies at Schous Institute of Technology in Oslo. In 1957 he received the BS and MS degrees in Electrical Engineering from Michigan Technological University and in 1963 the PhD in Electrical Engineering from the Polytechnic Institute of Brooklyn.

Dr. Manders has been employed by Western Electric, IBM Research, and ITT. He has served as instructor at PBI and Assistant Professor at MIT. Since 1964 he has been employed by the University of Florida, GENESYS, where he is an Associate Professor of Electrical Engineering. He is currently conducting research in coding and design of signals for fading channels.

SUMMARY

Experimental studies of fading RF communication channels are usually very difficult to perform under actual operating conditions. This paper describes a fading channel simulator which is easily constructed and flexible in operation. Using this simulator, modulation and coding techniques suitable for use in communication and ranging equipment can be tested and optimized for operation during periods of severe fading. The simulator uses ultrasonic acoustical waves transmitted through water to achieve favorable changes in the space and time scales; typical fading rates are from 2 to 30 fades per second.

The paper also describes simple hybrid statistical measurement equipment used in studying the probability distributions and correlation functions of the received signals.

INTRODUCTION

Although today we normally think of *simulation* of systems primarily by means of analog, digital, or hybrid computing setups, we should not forget that some important phenomena are not easily modeled by a finite number of ordinary differential-difference equations which can then be "solved" by conventional computer methods.

An important case where a different approach is required is the simulation of a (randomly) fading radio-communication channel. Here we have a system with randomly varying, space- and time-dependent distributed parameters. Simulation of such channels is important to the study of many communication and ranging systems.

Although the establishment of a real-world channel using a conventional radio transmitter/receiver link is not too difficult, this approach has several obvious disadvantages:

1. Useful studies of many long-distance propagation phenomena require great distances between transmitter and receiver, with the attendant difficulties of logistics, off-line communication, etc.
2. Even if the physical distance were not enough of a problem, one has the additional problem of accurately establishing precise, coherent frequency and time references at both ends of a long-distance channel.
3. Most real-world propagation paths do not possess a sufficient degree of stationarity in their statistical properties to permit precise, reproducible studies.
4. One cannot easily change the location of the transmitting and receiving stations, antennas, etc., and then return to some previous configuration for comparative purposes.
5. The rates-of-change of real-world fading phenomena are relatively slow, thus making statistical measurements tedious and time-consuming.

This paper describes a relatively inexpensive channel simulator that operates with much more convenient distance and time scales. It is useful both as an instructional device and as a reliable research tool. A simple statistical analyzer for making amplitude distribution and correlation measurements is also described.

The technique involves the use of ultrasonic (200 kHz) acoustic waves, in our case transmitted through a water tank. This method, although not new, is perhaps not as well known as it should be (see, e.g., reference 1). We have found the simulator to be a surprisingly stable, reliable mechanism for studying fading-channel phenomena. For example, the envelope and phase statistics of transmitted signals may be studied using a variety of propagation modes, including pure random (incoherent) scattering, pure specular, and mixtures of random and specular. We have used the simulator in the testing of new designs for phase-lock loops and other received-signal processing systems.

GENERAL DESCRIPTION

Figure 1 shows a sketch of a typical tank setup, including a block diagram of the associated electronics. The simulator consists of a 5 × 3 × 2-foot plywood tank partially filled with tap water (figure 2). Two narrow-beam ultrasonic transducers (figure 3) are used as receiving and transmitting "antennas." Sound-absorbent material is placed along the walls of the tanks as needed, to prevent undesired reflections and standing waves.

The nominal carrier frequency is 200 kHz, corresponding to an acoustic wavelength of about 7 mm. This short wave length allows farfield operation in a relatively small tank, and permits strong reflections from small objects.

Two interesting conditions that have been studied extensively are the case where all the received signal is due to reflection from random scatterers, and the case where a steady (specular) component is added to the randomly-scattered signal. The random scattering is produced by reflecting the signal off a column of air bubbles; the specular component is produced by placing a solid reflecting object near the bubble column.
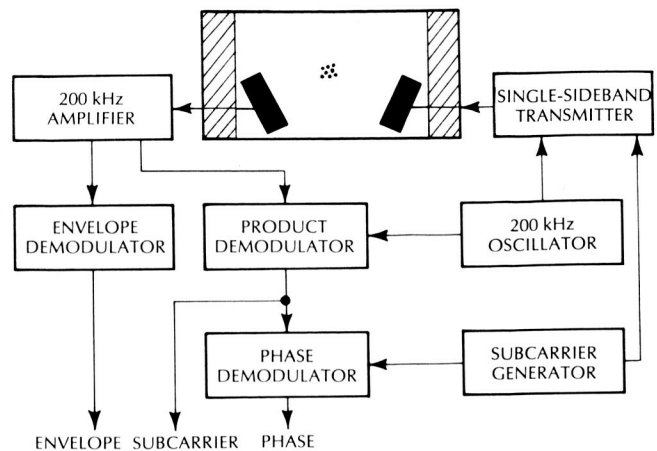
The water tank (figure 2)

This unit was constructed from 3/4-inch top-grade marine plywood; the corners and bottom are reinforced with 2-inch angle iron. The seams are caulked with Dow SIL-ASTIC clear sealer, and the inside walls are coated with an epoxy-based marine paint. An occasional addition of swimming-pool chlorinating compound keeps the water free of algae, etc., so that water changes are infrequent. A garden-hose siphon to a floor drain provides an adequate method for water removal.

Acoustic transducers (figure 3)

Identical units are used for transmitting and receiving. They are ordinary acoustic transducers used in marine depth finders (APELCO Model 1443A00), mounted on movable stands. They exhibit an impedance of 200 ohms and a lateral 3-dB beamwidth of ±6.5 degrees.

Turbulence generators, reflectors, absorbers

Rapidly varying (10-30 cps) random scattering can be obtained by bouncing the acoustic waves off a column of air bubbles that is generated by a group of three or four ordinary aquarium aerators. Several aerators may be supplied from a single small reciprocating air pump (the entire bubble system, with control valves and plastic tubing, was purchased at a local dime store). Aluminum plates and strips on movable stands serve as specular reflectors. Two-inch-thick batts of rubberized horse hair serve as effective (and surprisingly durable) signal absorbing pads.

A more slowly changing turbulence may be created by using an electric heating element or motor-driven vanes. (The suggestion of using a small school of piranha-fish was discarded, due to the obvious hazards to operating personnel.)
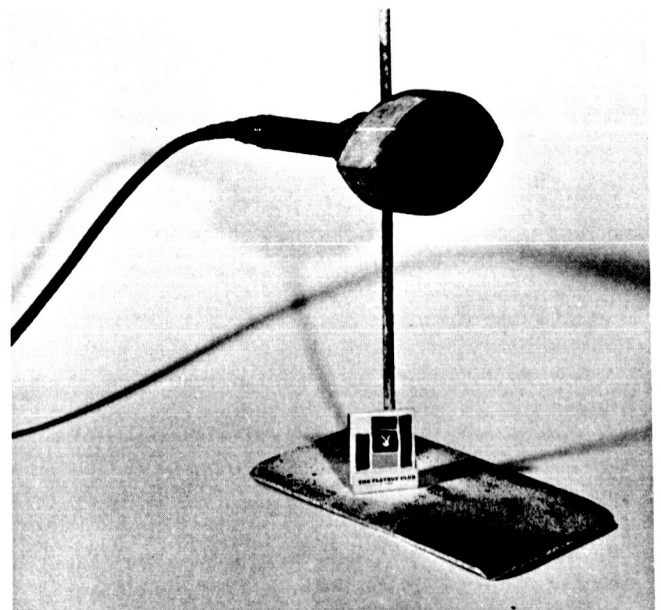

Figure 1 — Typical tank setup, including SSB transmitter and receiver

Figure 2 — Photograph of water tank

Figure 3 — Photograph of acoustic transducer

CHANNEL CHARACTERISTICS:
TYPICAL SETUPS

The transmitting transducer must be driven by a signal of the order of 3-4 V peak-to-peak; a suitable driving circuit is shown in figure 4. Channel measurements indicate that the attenuation (expressed in terms of the ratio between the transmitting and receiving transducer voltage levels) may be estimated by the following formula

Attenuation in dB $= (10 + 0.35D)$,   where D is in inches Path lengths of over 40 inches have been successfully used.

Figure 5a shows a typical setup to obtain an almost completely random scatter mode of transmission, with little or no specular component. A sample of the received signal is shown in figure 6; it typically exhibits 5-10 deep fades per second and 20-30 minor fades per second.

Figure 5b shows the method for obtaining a strong specular component. A solid object (e.g., a metal strip) is placed in front of the air bubbles. This permits independent adjustment of the specular and random reflection components over a reasonably wide range. It is essential that the bubble generators, transducers, specular reflectors and absorbing pads be carefully positioned to insure elimination of secondary reflection components. It has been found that the extraneous signal content in the simulated channel can be made negligible (—60 dB or less).[6]

Figure 4 — Transducer driving circuit

(a)

A TYPICAL COMMUNICATION
SYSTEM MODEL (figure 1)

The communication system currently used in connection with the multipath simulator is shown in figure 1. The transmitter has a crystal-controlled carrier frequency of 200 kHz. This carrier is single-sideband (SSB) modulated with a 4 kHz subcarrier. The resulting SSB signal is fed to the transmitting transducer, whence it is radiated in the water. The receiving transducer picks up the signal reflected from the air bubbles and other reflecting objects placed in intersection of the two transducer patterns. This signal is amplified and demodulated in the receiver. The receiver performs both envelope and product demodulation. The output from the envelope demodulator provides information about the fading of the envelope of the received signal. The output from the product demodulator is used to extract information about the phase of the fading signal. Reference signals for the product demodulator and for the subcarrier phase comparison are obtained directly from the transmitter by means of cables. This results in one great additional advantage of using the multipath simulator rather than "the real thing." Since the reference signals used at the receiver can be obtained directly from the transmitter, they do not have to be generated locally at the receiver. This eliminates a great source of uncertainty as to whether observed effects are due to the channel behavior or due to time-, frequency-, and phase-reference errors at the receiver.
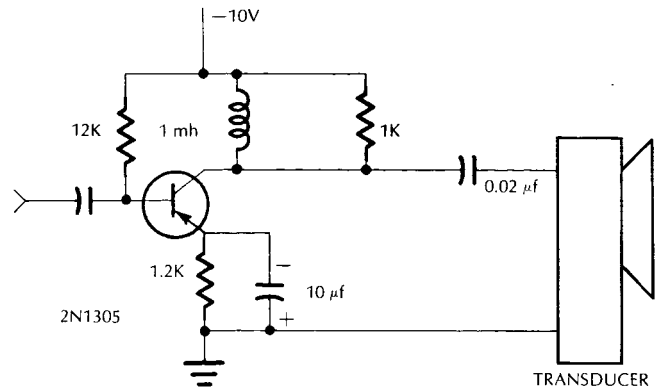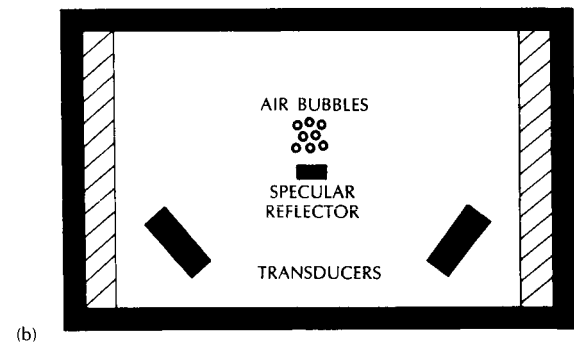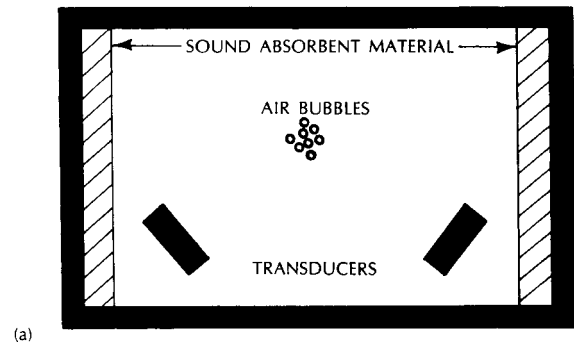

