THE OSCILLOGENIC INSTRUMENT

by

Peter Kip Mercure

Dissertation submitted to the Faculty of the

Virginia Polytechnic Institute and State University

in partial fulfuillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Chemical Engineering

APPROVED:

P. R. Rony, Chairman

_____   _____
A. A. Beex   M. E. Davis

_____   _____
V. Guruswamy   H. A. McGee Jr.

_____
H. M. McNair

June, 1986
Blacksburg, Virginia

THE OSCILLOGENIC INSTRUMENT

by

Peter Kip Mercure

(ABSTRACT)

A generalized chemical oscillator has been invented consisting of input
and output interfaces to a chemical system, with the appropriate feedback
external to the chemical system such that the system oscillates. The
oscillation frequency can be made a function of concentration, reaction
kinetics, transport phenomena, and other physical properties. The idea
was reduced to practice with an electronic system coupled to an
electrochemical system, and gave a frequency output linear with
concentration for a number of ions in solution. A general mathematical
model of the electrochemical system was devised and programmed in
FORTRAN on a digital computer, and a mathematical model of the
oscillogenic instrument was used to conceptually test the idea. The use
of recursive parameter estimation was also considered for this
instrument.

Acknowledgements

I would like to thank the many people at the Virginia Polytechnic Institute and State University at Blacksburg Virginia for their kind help over the years which this work took. Special thanks must go to the faculty of the Chemical Engineering department, who taught me the subjects that I went to graduate school to learn, and to the faculty of the Electrical Engineering department, who introduced me to a whole new area of knowledge in control and estimation theory. The VPI&SU Chemical Engineering department had faith in converting chemists to chemical engineers and supported much of this dissertation work by assistantships and a tuition grant. Full support was provided by a Cunningham scholarship for the 1983-84 academic year. The DuPont de Nemours Company provided a scholarship for the years 1982-84 as a part of an effort to encourage PhD's in Chemical Engineering. I must thank Professor Peter Rony for his guidance over the years, never losing faith in the power of a good idea, and always providing a source of enthusiasm about learning. Words cannot express my gratitude to my wife,   for waiting the long time that it has taken to bring this work to the light; she never ceased wishing this work finished. Thanks must go to   and   for sharing their home with me during the times I have been a transient in Blacksburg, and some thanks for recreational therapy must go to   . The names of all the people who have helped over the years is too long to list here. If you think you should have been listed by name, you are right and I apologize.

Table of Contents

Chapter 1. History and Motivation

"My father used to invoke the ultimate authority by saying to me 'It stands written!' But then I read what stood written, imagine a human author, infer his values, and finally agree or disagree with him."
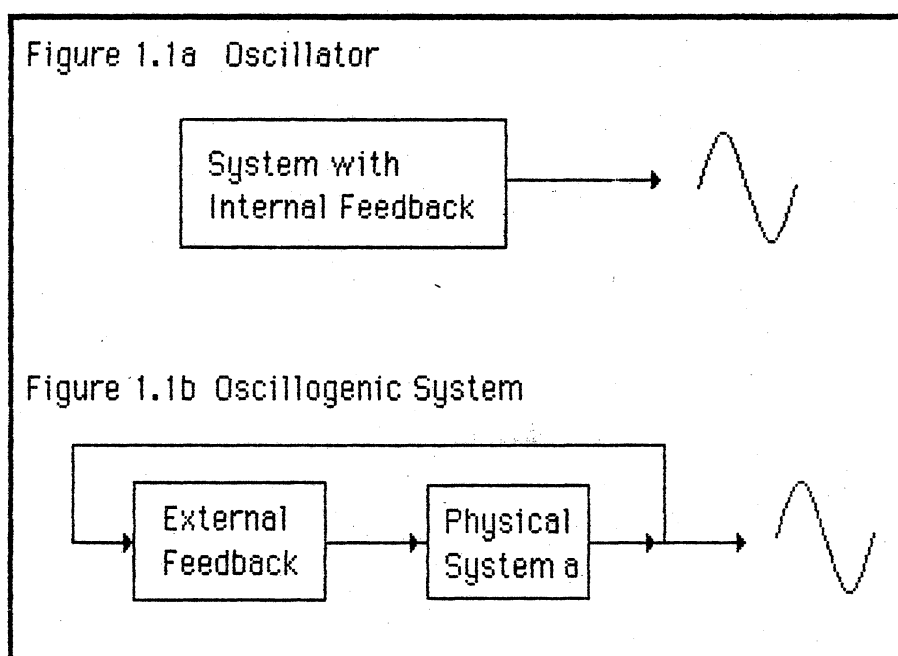
Joseph Weizenbaum,
<u>Computer Power and Human Reason,</u>
1976