(b)

Figure 5 — Two types of multipath channels: (a) pure random scattering, (b) random scattering plus strong specular
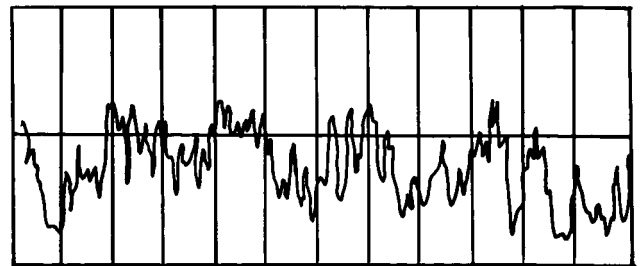

Figure 6 — Amplitude record of the fading carrier envelope, pure random scattering

Statistical analyzer

Relatively simple statistical analyzing equipment may be used to study many properties of a typical communication model. Figure 7 shows block diagrams of the systems used for measuring *amplitude distributions and binary (two-level) correlation functions*. The design of these systems was based upon the following characteristics of the received signals:

a. Over a period of several hours, the statistical properties of the received signals are essentially *stationary*, thus permitting time-averaged measurements.

b. Many of the signals for which auto- and cross-correlation measurements were desired are approximately Gaussian with zero mean; this permits the results of simple binary (two-level) correlation measurements to be used to reconstruct the true analog correlation.[6] Two-level correlation techniques are also self-normalizing, *i.e.*, they yield the normalized correlation coefficient of the ac component of a Gaussian signal directly, so that precise monitoring of signal levels is not required during a test.

c. Signal bandwidths of the order of 10 to 30 Hz permit the use of either continuous or sampled-data averaging techniques where desired.

Amplitude distribution analyzer

(see also reference 3, section 5-12)

The system shown in figure 7a performs a continuous averaging of the output of a dual-slicer circuit to derive an estimate of the amplitude probability density function $f(X)$, where

$$f(X) = \text{Prob } (X - \delta X < x \leq X + \delta X)$$

by measuring $(1/\Delta X) \cdot \text{Prob } [X - (\Delta X/2) < X < X + (\Delta X/2)]$; ΔX small compared to the variance of X.

The rate-of-change of the ramp X_o and the output filter time constant, $T = 10 C$ (T in seconds, C in μf), must be chosen for the particular process to be measured. In particular we must choose K and C such that (for "low-pass" processes):

$$K = dX_o/dt << \Delta X/T \text{ v/sec}$$

and

$$T = 10 C_{mfd} >> N_o \simeq 1/f_o$$

where ΔX is the total width of the slicer window $= 0.2$ v

$T = 10C$ is the output filter time constant

f_o = approximate bandwidth of the process $x(t)$ in Hz.

(*i.e.*, N_o, the zero-crossing rate of x is of the *order* of $1/f_o$)

In our case, f_o is of the order of 10 Hz. Successful distribution plots were made with $C = 2\ \mu f$, $T = 20$ sec. The ramp rate was kept small compared to $\Delta X/T = 0.05$; typically $K = 0.005$ v/sec. Faster sweep rates cause excessive distortion of the plot due to filter time lag. Wider output filter bandwidth causes the "noise" in the plot to be excessive, due to inadequate averaging time. The input process should be scaled so that its rms value is about 3 volts, and its mean is close enough to zero so that the range of
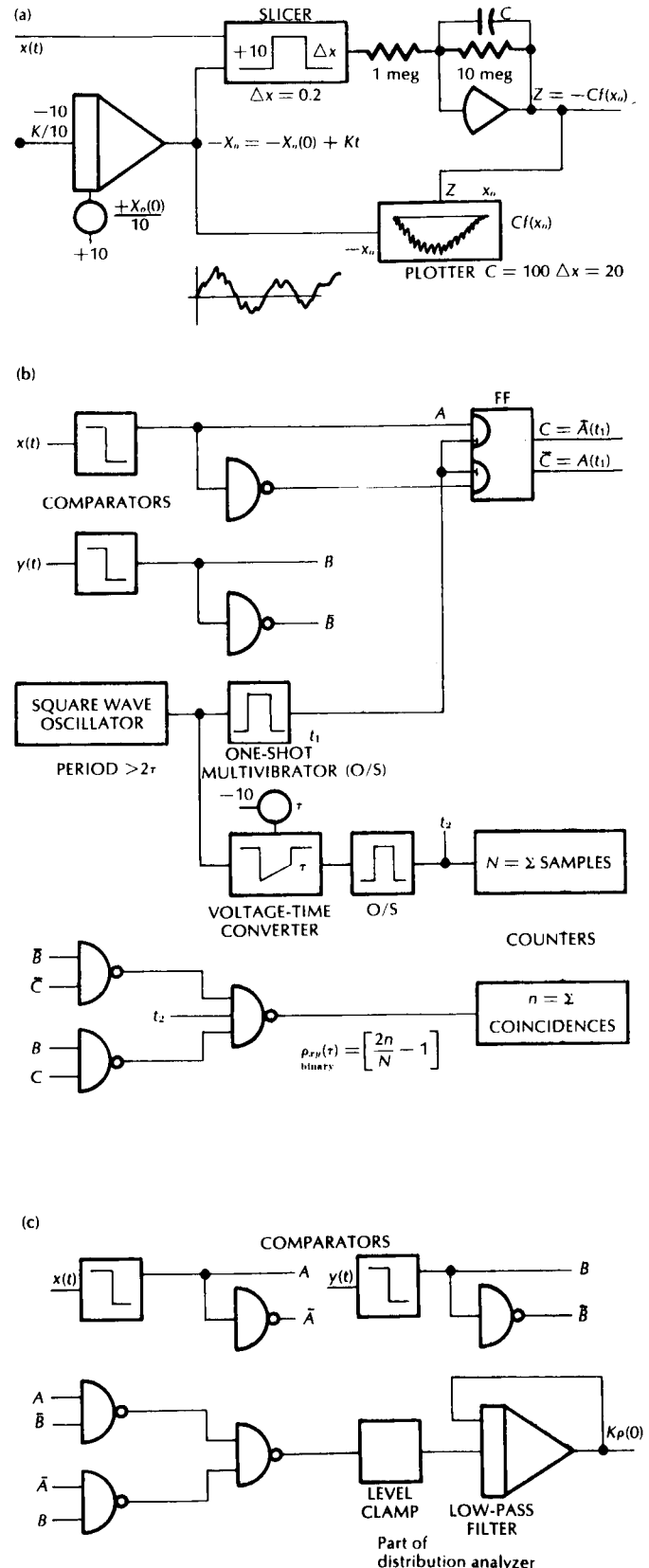

Figure 7 — Block diagrams of statistical measuring systems: (a) amplitude distribution analyzer, (b) sampled-data binary correlator, (c) binary cross-power estimator

the process is within $\pm 10\,\mathrm{v}$ range of the sweep voltage. For Gaussian- and Rayleigh-distributed signals this is straightforward. A complete plot typically requires $200/0.005 = 4000$ seconds (1 hr, 15 min)! This demonstrates the need for good stationarity of the channel parameters and the advantages of being able to simulate a channel on as fast a time scale as possible. This type of distribution measurement proved suitable for our case, but would be cumbersome, if the fading rates were appreciably slower.

Binary correlator (figure 7b; see also reference 5 and reference 3, section 6-5)

The binary (or polarity-coincidence) correlator is suitable for making estimate of the auto- and cross-correlation functions of zero-mean Gaussian stationary random processes. It was useful in our case for studying the phase components of the receiver signal, but would not be valid for performing correlation analysis of envelope signals (which may nevertheless be studied with a low-frequency spectrum analyzer).

The binary correlator yields an estimate of the two-level or one-bit correlation function

$$\rho_{xy}(\tau)_{\text{binary}} = E\left[\operatorname{sgn} x(t_1)\,\operatorname{sgn} y(t_2)\right] = \frac{2}{N}(\Sigma\, n_i - 1); \quad \tau = t_2 - t_1$$

where $n_i = 1$ if $x(t_1)$ and $y(t_2)$ have the same polarity on the ith sample.

It may be shown that, if $x(t)$ and $y(t)$ are zero-mean Gaussian processes

$$\rho_{xy}(\tau)_{\text{analog}} = \sin\frac{\pi}{2}\,\rho_{xy}(\tau)_{\text{binary}} = \frac{E\left\{[x(t) - x_{dc}]\,[y(t+\tau) - y_{dc}]\right\}}{(X_{\text{rms}}\,Y_{\text{rms}})}$$

Again, considerable time is required for measuring a correlation function, since at least 2000 samples must be taken for each value of τ. In our case, τ ranged from 5 milliseconds to 0.1 second. If the actual unnormalized correlation function is desired, one must also make careful measurements of the mean and rms values of $X(t)$ and $Y(t)$ during the measurement period; then one can estimate the correlation function by

$$R_{xy}(\tau) = X_{dc}Y_{dc} + X_{\text{rms}}Y_{\text{rms}}\rho_{xy}(\tau)_{\text{analog}}$$

Cross-power measurement (figure 7c)

In many cases, we are interested in only $\rho_{xy}(0)$, and an analog averaging method may be used in place of the sampled-data method used in the correlator above. Here the output is related to

$$\rho_{xy}(\tau = 0)_{\text{binary}} = \frac{2}{\pi}\arcsin\rho(0)_{\text{analog}}$$

$$\rho(0)_{\text{analog}} = \frac{E\left\{[x(t) - X_{dc}]\,[y(t) - Y_{dc}]\right\}}{(X_{\text{rms}}\,Y_{\text{rms}})}$$

Although these simple hybrid statistical analysis systems used in the initial studies proved satisfactory for studying many basic channel properties, faster, more accurate, and more sophisticated measurements could be made with a small general-purpose digital computer and analog/digital converter interface. Faster-than-real-time processing of channel output signals could be accomplished by using a precision instrumentation tape recorder with servo speed control to record the signals and a reference carrier for playback at a faster tape speed.

CONCLUSIONS

A multipath simulator has been described that is readily constructed and flexible in operation. The multipath characteristics of the channel can be varied over a wide range of conditions, allowing simulation of nearly all desired types of multipath channels.

Since the transmitter and receiver are located close together, reference and synchronization signals required at the receiver can, if desired, be obtained by cables directly from the transmitter. Difficulties with the local reference signals can thus be avoided, allowing a more exact investigation of the effects of multipath conditions on the transmitted signal and on the operation of the receiver.

REFERENCES

1 H E GULLSTAD   W K ROBERTS
   A radio propagation simulator using radio-frequency acoustic waves in a liquid
   FCC Laboratory Division Project No 32801 May 18 1965

2 A M MANDERS
   A multipath simulator for use in evaluation of spacecraft ranging and communication systems
   Proceedings Third Space Congress Cocoa Beach Florida March 1966

3 G A KORN
   Random-process simulation and measurement
   McGraw-Hill New York 1966

4 M SCHWARTZ
   Information transmission modulation and noise
   McGraw-Hill New York 1959

5 C L BECKER   J V WAIT
   Two-level correlation on an analog computer
   Transactions IRE/PGEC Vol EC-10 No 4 pp 752-758 Dec 1961 (Also Aug 1962)

6 E J CLAIRE   L LERNER   D M NEWLANDS   J G STOIDES
   A system analysis of the GENESYS multipath simulator utilizing a receiver with a unique phase-locked loop design
   Unpublished Laboratory Report
   University of Florida GENESYS April 18 1966

7 G A KORN   T M KORN
   Electronic analog and hybrid computers
   McGraw-Hill New York 1964

Solution of optimal control problems on a high-speed hybrid computer

by RICHARD L. MAYBACH
University of Arizona

RICHARD MAYBACH grew up in the paper-mill town of Camas, Washington on the lower Columbia River. From there he went south to Corvallis, Oregon and a BS degree in electrical engineering from Oregon State in 1959. Then, ignorant of the effects of thermal shock on physical bodies, he moved to Tucson, Arizona to enroll in graduate school. It was during this time that he was introduced to Dr. G. A. Korn, who said, "Glad to know you; here is your project." Nothing was seen of Dick for several days while he attempted to find out: (1) What is a servo multiplier? (2) How is the thing on this chassis (purportedly a servo multiplier) supposed to work? (3) Why doesn't it? Presumably he found out, for he received his MS degree in 1961.

Dick then entered the Army Signal Corps and was assigned to the research and development lab at Fort Monmouth. As project officer he worked on a contract to develop input/output equipment compatible with the Army's FIELData family of computers. The time was spent designing digital pattern generators and running operational and environmental tests on the delivered equipment. He learned two very valuable things there—logic design and that everything fails in the humidity chamber; 134° F. and 99% humidity will corrode even those shiny gold 2nd Lt's bars.

In 1963 he returned to Arizona to find that the Analog Computer Lab had become the Analog/Hybrid Computer Lab. It was back to logic design, but at least the debugging could be done in air-conditioned comfort. During this time he has been involved in a number of logic and circuit design projects, including some of the subsystems of ASTRAC II. Some of his current work is described in this paper, which forms a portion of his PhD dissertation.

Mr. Maybach is a member of SCi, IEEE, Tau Beta Pi, Eta Kappa Nu, and Sigma Xi.

EDITOR'S NOTE

Since the opening of the *Computers in education* section last May, all the articles published have been written either by educators or established professionals in the field. The gist of most of these articles has been the use of analog or digital computers as teaching aids and as problem-solving tools, and the emphasis has been on undergraduate work. Now the time has come to have a look at graduate use of analog and hybrid equipment, and since the student, rather than the professor or machine, is the star in this show, we present here an article showing what can be done by a student. Such work is an outstanding example of the payoff that can be expected from initiating undergraduate students to analog and hybrid computers.

ACS

SUMMARY

This paper presents a method for finding the solutions to minimum-time optimal control problems. The procedure is to implement Pontryagin's Maximum Principle on an iterative hybrid computer. The state and adjoint equations as well as the control law are simulated using conventional analog components. The troublesome two-point boundary-value problem, which is always associated with Maximum Principle, is solved by iteration, using a digital parameter optimizer. Thus, a manual trial-and-error search for the proper initial values of the adjoint variables is unnecessary.

We show that, for a large class of systems, it is not necessary to generate the Hamiltonian, because the necessary condition that it normally must satisfy is redundant. This allows many problems to be greatly simplified. We also present an optimizing routine that solves the boundary-value problem. This permits the proposed method to be used on any hybrid computer that incorporates a general-purpose digital computer.

The solutions to two problems show that the proposed method is feasible. Average convergence times range from less than one second to about 70 seconds. These vary with the initial conditions on the state variables. The examples were solved using ASTRAC II, a small (40 amplifier), high speed (up to 1000 solutions per second), iterative hybrid computer with only modest component accuracy (0.25 per cent).

Although the discussion and examples are limited to a minimum-time performance index, the method is easily extended to cover other criteria.

1. PONTRYAGIN'S MAXIMUM PRINCIPLE[1]

Consider a dynamical system whose state is governed by n state equations

$$\frac{dx_i}{dt} = \dot{x}_i = f_i(x_1, x_2, \cdots, x_n, u_1, u_2, \cdots, u_r)$$

$$= f_i(x, u), \quad i = 1, 2, \cdots, n \qquad (1.1)$$

where the x_i are *state variables*, and the u_j, *controls*. Admissible u_j are bounded, piecewise-continuous functions of time. In this paper we will consider only cases in which both the *initial state*, $x(0) \equiv [x_1(0), x_2(0), \cdots, x_n(0)]$, and *final state*, $x(T) \equiv [x_1(T), x_2(T), \cdots, x_n(T)]$ are given. We will further restrict ourselves to consideration of cases in which we desire to move the system from its given initial state to its given final state in such a way that the time, T, is minimum.

Define the $n + 1$ *adjoint equations* as

$$\dot{p}_0 = 0, \quad \dot{p}_i = -\sum_{j=1}^{n} p_j \frac{\partial f_j}{\partial x_j}, \quad i = 1, 2, \cdots, n \qquad (1.2)$$

where the p_i are *adjoint variables*. Also define the Hamiltonian as

$$H = p_0 + \sum_{j=1}^{n} p_i f_i(x, u) \qquad (1.3)$$

Note that this is valid only for the minimum-time case. The Maximum Principle is, of course, applicable to a much wider range of problems.

It is now possible to state *Pontryagin's Maximum Principle*. Let $u_j(t)$, $j = 1, 2, \cdots, n$, be an admissible set of controls which transfers the state from its initial position, $x(0)$, to the final position, $x(T)$, and let $x_i(t)$, $i = 1, 2, \cdots, n$ be the corresponding trajectory. In order that the $u_j(t)$ and $x_i(t)$ yield the optimal solution to the problem, it is necessary that there exist nonzero continuous functions $p_i(t)$, $i = 1, 2, \cdots, n$, corresponding to the $x_i(t)$ and $u_j(t)$ such that:

1. For every t, $0 \le t \le T$, the function H of the variables u_j attain its maximum (with respect to the u_j) at the point $u_j = u_j(t)$;

2. At the terminal time, T, the relations $p_0(T) \le 0$, max $H(T)$ (with respect to u_j) $\le H^*(T) = 0$ are satisfied.

This principle provides only the necessary conditions for a minimum time, T. It will single out from the continuum of trajectories joining the given initial and final states, isolated candidates for the optimum. Hopefully this number will be small. The Maximum Principle is applicable only to systems where:

1. $f_i, \partial f_i/\partial x_j, i, j = 1, 2, \cdots, n$ are defined and continuous;

2. $u_j(t), j = 1, 2, \cdots, r$ are bounded piecewise-continuous functions.

2. A SIMPLIFICATION THEOREM

If a system can be described by equations of the form

$$\dot{x}_i = a_i(x_1, x_2, \cdots, x_n) + \sum_{j=1}^{r} b_{ij}(x_1, x_2, \cdots, x_n) u_j$$

$$= a_i(x) + \sum_{j=1}^{r} b_{ij}(x) u_j, \quad i = 1, 2, \cdots, n, \qquad (2.1)$$

where the controls, u_j, are subject to the constraints

$$|u_j| \le L_j, \quad j = 1, 2, \cdots, r, \qquad (2.2)$$

the functions a_i are such that

$$a_i(x_{1f}, x_{2f}, \cdots, x_{nf}) = 0, \quad i = 1, 2, \cdots, n, \qquad (2.3)$$

and the state is to be moved from a given initial state, $x(0) = x_0$, defined by $x_1(0) = x_{10}$, $x_2(0) = x_{20}, \cdots, x_n(0) = x_{n0}$ to a given final state, $x(T) = x_f$, defined by $x_1(T) = x_{1f}$, $x_2(T) = x_{2f}, \cdots, x_n(T) = x_{nf}$, in minimum time, T, then the condition that $H^*(T) = 0$ is redundant.

Proof:

From equations (1.1), (1.3), and (2.1),

$$H = p_0 + \sum_{i=1}^{n} p_i \dot{x}_i = p_0 + \sum_{i=1}^{n} p_i \left[a_i(x) + \sum_{j=1}^{r} b_{ij}(x) u_j \right]$$

$$= p_0 + \sum_{i=1}^{n} p_i a_i(x) + \sum_{j=1}^{r} u_j \sum_{i=1}^{n} p_i b_{ij}(x).$$

H is maximized with respect to the u_j, due to equation (2.2), if

$$u_j = L_j \, \mathrm{sgn} \left[\sum_{i=1}^{n} p_i b_{ij}(x) \right].$$

$$H^* = p_0 + \sum_{i=1}^{n} p_i a_i(x)$$

$$+ \sum_{j=1}^{r} L_j \, \mathrm{sgn} \left[\sum_{i=1}^{n} p_i b_{ij}(x) \right] \sum_{i=1}^{n} p_i b_{ij}(x)$$

$$= p_0 + \sum_{i=1}^{n} p_i a_i(x) + \sum_{j=1}^{r} L_j \left| \sum_{i=1}^{n} p_i b_{ij}(x) \right| \qquad (2.4)$$

The only condition on p_0 is that $p_0(T) \le 0$. This is satisfied if

$$p_0(T) = p_0 = -\sum_{j=1}^{r} L_j \left| \sum_{i=1}^{n} p_i(T) b_{ij}[x(T)] \right|,$$

then

$$H^*(T) = \sum_{i=1}^{n} p_i(T) a_i[x(T)] \qquad (2.5)$$

If the boundary condition $x(T) = x_f$ is satisfied,

$$a_i[x(T)] = 0$$

and $H^*(T) = 0$, regardless of the values of the $p_i(T)$. Thus the theorem is proved.

There are now $2n$ equations, n state equations and n adjoint equations, in the system. There are also $2n$ boundary conditions, the n given initial and the n final values for the state variables. The n initial conditions can be directly set, but the n final conditions must be satisfied by properly adjusting the n initial values of the adjoint variables and the time, T. Clearly one of these adjustments must be redundant. Note that nothing in the above has affected the adjoint equations (1.2). If $p_1, p_2, \cdots, p_n$ is a solution, so is $Ap_1, Ap_2, \cdots, Ap_n$ for any constant A. We can thus arbitrarily fix the magnitude of the initial value of one of the adjoint variables. However, the signs of all the p_i are important due to (2.4). We have reduced the boundary-value problem from one involving an $(n+1)$-dimensional search (over $p_1(0), p_2(0), \cdots, p_n(0), T$) to one involving only an n-dimensional search plus a binary decision.

All this is of no great theoretical interest, *but it results in a considerable saving in computer setup complexity. First, it is no longer necessary even to generate the Hamiltonian, much less enforce any boundary condition on it. Furthermore, by reducing the dimension of the search, a significant improvement is made in the convergence of the boundary-value enforcement process.*

3. THE COMPUTER

The ASTRAC II system is used in the configuration shown in figure 1. The state and adjoint equations are simulated on high-speed analog elements at solution rates of up to 1000 per second. During the optimization process (search for the correct values of the adjustable parameters), the analog simulation generates a value of a criterion function for each set of trial parameter values. The criterion function is

$$f(t) = -\sum_{i=1}^{n} \alpha_i \, | \, x_i(t) - x_{if} \, | \qquad (3.1)$$

where the α_i are weighting factors, and the x_{if}, the given final conditions. The optimizer searches for sets of values of the adjustable parameters that maximize $f(T)$, the final value of $f(t)$. Note that the maximum value is zero, and that when this is attained, the boundary-value problem is solved. After the optimization process has converged, the analog portion will generate and display the optimal trajectory and control.
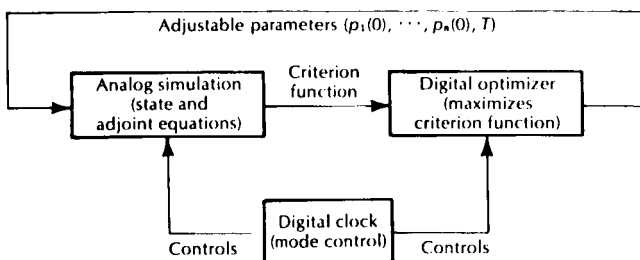
Adjustable parameters $(p_1(0), \cdots, p_n(0), T)$

Analog simulation (state and adjoint equations) — Criterion function → Digital optimizer (maximizes criterion function)

Digital clock (mode control)

Controls

Figure 1 – Simulation of Maximum Principle on ASTRAC II

The parameter optimizer's[2] search for the maximum value of $f(T)$ is made by performing a modified random walk with the values of up to four parameters. The initial value of the step size, $\triangle$, is 10 volts, and the dynamic range is ± 10 volts. The value of each parameter is perturbed by one of the values $+\triangle$, 0, or $-\triangle$, which are selected at random. The (boundary value) criterion function, $f(T)$, is then computed for the new parameter values. If $f(T)$ is larger than f_{max}, the largest value previously obtained, then the trial is termed a success, and the perturbed parameter values become the operating point for the next perturbation. On the other hand, if $f(T) < f_{max}$, the trial has failed. In this case the unsuccessful perturbations are discarded, and the parameters return to their previous values for the next perturbation.