The intention of this work is to introduce a mechanism for a generalized chemical oscillator and to demonstrate how it can be applied to the measurement of chemical quantities. The term "generalized" is used here to mean: a technique which can be applied to many different systems without dependence on special kinetics or transport phenomena. The term "chemical oscillator" indicates that an integral part of the oscillator is some chemical action. This usually means that a chemical reaction must take place and that the rate of reaction must change in a periodic fashion. Of course the reaction of two chemical species is included in this definition, but the reaction of electrons with chemical species (electrochemistry) and the reaction of photons with chemical species (photochemistry) are also included. Other interactions might also be considered if they result in chemical changes. Perhaps there might ultimately be examples of "magnetochemistry" (magnetic fields causing chemical change), "electrostatichemistry" (static electric fields causing chemical change), "gravichemistry" (gravitational fields causing chemical change), and even "nucleochemistry" (nuclear interactions causing chemical change).
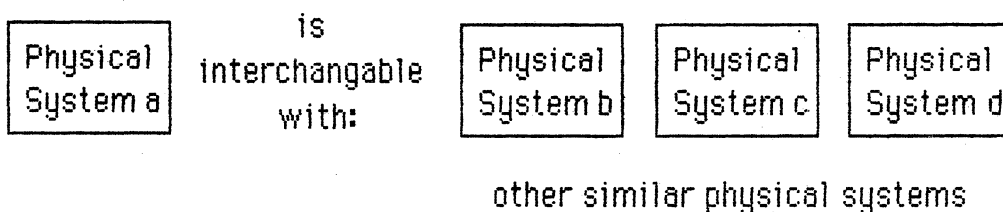
The name "oscillogenic", from the Latin for 'giving birth to swinging', is suggested for those techniques that yield generalized oscillators. An *oscillogenic* device is an oscillator that is comprised of two sections. First, a physical system with a clearly defined input and output. And second, a feedback system that can be applied to many different physical systems. Neither part oscillates by itself, but the combination of the two gives rise to oscillations. In the case of chemical oscillogenic techniques, the physical systems are those that include dynamic chemical interactions. In combination with an oscillogenic feedback system, the physical systems can generate periodic signals that can be used to measure quantities of chemical interest: concentrations, transport properties, reaction rate expressions, and equilibrium coefficients. In a sentence, *chemical oscillogenic techniques* use chemical reactions, transport properties, or equilibria to generate oscillations for the purpose of extracting chemical information from chemical systems.

A schematic view of the difference between an oscillator and an oscillogenic system can be seen in Figure 1.1. An oscillator is a self-contained system. Some quantities associated with the oscillator fluctuate and can be observed from outside the system (Figure 1.1a). An oscillogenic system is a special type of oscillator (Figure 1.1b). A physical or chemical system of a certain class is connected to an external circuit, which provides feedback such that the combined system oscillates. If the system boundary is drawn to include the external feedback device, then the system will appear to be self-contained. The

frequency and the shape of the waveform produced by the oscillogenic system will be related to the properties of the physical or chemical system. Since the oscillogenic instrument is designed to have an interchangable physical or chemical system (as long as it is in the class of systems for which the oscillogenic instrument is designed), information can be extracted from many different chemical or physical systems.

Figure 1.1a Oscillator

System with Internal Feedback

Figure 1.1b Oscillogenic System

External Feedback

Physical System a

Note:

Physical System a   is interchangable with: Physical System b   Physical System c   Physical System d

other similar physical systems

Chemical oscillators have been the subject of keen interest, especially in the last twenty years. Oscillations have been investigated

in: catalytic oxidation (heterogeneous phase kinetics), exothermic
reactions in stirred tank reactors (thermokinetic oscillators),
electrochemical reactions, and homogeneous reactions in liquids. Several
attributes can be used to classify these various oscillating chemical
systems. First, is the system open or closed to the passage of matter and
energy? And second, is the system homogeneous or heterogeneous; that
is, how many phases of matter are required in the chemical system to
make the oscillator? A summary of several systems is presented in the
table below:

	boundary	phases
catalytic oxidation	open	two
thermokinetic	open	two
electrochemical	open	two
biochemical	closed	two
Belousov-Zhabotinsky	closed	one
Bray	closed	one
Brandeis group	open	one

Chemists have lavished the greatest interest on the homogeneous
closed chemical systems. The possible existence of these systems was
the basis of A. J. Lotka's classic papers [Lotka (1910, 1920)], which were
closely followed by the discovery of an actual oscillating homogeneous
chemical system by W. C. Bray [Bray (1921)]. The modern study of
chemical oscillators began with the initially ignored work of B. P.
Belousov in 1951 [Belousov (1959), Winfree (1984)]. This work was given

exposure to the international chemical community and credited to Belousov by A. M. Zhabotinskii [Zhabotinskii (1964a, 1964b)]. Zhabotinskii's work popularized the reaction outside of the USSR and started much analysis of the system, including the work of H. Degn [Degn (1967, 1968, 1972)], and the definitive study of the kinetics by R J Field and R M Noyes [Noyes (1972), Field (1972)]. This reaction also figured promenently in Prigogine's work [Prigogine (1970), Nicolis (1977)] on self-organizing systems. So much work exists on the Belousov system that it is difficult to give an exhaustive listing of all of the literature. Some of the more prominent articles are listed in the bibliography [Bar-Eli (1984), Belousov (1959), Zhabotinskii (1964a,b), Degn (1967), Bruice and Kasperek (1971), Winfree (1972), Noyes et al (1972), Degn (1972), Field (1972a,b), Demas (1973), Winfree (1973), Faraday Society (1974), Winfree (1974), Field (1977), Walker (1980), Botre (1981), Edelson (1981), Mittal (1981), Mankin (1982), Madore and Freedman (1983), Bar-Eli (1984), Winfree (1984), Rovinsky and Zhabotinski (1984), Rouff and Schwitters (1984), Field and Burger (1985)]. All in all, there has been a substantial amount of analysis of this system. The reader seeking a comprehensive review of homogeneous chemical oscillators should consult the book by Field and Burger [1985], as well as the report by Burger et al [1983], for which extensive resources were available for reference accumulation, including a substantial contribution from an analysis of magnetic tape data from the Science Citation Index [Burger and Bujdoso (1985)].

Since the above works dealt mainly (although not exclusively) with homogeneous systems, the Field and Burger [1985] text slights two major

categories of investigators. The electrochemist's oscillators are invariably heterogeneous because of the need for electrodes [Krüger (1906), Tanamushi (1966), Levie(1970), Wojtowicz (review 1972), Poncet (1977), Kelzer (1980), and Yoshikawa and Matsubara (1984)]. The work of chemical engineers is mostly on open systems and occasionally with external feedback [Aris and Amundson (1958), Douglas (1967), Bush (1969), Baccaro (1970), Denbigh (1971), Douglas (1972), Bailey (1973), Oktem (1974), Liaw (1974), Bailey(1977), Padmanabhan and Lapidus (1977), Ray (1980), Morton (1981), Mankin(1981), Matsubara (1982), Thompson (1982), Chang and Chen (1984), Lyberatos (1984)].

The Belousov-Zhabotinskii reaction and the other oscillating systems mentioned have been studied as ends in themselves. For the most part, there has been "a surfeit of simulations" [Bailey(1977)]. Hans Degn stated the next step clearly [Degn(1972)]: "One important task which remains to be done is the 'synthesis' of a new oscillating reaction from reaction elements chosen purposely to create the right feed-back properties for oscillations to occur."