This perturb-test procedure continues until a number of consecutive failures result. The step size, $\triangle$, then halved, and the process continues until $\triangle$ reaches its smallest value (about 80 mv). Here a test is made to determine if the process has converged to a local maximum. The true maximum is known to be zero, and if f_{max} is significantly less than this, the whole process starts anew with $\triangle = 10$ volts.

The internal operation of the optimizer is best described by flow diagrams, figures 2 and 3. There are two ways in which the computer can enter the optimization routine from analog computation. At time $t = T$, $f(T)$ has been formed, and the success-failure test is made. If however, a computing element overloads prior to this time, computation ceases, and the trial is judged a failure.

If the trial is a success $[f(T) > f_{max}]$, the sequence of operations proceeds as on the right of figure 2. The success counter is advanced, the failure counter reset (only consecutive events are counted), and f_{max} becomes the present value of $f(T)$. If the number of successes, n_S, has reached a preset number, N_S, the step size, $\triangle$, is doubled. $\triangle$ is stored as a single "1" in an eight-bit shift register. Thus "shift left" is equivalent to "double $\triangle$." This ability to increase the step size was found to have no significant effect in the examples discussed in this paper.

If the trial is a failure, things are more complex. First the unsuccessful perturbations are removed. The failure counter is advanced, and the success counter reset. If the process has not yet converged $(\triangle > \triangle_{min})$, $\triangle$ is halved when n_F, the consecutive number of failures, reaches the (preset) number N_F. If the process has converged, f_{max} is tested to see if it is a local maximum. The largest possible value of $f(T)$ is zero. If f_{max} is less than a preset number r $(r < 0)$, it fails the local maximum test. After N_M consecutive failures, all parameter values are reset, $\triangle$ is made large, f_{max} made small, and the maximization process starts anew.

The lower right portion of the flow diagram sets $f_{max} = f(T)$ after N_F consecutive failures if the process has converged to the global maximum. f_{max} is scored in an analog sample-hold circuit which, of course, is subject to drift. This periodic resetting prevents any drift from causing large errors in the value f_{max}. By allowing the optimizer to continue its local search after convergence, we enable it to

track a slowly moving maximum. This might arise due to a time-varying system or to slowly changing initial state values.

Figure 3 shows the process of actually perturbing the parameters, a_i. A digital noise generator (3) is stepped to develop a random word, which in turn generates the perturbation δ_1. δ_1 may have one of three values, $+\triangle$, 0, or $-\triangle$. This value is stored, and the process repeated for δ_2, δ_3, and δ_4. The optimizer contains a provision for correlat-

ing the δ_i with the success of the previous run. That is, altering the probabilities so that the perturbations, δ_i, tend to move the a_i in the same direction as in the preceding run if it was successful, but in the opposite direction as in the preceding run if it was a failure. This feature was not found useful in this study.

After all the δ_i and a_i have been developed, the computer again enters its analog computation phase to compute the resulting $f(T)$.

Figure 2 — Optimizer flow diagram, determination of step size

Figure 3 — Optimizer flow diagram, parameter perturbation

4. SECOND-ORDER EXAMPLE

Consider the system $\ddot{x} = u$, $|u| \leq 1$, $\dot{x}(T) = x(T) = 0$
with a minimum-time performance index. Identify
$x_1 = x$ and $x_2 = \dot{x}$. Then

$\dot{x}_1 = x_2$, $\dot{x}_2 = u$	(state equations)	(4.1)
$H = p_0 + p_1 x_2 + p_2 u$	(Hamiltonian)	(4.2)
$\dot{p}_0 = 0$, $\dot{p}_1 = 0$, $\dot{p}_2 = -p_1$	(adjoint equations)	(4.3)
$u = \text{sign } (p_2)$	(control law)	(4.4)
$f(t) = -\alpha_1|x_1(t)| - \alpha_2|x_2(t)|$	(criterion function)	(4.5)

The considerations of section 2 apply here, and we can let

$$p_1 = \pm 1. \qquad (4.6)$$

The problem now is to find the values of $p_2(0)$ and T and the sign of p_1 that cause $x_1(T) = x_2(T) = 0$. The analog simulation is shown in figure 4.[4] All integrators are controlled by the signal R'' ($R'' = 1$ implies reset). Electronic switch number 1 causes $10p_1$ to be $+10$ volts or -10 volts, depending on the most significant bit of the optimizer's parameter a_1. Digital-to-analog converter number 2-11 converts the parameter a_2 (an eight-bit digital word) to the analog signal $p_2(0)$. The control, u, is developed by comparator number 6 followed by a precision diode-bridge

limiter. The limiting level is accurately adjusted to $= \pm 1$ volt with pot number 13.

Figure 5 shows the conversion circuits needed to present the results of a trial to the optimizer, as well as the circuit that generates R''. Integrator number 23 is controlled by R, which is a logical 0 for 1 msec, then a logical 1 for 1 msec and repeat. The output of this integrator is a -10 to $+10$ volt ramp. Parameter a_1 then determines the time at which the output of comparator number 4-22 undergoes a positive transition and hence the time interval for which $R'' = 0$ (compute). Thus a_3 becomes the time T. The signals R and V are combined in NAND gate number 18. $\bar{R}$ holds the integrators in reset while integrator number 23 is reset. V holds them in reset after an overload develops. This signal is generated by the optimizer from the signal $\overline{OV}$.

The signal $\bar{S}_2$ is normally a logical one. Some fixed time prior to time $t = 0$, it moves to zero, thus placing track-hold number 28 in track. This is necessary as the operation of the sample-holds becomes erratic if their track periods are too short. Just after time $t = 0$, $\bar{S}_2$ becomes a one, thus returning the control of the sample-hold to R''. Sample-hold number 28 moves from track to hold at time T, storing $f(T)$. $f(T)$ is compared to f_{max} by comparator number 5-30.

Notes: All integrator gains = 20,000
——▶Denotes digital input

Figure 4—Simulation of second-order example

Notes:
—6 volts = logical one
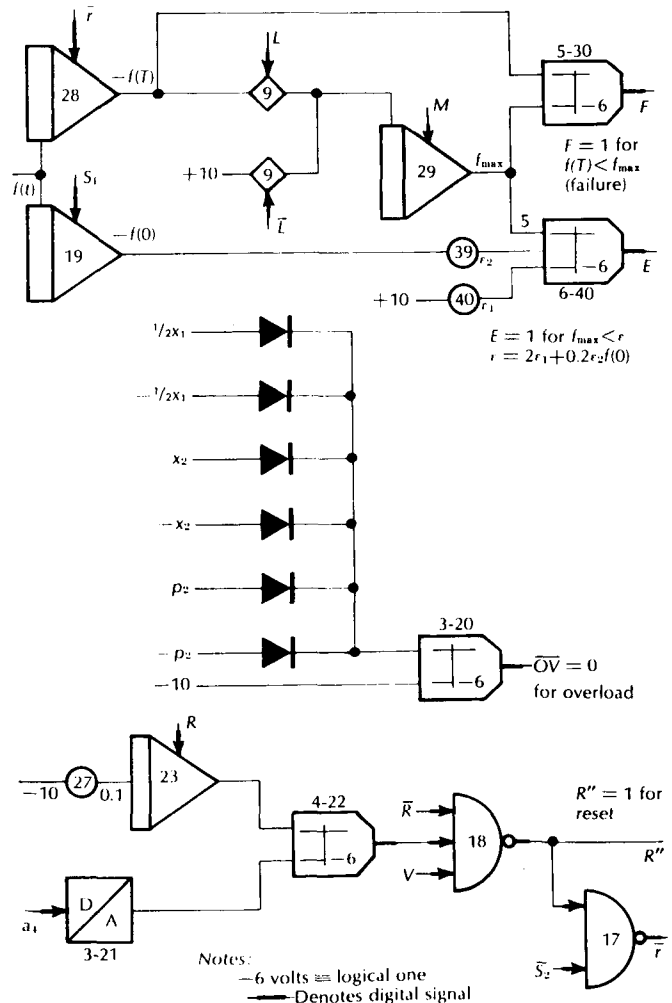——Denotes digital signal

Figure 5—Conversion circuits

The output F informs the optimizer of the success or failure of the run. If the run was successful, the optimizer forms $L = 1$, $M = 1$ which causes $f_{max} = f(T)$. The combination $L = 0$, $M = 1$ is used to set $f_{max} = -10$ (section 3).

The signal S_1 causes sample-hold number 19 to store the value of $f(t)$ at $t = 0$. This permits ε to be adaptive. As $x_1(0)$ and $x_2(0)$ become larger, so does T^*, the optimum T. As T^* becomes larger, $f(T)$ becomes more sensitive to small changes in the parameters. Thus ε must also increase to prevent the resulting jitter in $f(T)$ from being interpreted as a local maximum.

The contour of $f(T)$ for the initial condition $x_1(0) = 0$, $x_2(0) = 5$ is shown in figure 6. The true maximum is located
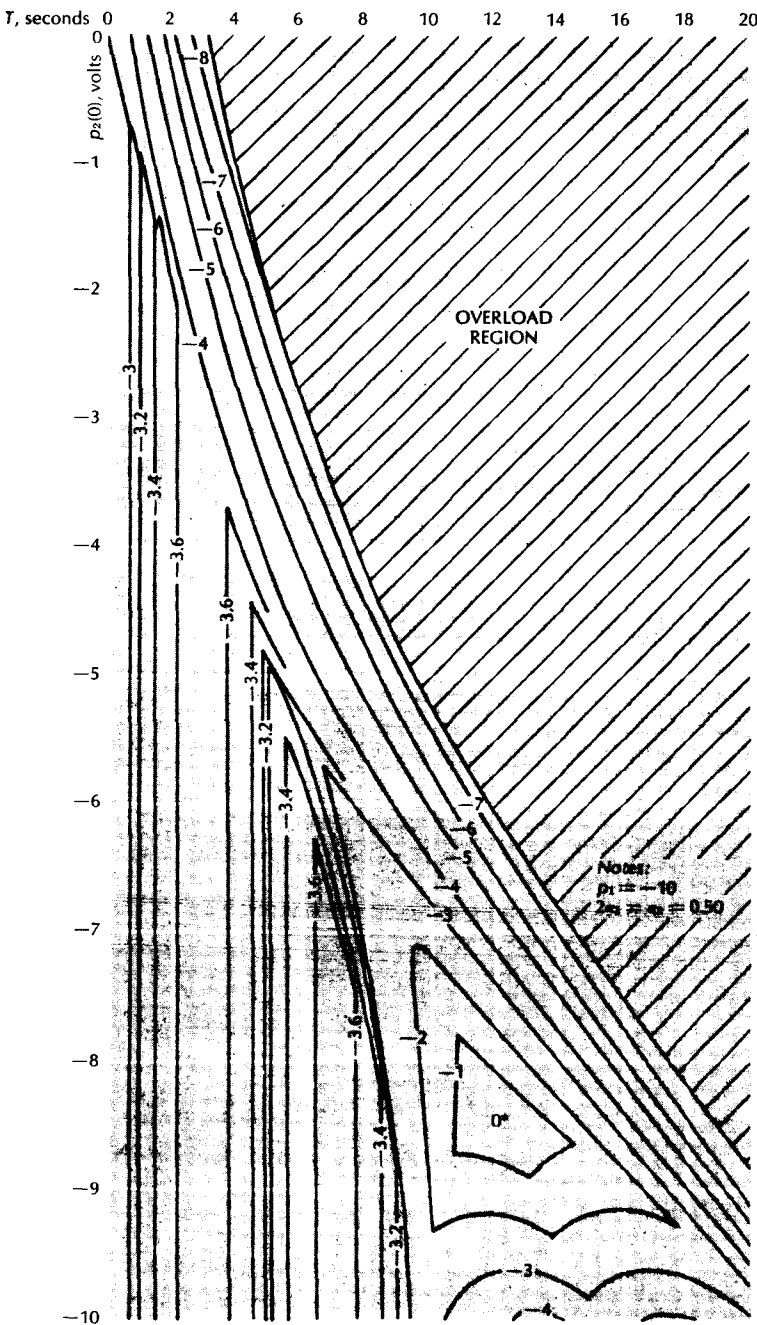
Figure 7