This challenge stood until 1981, when a group at Brandeis University, I. R. Epstein, K. Kustin, P. de Kepper, and M. Orban, synthesized an oscillating chemical system [De Kepper (1981a,b)] . These workers gathered together the known behavior of a number of chemical reactions and relaxed the constraint of using a closed system. They studied reactions in the open system of the continuous stirred tank reactor well known to chemical engineers [Aris and Amundson (1958), Douglas (1972), Russell and Denn (1972), Froment and Bischoff (1979), Levenspiel (1979),

Ray and Jensen (1980), Smith (1981)]. From their work, they formulated some rules of thumb [Epstein et al (1983)] for synthesizing chemical oscillators: 1) The system must be far from equilibrium, 2) There must be some feedback such that some product indirectly affects the rate of its own formation, and 3) The system must exhibit bistability, that is it must have conditions where two different sets of steady state concentrations can exist. The design process used was described more systematically [Epstein (1984b)]:

1. Choose an autocatalytic reaction R
2. Run the reaction, R, in a CSTR (Continuous Stirred Tank Reactor) and seek conditions under which the system is bistable.
3. Choose a feedback species Z which perturbs the system by different amounts on the two branches of steady states.
4. By increasing the input of Z into the CSTR, seek the critical point at which bistability disappears and oscillations begin.

In general, the chemists who have investigated oscillators have tended to be purists; systems of interest have been narrowly defined as homogeneous and closed. There has been some work on systems which violate these constraints, but until recently with much less interest. There are three main areas in which the constraints of homogeneous and closed systems are relaxed. The first area is electrochemical oscillators, which are always heterogeneous [Kelzer and Scherson 1980, Krüger 1906, Levie 1970, Poncet 1977, Tanamushi 1966, Wojtowicz 1972, Yoshikawa and Matsubara 1984]. The second area is the Brandeis group's general use

of an open system (Continuous Stirred Tank Reactor) to study chemical oscillators [Alamgir et al 1983, Boissonade and de Kepper 1980, Dateo 1982, de Kepper and Boissande 1985, de Kepper et al 1981a,b, Epstein 1983a,b 1984a,b, Epstein and Orban 1985, Maselko and Epstein 1984, Orban 1982, Orban et al 1982a,b, Orban and Epstein 1982]. As a third area, there was the recent use of heterogeneous electrodes in studying the Belousov-Zhabotinskii reaction by Crowley and Field [1986].

Opposed to purists are engineers, who tend to be more pragmatic: they seek to create systems which can be used for a purpose. Relaxing the constraint of a closed and homogeneous system is justified if a method of making any chemical system oscillate can be found. Also, once the system oscillates, for what practical purposes can the oscillation be used? One notable individual in the "application" of chemical oscillating systems has been J. E. Bailey, who has studied the possiblity of increasing reaction yield by using an oscillating system [1973, 1977] and the discrimination between reaction mechanisms based on the onset of controller-induced oscillation [Liaw and Bailey 1974, Lyberatos et al 1984, 1985a,b].

The work of the chemists to this point have been, primarily with "found" oscillators. Epstein [1984b] notes: "...one might summarize the sources of oscillatory reaction prior to 1980 as: 1. Accident, 2. Biology, 3. Variation on a theme". Serendipitous discovery of oscillating systems comprised most of the demonstrations. Only recently has the Brandeis group systematically devised chemical oscillators, but even then with what might be termed "found" kinetics: these 'designed' oscillators are

all based on chemistry of oxy-halogen containing systems. A recent reference [Nagy and Treindl 1986] showed a non-halogen oscillatory system, the design of which essentially followed the Brandeis method with a system previously known to exhibit bistability. But the discovery of the bistability of the system [de Kepper 1984, Reckley and Showalter 1981, Epstein 1981] may still be considered "finding the kinetics" rather than designing the system.

The only presently known homogeneous chemical oscillator containing neither "metal nor oxyhalogen ions" is that of Burger and Field [1984]. This system was not designed by the methods of the Brandeis group: "we have not yet observed bistability in this system" [Burger and Field (1984)]. One could characterize the discovery of this system as serendipitous. In summary, the choice of chemical systems for the synthesis of chemical oscillators has been very restricted.