Figure 8

Figure 6—Contour of $f(T)$ for $x_1(0) = 0$, $x_2(0) = 5$
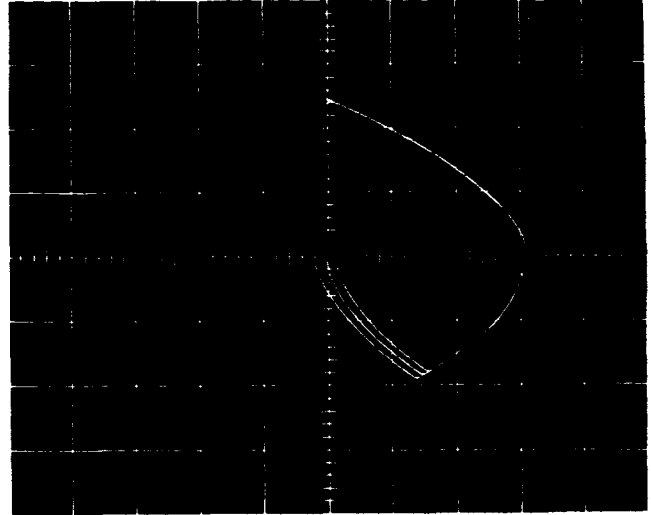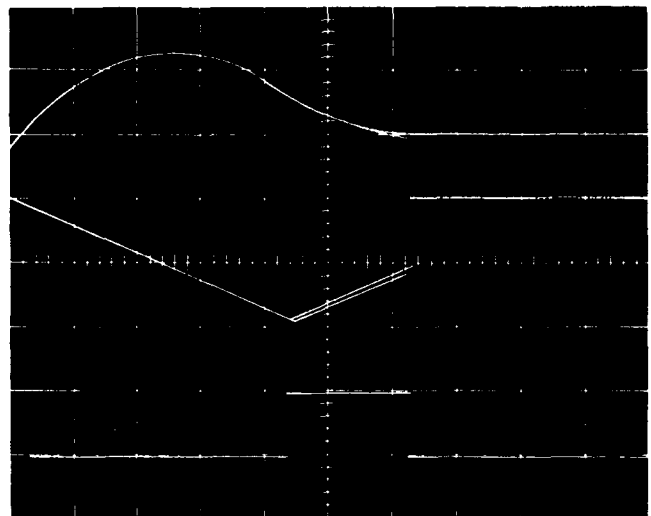
at $T = 12$ seconds, $p_2(0) = -8.4$. The only effect of $p_2(0)$ is to alter the time, t_s, at which the control changes sign. t_s is located on a straight line connecting the origin and point $(10, -10)$. To the left of this line, $f(T)$ is independent from the value of $p_2(0)$. Note the local maxima, $T = 0$ and $T = 5$, $-10 \leq p_2(0) \leq -5$. Because of these, a hill-climbing technique is not assured of success. However, since the value of the true maximum is known, convergence to a local maximum is easily determined. A hill-climbing technique with a random starting point and a local maximum rejection feature will eventually succeed.

Figure 7 shows a phase-plane display of the response to the initial condition $x_1(0) = 0$, $x_2(0) = 5$. $0.5x_1$ is displayed horizontally and x_2 is displayed vertically, both at 2 volts/cm. The search is shown continuing at the smallest step size (about 80 mv). Note the sensitivity of the final values to small changes in the initial conditions of the adjoint variables. Figure 8 shows the time-response to the same initial condition. The top trace is $0.5x$ at 5 volts/cm, the middle is x_2 at 5 volts/cm, and the bottom one is $5u$ at 10 volts/cm. Real time is 2 seconds/cm, and computer time is 0.1 msec/cm.

Table I compares measured and theoretical response times for two different iteration rates. For these data the optimizer was disconnected, and p_1, $p_2(0)$, and T were set by hand.

Table I—Comparison of theoretical and measured response times

		Theoretical		Measured, 500 rps		Measured, 50 rps	
$x_1(0)$	$x_2(0)$	t_c	T	t_c	T	t_c	T
			12.07		12.08	8.46	12.15
			9.65		9.64	6.84	9.64
			7.24		7.32	5.12	7.28
			4.83		4.84	3.40	4.88
			2.41		2.40	1.72	2.40
			6.33		6.36	3.12	6.36
			5.66	2.84	5.72	2.84	5.68
			4.90	2.44	4.96	2.48	4.96
			4.00	1.96	4.00	2.00	4.00
			2.82	1.44	2.84	1.40	2.84

Table II shows average convergence times for various initial conditions.

Table II—Average convergence times

$x_1(0)$	$x_2(0)$	Convergence time, seconds	Number of iterations
	5	0.8	400
	4	2.0	1000
	3	3.5	1750
	2	70.2	35,100
	1	28.0	14,000
2	0	9.9	4950
4	0	12.3	6150
6	0	22.6	11,300
8	0	4.7	2350
10	0	3.1	1550

Notes: $a = 0.05 + 0.1f(0)$
$N_p = 5$
$N_M = 60$
Times are averaged over 10 to 20 trials.

5. THIRD-ORDER EXAMPLE

Consider the system

$$\ddot{x} = u, |u| = 1, \ddot{x}(T) = \dot{x}(T) = x(T) = 0$$

again with a minimum time performance index. Let $x_1 = x$, $x_2 = \dot{x}$, and $x_3 = \ddot{x}$, then

$\dot{x}_1 = x_2, \ \dot{x}_2 = x_3, \ \dot{x}_3 = u$	(state equations)	(5.1)
$H = p_0 + p_1 x_2 + p_2 x_3 + p_3 u$	(Hamiltonian)	(5.2)
$\dot{p}_0 = 0, \ \dot{p}_1 = 0, \ \dot{p}_2 = -p_1$	(adjoint equations)	(5.3)
$\dot{p}_3 = -p_2$		
$u = \text{sgn} \ (p_3)$	(control law)	(5.4)
$f(t) = -\alpha_1 |x_1(t)| - \alpha_2 |x_2(t)|$		
$\quad -\alpha_3 |x_3(t)|$		
	(criterion function)	(5.5)

The considerations of section 2 also apply here. The simulation is shown in figure 9, and the comments of figure 4 apply. The simulation-to-optimizer conversion is identical to that of the second-order case, shown in figure 5, except that there are four more diodes associated with comparator number 3-20 to detect overloads of x_3 and p_3. Time responses for two different initial conditions are shown in
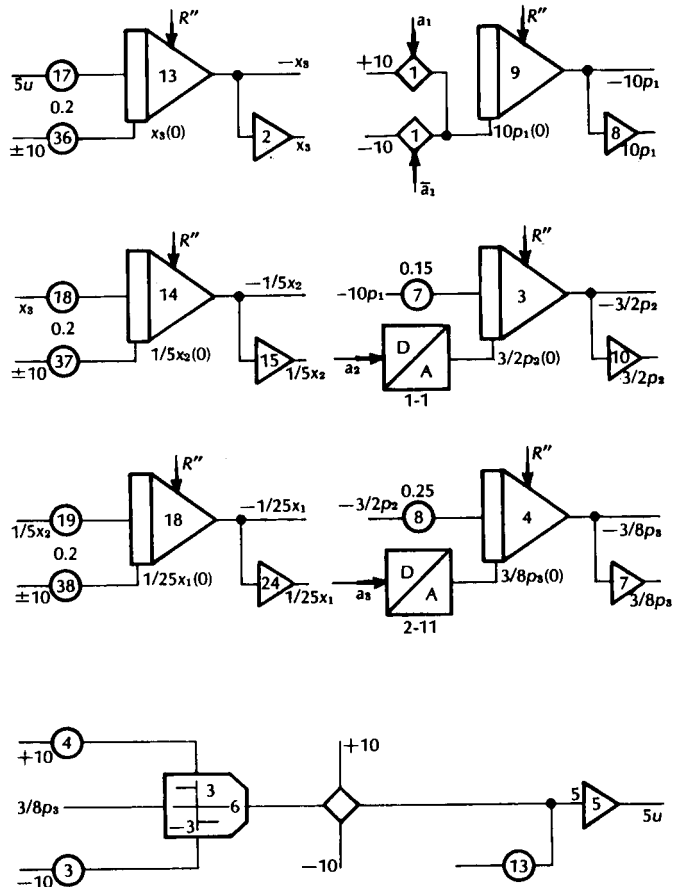

Figure 9—Simulation of third-order example

figures 10 and 11. The traces from the top are $0.04x_1$, $0.2x_2$, x_3, and $5u$. The top three are at 2 volts/cm, and the bottom is at 10 volts/cm. The real time scale is 2 sec/cm, and computer time is 0.1 msec/cm. Average convergence times for the initial condition $x_1(0) = 50$, $x_2(0) = x_3(0) = 0$ is 24.6 seconds (12,300 iterations).

Any large change of these values requires rescaling of the adjoint variables. The problem here is that $f(T)$ is very sensitive to small changes in the initial conditions of the adjoint variables. This condition is so extreme that it is very difficult to distinguish local maxima. An eight-bit word just does not offer a sufficiently small minimum step size.

ACKNOWLEDGMENT

6. CONCLUDING REMARKS

The above experiments have shown that this method of solving optimal control problems is feasible. There are three major problems to be solved before it can be regarded as practical. The sensitivity of the solutions to the initial values of the adjoint variables requires that the latter be adjustable with a high resolution. The eight-bit word length used for these experiments must be increased. Also, table II indicates that the present optimization strategy is inefficient.

In addition to these practical problems, there is an unresolved theoretical difficulty. Pontryagin's Maximum Principle states that of the $u(t)$ that move the state of the system so as to satisfy the given boundary conditions, we want to choose those that maximize H. Our proposed method does not solve this problem. We state that of the $u(t)$ that maximize H, we want to choose those that move the state so as to satisfy the given boundary conditions. As far as the author knows, these two problems have not been shown to be equivalent.

In section 3, we noted that even after the process has converged to the optimal solution, we allow the optimizer to continue a local search. As a result, one can slowly vary an initial condition or a parameter of the system and continuously observe the effects of this change on the optimum solution. The rapport between the problem and the investigator that has always been of such value in analog simulation is now available in optimal control studies. The advantages of this are difficult to exaggerate.
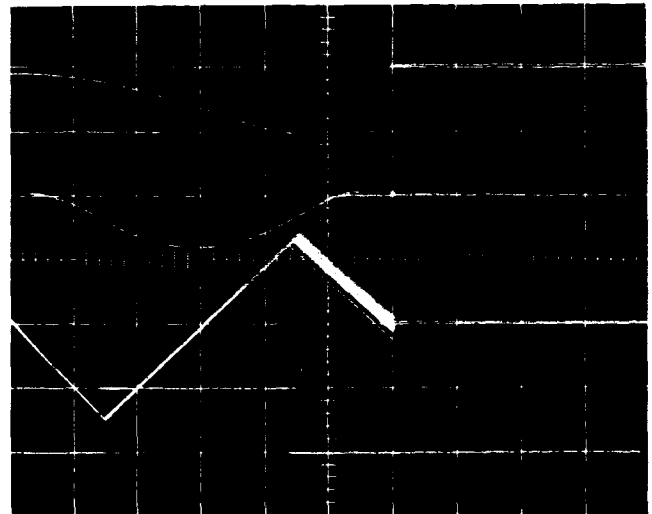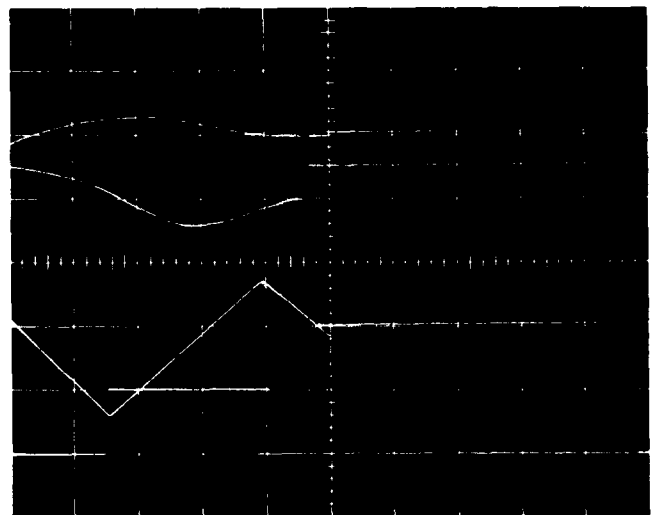

Figure 10

Figure 11

REFERENCES

1 PONTRYAGIN et al
The mathematical theory of optimal processes
Translated by K N Trirogoff
Interscience Publishers New York 1962

2 B A MITCHELL
A hybrid analog-digital parameter optimizer for ASTRAC II
Proceedings 1964 Spring Joint Computer Conference
SIMULATION vol 4 no 6 June 1965 pp 399-411

3 R L T HAMPTON
A hybrid analog-digital pseudo-random noise generator
Proceedings 1964 Spring Joint Computer Conference
SIMULATION vol 4 no 3 March 1965 pp 179-187

4 Report
Uniform graphics for SIMULATION
SIMULATION vol 6 no 3 March 1966 pp 137-140

GENERATION OF INVERSE FUNCTIONS
BY THE METHOD OF STEEPEST DESCENT *

by R.L. MAYBACH **

ABSTRACT. — The technique of steepest descent can be used to generate the inverse of a function with standard analog or hybrid computer components. The resulting system is often free from the instability problems that plague inverse function generation by implicit computation. When the method is applied to forming division from multiplication a four-quadrant divider results. Neither of the two most common penalty functions, error squared and absolute error produce wholly satisfactory operation. If the loop gain is allowed to increase without bound, it causes relay or bang-bang type of performance that is free from the objectionable response of the previous systems.

Because of their cost, relatively few special purpose function generators, usually multipliers, and sine and cosine generators, are available on most analog computer patchbays. All other functions must be laboriously set up on general purpose diode function generators. Even if the diode function generator is of the programmable type, the available number of programs is limited. It is worthwhile then to investigate methods of generating inverse functions $[y = f^{-1}(x)]$ from the function itself $[x = f(y)]$. This will provide division, square root, arcsine, and arccosine from available special purpose function generators.

The generation of inverse functions can be accomplished by the use of a function generator in an amplifier feedback loop (fig. 1) [1]. This system
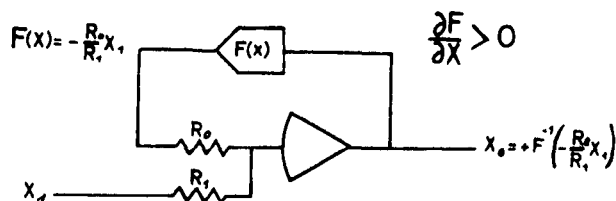

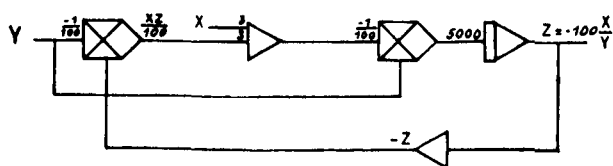Fig. 1. — Generation of inverse Functions by implicit computations.

Fig. 2. — Block Diagram for System 1,

$$\phi = \frac{1}{2} \left(\frac{YZ}{100} + X \right)^2$$

often exhibits high-frequency oscillation due to the phase shifts of the amplifier and function generator. Instability is especially likely if the function generator is of the carrier type, for example, a time-division or triangle-integration multiplier. Devices such as these often use sharp cutoff, high phase-shift filters to

maximize bandwidth. It is not possible to obtain the inverse by this means if the function generator is sufficiently perverse. This problem alone is serious enough to justify more complex methods of generating inverse functions.

Methods of Steepest Descent.

Consider a cost function $\phi = \phi(x_1, ..., x_n)$. This may be minimized with respect to each x_i by forcing $\dot{x}_i = -K(\partial \phi/\partial x_i)$ [1, 2]. The speed of response is clearly proportional to K. This method has the limitation that ϕ must have no local minima. Fortunately such ϕ can often be found when this technique is applied to generating inverse function.

Suppose we wish to generate $y = f^{-1}(x)$ given $x = f(y)$. Two cost functions have been suggested,

$$\phi_1 = \frac{1}{2} [x - f(y)]^2 \text{ and } \phi_2 = |x - f(y)| \quad [1].$$

These can be minimized if $f(y)$ increases monotonically with y. If y can be varied to reduce ϕ to zero, then the desired inverse has been found, since $\phi = 0$ is true only if $y = f^{-1}(x)$. The resulting expressions for $\dot{y}$ are :

$$\dot{y} = K [x - f(y)] (\partial f/\partial y) \text{ if } \phi = \frac{1}{2} [x - f(y)]^2,$$

$$\dot{y} = K \text{ sign } [x - f(y)] (\partial f/\partial y) \text{ if } \phi = |x - f(y)|.$$

Since $f(y)$ is known $\partial f/\partial y$ can be generated explicitly.

As K is increased the response time will decrease. The dynamic characteristics will be most favorable if K can be made large. But if $\dot{y}$ becomes too large the integrator that is attempting to generate y will overload. Thus, we wish to let $K \to \infty$ under the constraint that $|\dot{y}| \le A$ (constant). If we apply this to the above we find that

$$\dot{y} = K \text{ sign } (\partial f/\partial y) \text{ sign } [x - f(y)].$$

The same relation results from either ϕ. This is interesting for two reasons. The system is always driving at full speed toward the minimum. The indicated multiplication of sign functions can be performed digitally by an exclusive or rather than by an expensive analog multiplier.

To find the possible advantages of each scheme, the 3 systems above were applied to the problem of generating $z = -100 \ (x/y)$ from $x = -(1/100) \ yz$. This has the interesting additional property that the result is a four-quadrant divider which is stable for $x = y = 0$.

System 1. $\phi = \dfrac{1}{2} \ (\dfrac{yz}{100} + x)^2$.

In this case $\dot{z} = -\text{K} \ (\dfrac{yz}{100} + x) \ (y/100)$. The computer set-up for this is shown in figure 2. The gains shown result in a stable system and allow x to undergo a 40 volt step without causing the summer to overload. Larger steps could, of course, be accommodated (at the expense of risetime) if the summer's gain were reduced. The static accuracy is essentially that of the multipliers, and the dynamic characteristics are shown in figure 3.
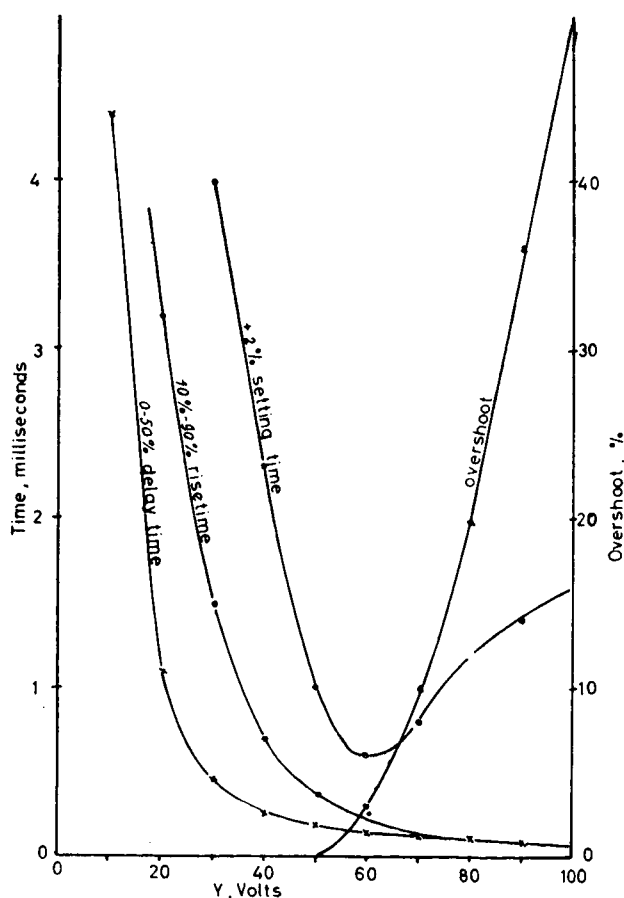

Fig. 3. — Transient Response of System 1,

$$\phi = \tfrac{1}{2} \ (\frac{\text{YZ}}{100} + \text{X})^2$$

Although the response times are independent of x, they do vary with y, and below 50 volts the system becomes relatively sluggish. Also, the scheme requires 2 expensive multipliers instead of the one required by most dividing schemes.

System 2. $\phi = (yz/100) + x$.

For this case $\dot{z} = -\text{K} \ (y/100) \ \text{sign} \ (\dfrac{yz}{100} + x)$. One of the multipliers of system 1 is replaced by a comparator and a switch. This is much cheaper, especially since the switch need not be particularly accurate. The resulting computer diagram is shown in figure 4.

The static accuracy of this system is again essentially that of the multiplier. The dynamic response is markedly different from that of system 1. Instead of resembling a linear second-order system, the step response is a ramp with slope proportional to y. The
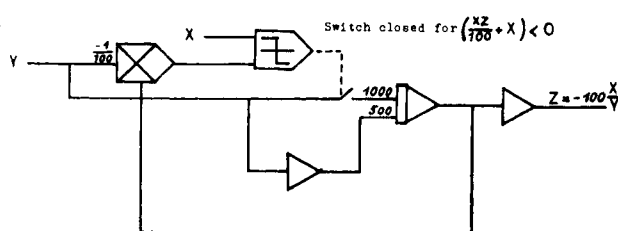

Fig. 4. — Block Diagram for System 2,

$$\phi = (\frac{\text{YZ}}{100} + \text{X})$$

slope is approximately $\dfrac{1}{2} \ y$ volts/millisecond. The 0-100 percent rise-time shown in figure 5 is roughly comparable to the $\pm$ 2 percent settling time of figure 3. Comparison of these two figures shows that system 2 has the same sluggish response at low values of y that troubles 1.

System 3. $\text{K} \rightarrow \infty$.

For this case $\dot{z} = -\text{A} \ \text{sign} \ (\dfrac{yz}{100} + x) \ \text{sign} \ (y)$.

Define two binary variables U_ϕ and U_y.

$$\text{U}_\phi = \text{sign} \ (\frac{yz}{100} + x),$$

$$\text{U}_y = \text{sign} \ (y).$$

Then in Boolean algebra notation

$$\dot{z} = -\text{A} \quad \text{if} \ \text{U}_\phi \cdot \overline{\text{U}_y} + \overline{\text{U}_\phi} \cdot \text{U}_y = 0,$$

$$\dot{z} = +\text{A} \quad \text{if} \ \text{U}_\phi \cdot \overline{\text{U}_y} + \overline{\text{U}_\phi} \cdot \text{U}_y = 1.$$

The resulting system is shown in figure 6.