The starting point for this investigation has been well stated by Epstein: "While many chemical engineers were familiar with reactor oscillations (and in fact devoted considerable effort to eliminating them), the majority of chemists were convinced that sustained periodic behavior in relatively simple chemical systems must be artifactual at best, fraudulent at worst, a thermodynamic impossibility" [Epstein 1984b]. As will be presented (chapter 3), a central issue is why engineers (chemical and electrical) found oscillators a natural occurrance and yet chemists found them difficult to accept.
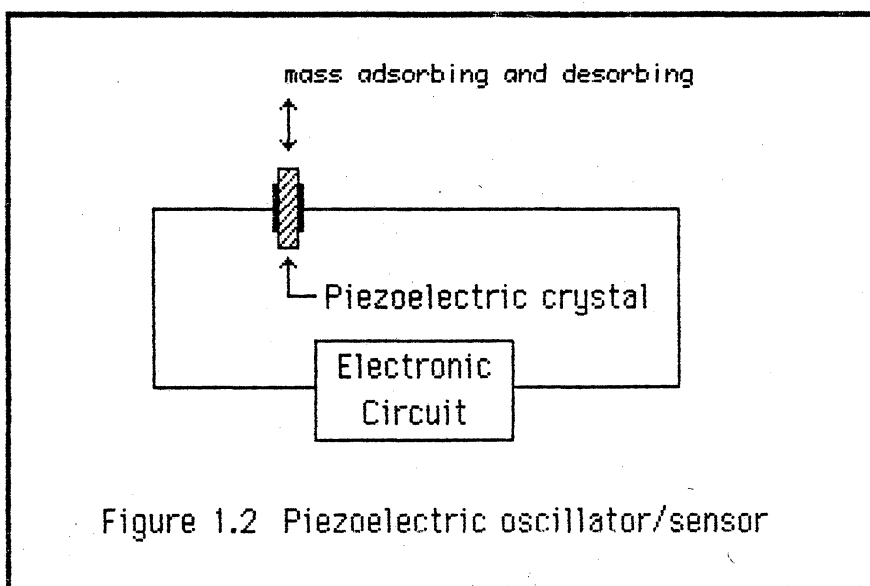
This work proposes a method for producing chemical oscillators from a wide variety of chemical systems in order to create a new class of

instruments, called oscillogenic instruments. Before detailing the design of oscillogenic instruments, it is prudent to examine existing methods which might already be oscillogenic. The work of Bailey, mentioned above [Liaw and Bailey 1974, Lyberatos et al 1984, 1985a,b], might be considered oscillogenic because of the use of external feedback (proportional controller) with a chemical system (reaction in a CSTR), but the actual oscillation was not used directly: <u>the controller gain at the onset of oscillation was the only quantity used in the measurements</u>. The systems of Bailey are not amenable for use as "chemical sensors".

Examination of the literature on oscillators used with chemical systems shows two classes of devices already in use: piezoelectric oscillators for gas composition measurement and oscillators for measuring the electrical conductivity of liquids.

Piezoelectric crystals (most commonly quartz) are widely used in oscillator circuits. These crystals provide a very precise control of the oscillator frequency. Investigators have shown that when the crystals of oscillators of this type are exposed to a gas sample, the frequency can be a function of gas composition. There have been review papers [Guilbault 1980, Pinkhusovich 1979] describing a wide variety of composition measurements which can be obtained by this method, and the method has been patented [Guajardo 1972]. The technique depends on components of the gas depositing in some manner onto the surface of the crystals. This changes the mass of the crystals and thus the frequency of the oscillator. Piezoelectric sensors differ from the above definition of chemical oscillogenic techniques above in that the oscillator dynamics do not

depend <u>chemically</u> on the measured system. Another way of illustrating the contrast between this oscillator and an oscillogenic system is to consider the input and output of the physical system as illustrated in Figure 1.2.

mass adsorbing and desorbing

Piezoelectric crystal

Electronic
Circuit

Figure 1.2 Piezoelectric oscillator/sensor

The piezoeletric oscillator does not perturb the physical system, and then use the system's response as a part of the oscillator mechanism -- there is no input to or output from the physical system as described above. The coupling between the electronics and the chemical system is <u>mechanical</u> not chemical. One consequence of this distinction is that <u>direct</u> extraction of information on the dynamics (reaction kinetics, transport phenomena) of a chemical system is not possible using these devices. This type of oscillator is an extremely sensitive gravimetric method of analysis.