The step response of this system is a ramp with a slope of about 55 volts/millisecond. The 0-100 percent rise-time is shown in figure 7. Comparison of
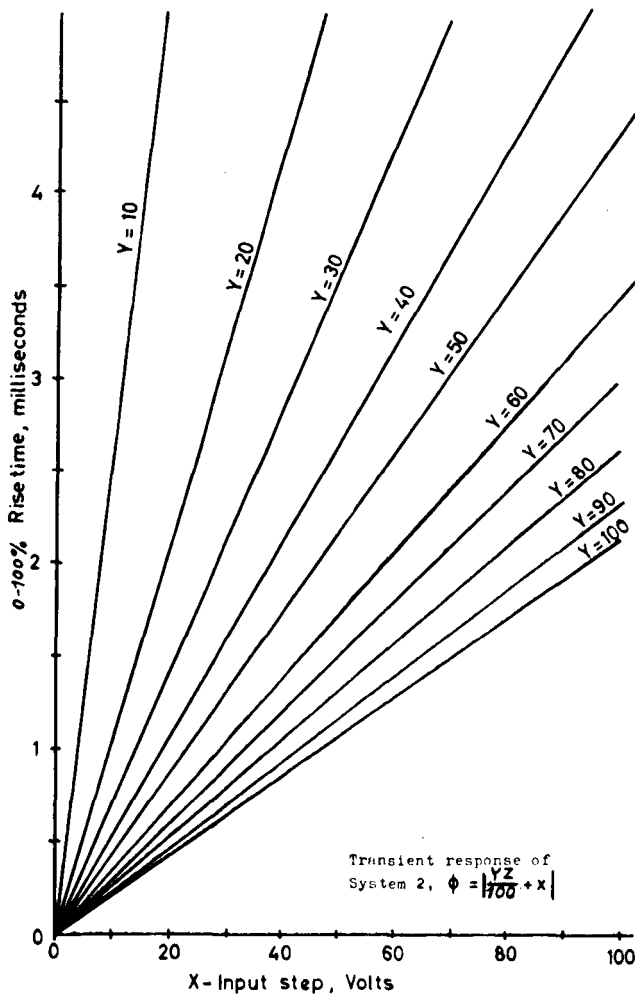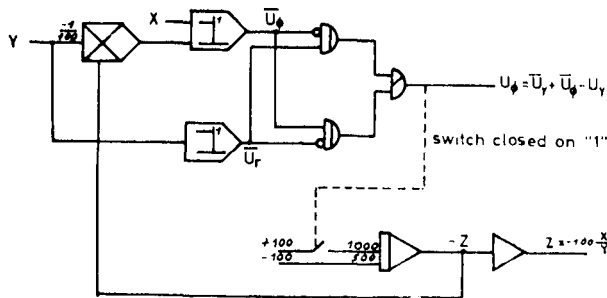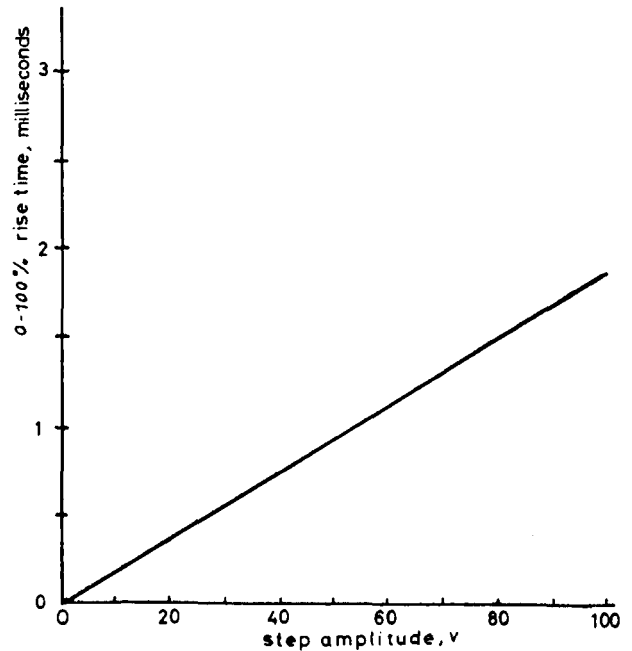
Fig. 5. — Transient Response of System 2,

$$\phi = (\frac{YZ}{100} + X)$$

Fig. 6. — Block Diagram for System 3, K → ∞

figures 3, 5, and 7 shows that system 3 has the most desirable transient response. Only system 3 is not sluggish at low values of y. Its static accuracy is again essentially that of the multiplier.

It will be interesting to see the effect of an error in the multiplier. Suppose that $x = (1/100)\, yz + \varepsilon$. where ε is the error. Then $z = -100\, \dfrac{x}{y} + 100\, \dfrac{\varepsilon}{y}$

Fig. 7. — Transient Response of System 3, K → ∞

The statement that the accuracy of the divider is the same as that of the multiplier is misleading. It would be more accurate to say that the error is multiplied by $100/y$. This is true of any system that obtains division implicitly from multiplication and is not a special defect of steepest descent.

Practical Considerations.

Quarter-square multipliers and Philbrick USA - 3 operational amplifiers are used in all of the systems.

System 1 is straight-forward, and no unusual precautions are necessary. The impedance level must be kept low, of course, in order to realize reasonably fast response.

A detailed circuit diagram of system 2 is shown in figure 8. Amplifier 1 forms a comparator with about ± 0.6 volt levels. These are large enough to reliably operate the switch on the input to amplifier 4. A comparator with higher switching levels will also serve, but it will introduce excessive noise into the input. This is because in the steady-state the comparator switches rapidly between its 2 levels and some feed-through is inevitable with the simple switch used here. The 500-1000 pf trimmer on the comparator is needed to keep it stable. Note the use of speed-up capacitors on amplifier 2, which could be replaced by a simple digital inverter.

A detailed block of system 3 is shown in figure 9. This discussion of system 2 also applies here. The diode marked with an asterisk is not necessary logically. Because of the low levels, however, there must be an equal number of diodes facing each way in all paths so that the saturation voltages cancel. If the extra diode is removed, the comparator levels must be increased which will raise the noise level.

REFERENCES

[1] Korn, G.A. and T.M. Korn : *Electronic Analog and Hybrid Computers*, McGraw-Hill, New York, 1964.

[2] Kelley, H.J. : Method of Gradients, in Leitmann, G., *Optimization Techniques with Application to Aerospace Systems*, Academic Press, New York, 1962.
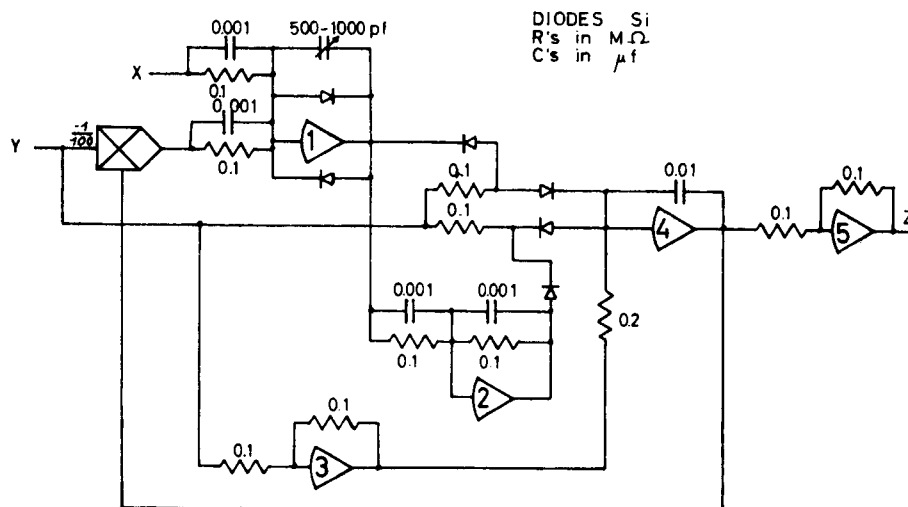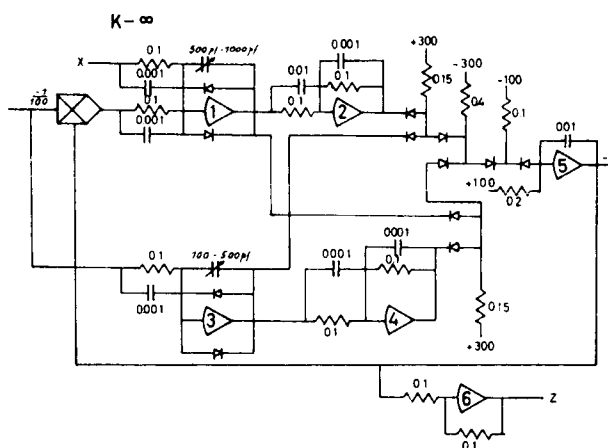

Fig. 8. — Circuit Diagram of System 2,

$$\phi = (\frac{YZ}{100} + X)$$

Fig. 9. — Circuit Diagram of System 3, K → ∞

Performance of Operational Amplifiers With Electronic Mode Switching*

G. A. KORN†

Summary—The linear-circuit performance equation for an important class of mode-switched operational amplifiers is derived to identify errors caused by finite feedback loop gains, switch resistance, and follower-amplifier source impedance. Errors due to capacitor dielectric absorption, switching spikes, dc leakage, and current limiting are discussed with reference to an electronically switched integrator (sample-hold or analog-memory circuit), and various design requirements are listed.

INTRODUCTION

ELECTRONICALLY-SWITCHED operational amplifiers of the type shown in Fig. 1 switch the input voltage and/or the mode of operation (transfer function) of an accurate operational amplifier with a single switch operated by associated analog comparators and/or digital circuits. The best-known examples of such techniques are electronically controlled integrators or analog-memory (sample-hold) circuits[1] and single-pole double-throw switches in modern iterative analog computers and related hybrid analog/digital computing equipment (Fig. 2). The input impedances

Fig. 1.

Fig. 2.

[1] T. Brubaker, "Precision analog memory has extended frequency response," *Electronics*, vol. xx, pp. 141–143; September 29, 1961.

Z_1 and Z_1' can be replaced by summing networks, and electronic commutators can be built with multiple switches.[2]

When the electronic switch in Fig. 1 is OFF, the input X_1 and the feedback impedances Z_1, Z_0 are essentially out of the circuit, which now functions as an ordinary operational amplifier with input X_1'. When the electronic switch is ON, the low-impedance follower-amplifier circuit connects the input X_1 and effectively shorts the input and feedback current through Z_1' and Z_0'.

The earliest circuits of this type employed simple cathode followers instead of the feedback amplifier followers indicated in Fig. 1; more accurate unity-gain followers (high-gain difference amplifiers with unit feedback) are now generally substituted to obtain reduced drift and lower output impedance.

A notable advantage of the arrangement of Fig. 1 over other mode-switching circuits is that the input diode limiter prevents signal voltages at the switch from ever exceeding five to ten volts, so that the electronic switch can be operated directly from low-voltage off-the-shelf-digital-computer modules, and low-voltage high-performance switching diodes and transistors can be used.

As an added advantage, the switch is driven by a low-impedance source and the main-amplifier summing point itself is maintained at a low impedance with respect to ground while the switch is ON.

Both six-diode and six-transistor switches are used; either type combines shunt and series switching to produce open-circuit transfer impedances in excess of 10^8 megohm at dc, 2000 megohms at 100 kc, and 20 megohms at 1 Mc. Presently available switches, then, have almost negligible signal leakage; errors in the OFF condition are mainly due to imperfectly cancelled switching spikes and dc leakage currents in diodes or transistors.

LINEAR-CIRCUIT ANALYSIS

We shall leave the discussion of switch-unbalance effects until later and consider the switch as an almost ideal switch with infinite back resistance and forward resistance R_{ON} of the order of 20–100 ohms. The linear operation of our circuit is then described by the equivalent circuit of Fig. 1. If the unity-gain driver is a
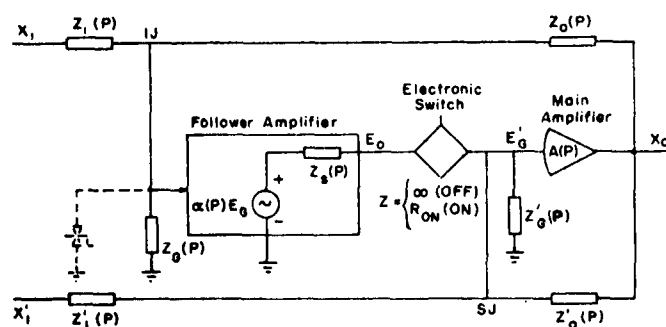
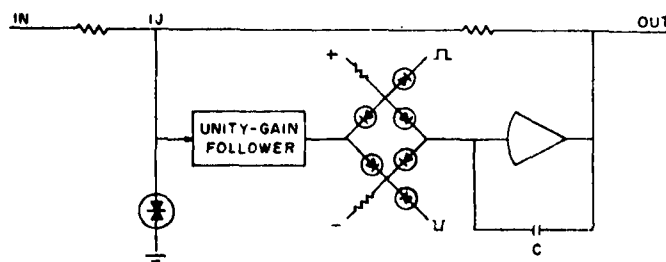[2] H. D. Huskey and G. A. Korn, "Computer Handbook," McGraw-Hill Book Co., Inc., New York, N. Y.; 1962.

high-gain feedback amplifier, α will be very close to one, and the internal impedance Z_S of the follower amplifier can be of the order of 10^{-2}–10 ohms at dc and 100–1000 ohms at 100 kc. Amplifier input impedances Z_G and Z_G' have been included to allow for input capacitances and input conductances, especially in the case of transistor circuits.

When the switch is OFF, its transfer impedance is, as noted before, practically infinite, so that we have an operational-amplifier circuit with the well-known performance equation,[2]

$$X_0 = \frac{A\beta_{\rm OFF}}{1 - A\beta_{\rm OFF}}\frac{Z_0'}{Z_1'}X_1'$$
$$= -\left(1 - \frac{1}{1 - A\beta_{\rm OFF}}\right)\frac{Z_0'}{Z_1'}X_1', \qquad (1a)$$

where

$$\beta_{\rm OFF} = \left(1 + \frac{Z_0'}{Z_1'} + \frac{Z_0'}{Z_G'}\right)^{-1}, \qquad (1b)$$

is the feedback ratio with the switch OFF.

When the switch is ON, its transfer impedance is essentially a resistance $R_{\rm ON}$ between 10 and 100 ohms, and we have

$$X_0 = \frac{A\beta_1}{1 - A\beta_1}Z_0'\left(\frac{\alpha}{Z_S + R_{\rm ON}}E_G + \frac{1}{Z_1'}X_1'\right), \qquad (2a)$$

where

$$\beta_1 = \left(1 + \frac{Z_0'}{Z_1'} + \frac{Z_0'}{Z_G'} + \frac{Z_0'}{Z_S + R_{\rm ON}}\right)^{-1} \qquad (2b)$$

is the feedback ratio reducing the gain of the main amplifier; Z_S is follower-amplifier source impedance, and

$$E_G = Z_0\left(\frac{X_1}{Z_1} + \frac{X_0}{Z_0}\right)\beta_{\rm ON}, \qquad (3a)$$

where

$$\beta_{\rm ON} = \left(1 + \frac{Z_0}{Z_1} + \frac{Z_0}{Z_G}\right)^{-1} \qquad (3b)$$

is the feedback ratio for the follower-amplifier input.

Combining (2) and (3), we find the desired performance equation in the switch-ON mode,

$$X_0 = -\left[1 - \frac{1}{1 - A_{\rm ON}\beta_{\rm ON}\dfrac{Z_0'\beta_1}{Z_S + R_{\rm ON}}}\right]$$
$$\left(\frac{Z_0}{Z_1}X_1 + \frac{Z_S + R_{\rm ON}}{\alpha\beta_{\rm ON}Z_1'}X_1'\right), \qquad (4a)$$

where

$$A_{\rm ON} = \frac{\alpha A}{1 - A\beta_1} \qquad (4b)$$

is the forward gain from the follower-amplifier input to the output of the main amplifier. If we consider $-(Z_0/Z_1)X_1$ as the correct output voltage, the error is

$$e = \frac{1}{1 - A_{\rm ON}\beta_{\rm ON}\dfrac{Z_0'\beta_1}{Z_S + R_{\rm ON}}}\frac{Z_0}{Z_1}X_1$$
$$- \left[1 - \frac{1}{1 - A_{\rm ON}\beta_{\rm ON}\dfrac{Z_0'\beta_1}{Z_S + R_{\rm ON}}}\right]\frac{Z_S + R_{\rm ON}}{\alpha\beta_{\rm ON}Z_1'}X_1', \quad (5)$$

which can usually be approximated by

$$e \approx -\frac{Z_S + R_{\rm ON}}{Z_0'\beta_{\rm ON}}\left(\frac{1}{A_{\rm ON}\beta_1}\frac{Z_0}{Z_1}X_1 + \frac{1}{\alpha}\frac{Z_0'}{Z_1'}X_1'\right). \qquad (6)$$

Discussion

Eqs. (4–6) indicate the importance of small $Z_S + R_{\rm ON}$ for accurate performance. $R_{\rm ON}$ will be between 10 and 100 ohms. The follower-amplifier source impedance Z_S may vary between substantially less than 0.01 ohm at dc and at most a hundred ohms at 1 Mc for high-quality solid-state follower amplifiers. If $Z_S + R_{\rm ON}$ is small compared to Z_0', Z_1', and Z_G', (2b) yields

$$\beta_1 \approx \frac{Z_S + R_{\rm ON}}{Z_0'}, \qquad (7)$$

so that (6) reduces to

$$e \approx -\left(\frac{1}{A_{\rm ON}\beta_{\rm ON}}\frac{Z_0}{Z_1}X_1 + \frac{Z_S + R_{\rm ON}}{\alpha\beta_{\rm ON}Z_1'}X_1'\right). \qquad (8)$$

Note that β_1, and hence $Z_S + R_{\rm ON}$, also affects $A_{\rm ON}$, so that reduction of $Z_S + R_{\rm ON}$ reduces *both* error terms in (8).

In all existing circuits of the type studied, the dc follower-amplifier gain $\alpha(0)$ is approximately 1, mainly because larger values of α might cause instability. Even a conditionally stable main amplifier and/or follower amplifier will exhibit ringing after each switching operation. Note also that, in the usual feedback-type follower circuits, any increase in α implies a proportional increase in the source impedance Z_S which should, if at all possible, be only a fraction of the switch resistance $R_{\rm ON}$.

Effects of Parasitic Impedances. Capacitor Dielectric Absorption

The effects of various nonideal computing impedances on operational-amplifier performance are discussed in detail by Single.[2] Noninductively-wound wirewound resistors can be used in summing amplifiers (and hence also as initial-condition resistors in switched integrators) operating below 1 Mc, since their small remaining parasitic inductance is tuned out by the distributed capacitance. It is recommended that such resistors be paralleled with 10–50 pf capacitors to improve frequency response and stability.[2] Integrator input resistors, on

the other hand, should have as little parallel capacitance as possible, so that film resistors are preferable for use in wideband repetitive analog computers. To reduce phase shift due to resistor and summing-point capacitances at frequencies above a few kilocycles, it is almost mandatory to employ feedback and summing resistances below 10 kc, so that the use of high-current transitor amplifiers is recommended.

While resistive capacitor leakage is usually negligible, capacitor dielectric absorption in high-quality polystyrene and teflon integrating or storage capacitors ($D<2\times10^{-4}$) will cause phase-shift and storage errors of the order of a few hundredths of one per cent for ordinary computing and holding times.[2] To visualize the effect of dielectric absorption on, say, a switched integrator (Fig. 2), imagine one or more series RC circuits across the main capacitance C. If the latter has been discharged for some time and is then quickly charged to a voltage E and placed into HOLD, the parasitic RC circuits will draw "relaxation" currents until their charges are equalized. Although errors due to capacitor dielectric absorption can be traced in a very accurate "slow" electronic analog computer, they will probably be masked by larger phase shift and switching errors in present-day iterative computers.

NONLINEAR EFFECTS: TRACK-HOLD CIRCUITS

In the TRACK/HOLD configuration of Fig. 2, the input X_1' is zero, and Z_0' is the impedance $1/CP$ of the storage capacitor. Z_1 and Z_2 are usually resistances of the order 10–100 K. For a given electronic switching circuit with a known average *error current*, i_c, due to switching spikes, leakage and dc offset, the storage capacitance C is determined by the required storage time T and the allowable absolute output-voltage error e, since

$$C = \frac{i_e T}{e},\qquad(9)$$

i_e appears to be between 10^{-3} and $10^{-2}\ \mu A$ for the best currently available diode and transistor switches.

With the value of C thus fixed, the magnitude of the signal current through the switch in TRACK is at least equal to the capacitor charging current. This current can become very appreciable when the unit is switched into TRACK, or when it tracks a high-frequency input voltage X_1. It follows that the high-frequency response of such sample-hold circuits is usually not determined by the small-signal bandwidth of the two amplifiers, but

rather by the ability of both amplifiers and switch to supply capacitor-changing current (rate limiting).

If, for example, the output of a sample-hold circuit is to track the input $a \sin \omega t$, then follower amplifier, main amplifier, and switch must supply a sinusoidal current of amplitude $A\omega C$. If, then, the 100 kc component of a signal has an amplitude of 10 volts, the peak charging current into a 0.001 μf capacitor is $2\pi\times10^5\times10 \times10^{-6}$ ma ≈ 6.3 ma. The follower amplifier supplies this current at a low voltage, but the main amplifier must supply the charging current, plus any current drawn by the load, at the full required output voltage.

Again, the time t required for the sample-hold output to follow a voltage step of magnitude E is not usually determined by the amplifier small-signal rise time, but by the largest current value i_{max} which can be supplied by the follower amplifier, the main amplifier, and the switch—whichever is smallest:

$$t = \frac{EC}{i_{max}}.\qquad(10)$$

In the case of diode-bridge switches (Fig. 2), note that the dc or pulse voltage supply used to hold the switch ON must also furnish at least the specified peak charging current, since the diode bridge will otherwise act as a current limiter; and the pulse circuit used to turn the bridge OFF must, again, be capable of furnishing currents of similar magnitude. This requirement is a distinct argument in favor of transistor switches for certain high-speed sample-hold circuits, since the current gain of the switching transistors materially reduces the control-current requirements.

In the author's opinion there is, however, still much to be learned about the choice of switching circuits for sample-hold and other-mode-switching devices; in particular, the relative advantages of various diode and transistor switches have not been conclusively established. It appears certain, however, that even the brute-force method of employing high-current amplifiers and switches to permit larger storage capacitances will ensure memory and reset accuracy within 0.1 per cent of FS for iterative differential analyzers. It is, in fact, possible to use all-electronic mode switching in "slow" electronic analog computers to obtain switching errors within 0.1 per cent of FS for at least 50-sec computes runs with 1 μf integrating capacitors.[3]

[3] G. A. Korn, "New high-speed analog and analog-digital computing techniques: the ASTRAC system," *Proc. 3rd AICA Conference*, Opatija, Yugoslavia, September, 1961; Presses Académiques Européennes, Brussels, Belgium; 1962.