Since electrical conductivity is an electrical property, it is only

natural to incorporate this property into an electrical oscillator. Indeed, oscillators are used widely to measure electrical properties. A number of patents have been issued for circuits which incorporate oscillators for the measurement of the conductivity of electrolytes [Acker (1973), Brown 1970, Jackson 1959, McBride 1981, Murdock 1975, Stephen 1982, Steuer 1975].

Solution as a variable resistor

Electronic
Circuit

Figure 1.3 Electrical conductivity measurement
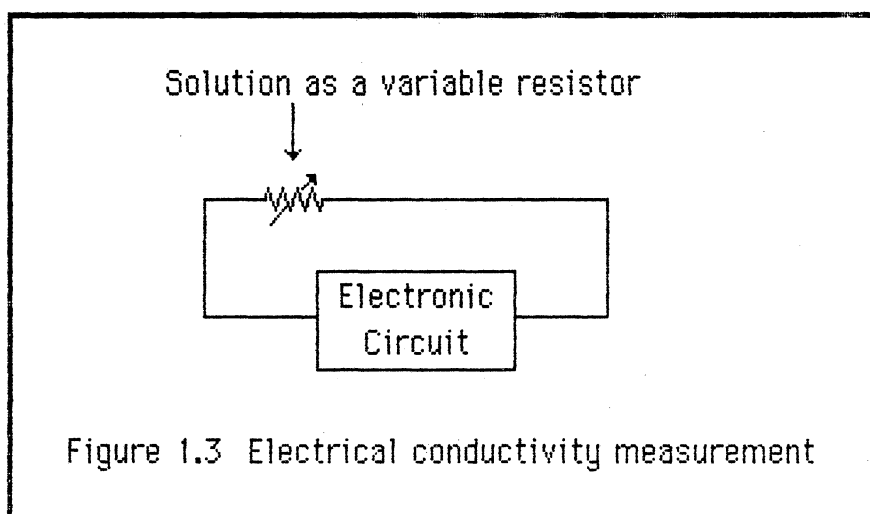
However, conductivity is not a very selective property; it is next to impossible to measure the concentration of a chemical species by conductivity when other species are varying in the mixture. The relationship between conductivity and chemical reactivity is not well characterized. In fact, a common technique in measuring AC conductivity is to insulate the electrodes from contact with the chemical mixture! While this removes problems with electrode contamination, it also removes the interaction with the reactive chemistry of the system. Conductivity is not the only measurement using resistance, oscillators

that measure resistance (1/conductivity) can be used with other types of sensors: platinum resistive temperature devices (RTD), strain gages, and thermistors to name a few. The key issue is that the chemical dynamics of the measured system do not control the dynamics of the oscillator.

In the open literature, the use of oscillating systems with a coupling between the generation of the oscillation and the chemical dynamics (kinetics or transport phenomena) for measuring chemical quantities is a new concept.

A patent disclosure has been filed with Virginia Polytechnic Institute and State University (date 9 March 1985) on the subject of oscillogenic instruments. Research Corporation (6840 East Broadway Boulevard, Tucson, Arizona 85710) has been pursuing the possiblity of a patent on this device. The firm of Scully, Scott, Murphy, and Presser (Attorneys at Law, 200 Garden City Plaza, Garden City, New York 11530) has conducted a patent search on this subject (letter of 16 April 1986). There were a number of patents returned for consideration as "prior art". The vast majority concerned: oscillators, chemical sensors, electronic control of electrochemical systems, and electronic measurement of sensor signals. These patents did not <u>combine</u> the principles of 1) electronic input/output interface to chemical systems, and 2) generating oscillations by electronic feedback and chemical dynamics. There were a number of oscillators used in measuring chemical quantities, and these fell into three categories: 1) oscillators measuring conductivity [Brown 1970, Frigato 1979, Jackson 1959, McBride 1981, Murdock 1975, Stephen 1982, Steuer 1975], 2) oscillators measuring capacitance [Bauerlen 1983, Ellis

1978, Grindheim 1983, Pile 1980], and 3) piezoelectric oscillators [Guajardo 1972]. The oscillators for measuring capacitance could determine some chemical parameters, such as concentration, but they did not interact chemically with the system and, similar to conductance measurements, they were not selective measurements. Selectivity is the ability to discriminate the quantity of one component in the presence of varying quantities of other components. Selectivity is based on a differential sensitivity of the measurement to different chemical species. In general, any sensor which creates an electronic signal can be used as part of an electronic oscillator (generically known as *voltage to frequency* converters), but this does not use the chemical system as an integral part of the oscillator. Essentially, these electronic systems will oscillate whether any chemical action takes place or not. These systems are not oscillogenic.

Two inventions were found which could be considered oscillogenic. First, two patents using zirconia oxygen sensors in oscillogenic circuits: Hetrick [1981b] and Hetrick and Fate [1983] (related non-oscillatory patents Hetrick [1981a] and Hetrick and Fate [1981]). These devices used two zirconia solid state oxygen concentration cells, one device pumped oxygen into one end of a gas filled cavity while the other detected the amount of oxygen at the other end of the cavity. The frequency was a measure of oxygen concentration in one case Hetrick [1981b], and oxygen diffusivity in the other case Hetrick and Fate [1983]. Because of the interaction with oxygen chemically, these devices have potential for other uses. However, these inventors did not see the generality of the principle

of oscillogenic instruments, and did not seem to have pursued the
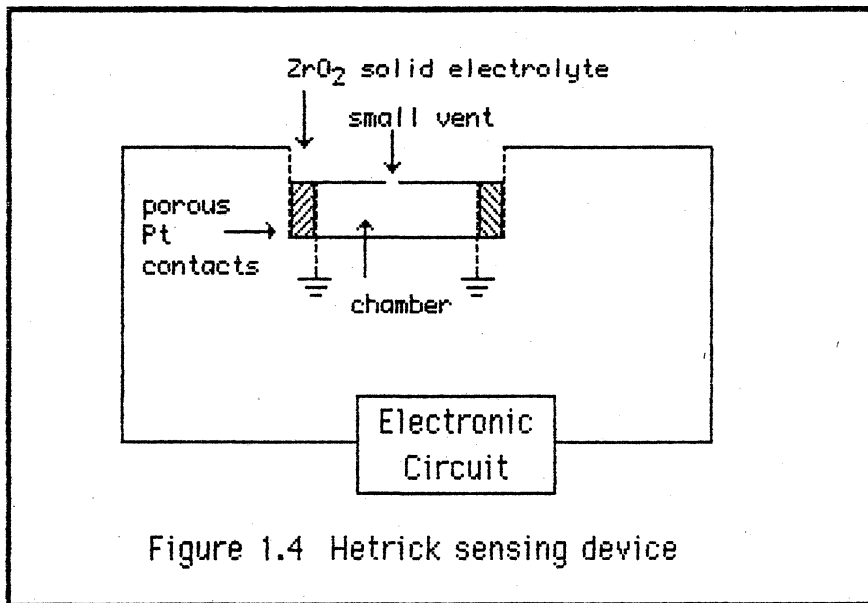principle to other uses.

Figure 1.4 Hetrick sensing device

The second invention was more general. Allan B. Fraser [1983] was
granted a patent titled: "Frequency Encoding Closed Loop Circuit with
Transducer". The intent of the original work was to increase the
immunity of a measurement to sensor and power supply noise. The patent
did describe a general method of incorporating a system into an oscillator,
such that the frequency is related to a physical quantity. The claims of
the patent included three specific physical systems: 1) optical output and
input devices (including light emitting diodes and photodetectors), 2)
thermistors, and 3) strain gage devices. The nature of the signals were
all physical and not chemical, and thus are examples of physical
oscillogenic systems. Figure 1.5, redrawn from U. S. Patent 4,408,169,
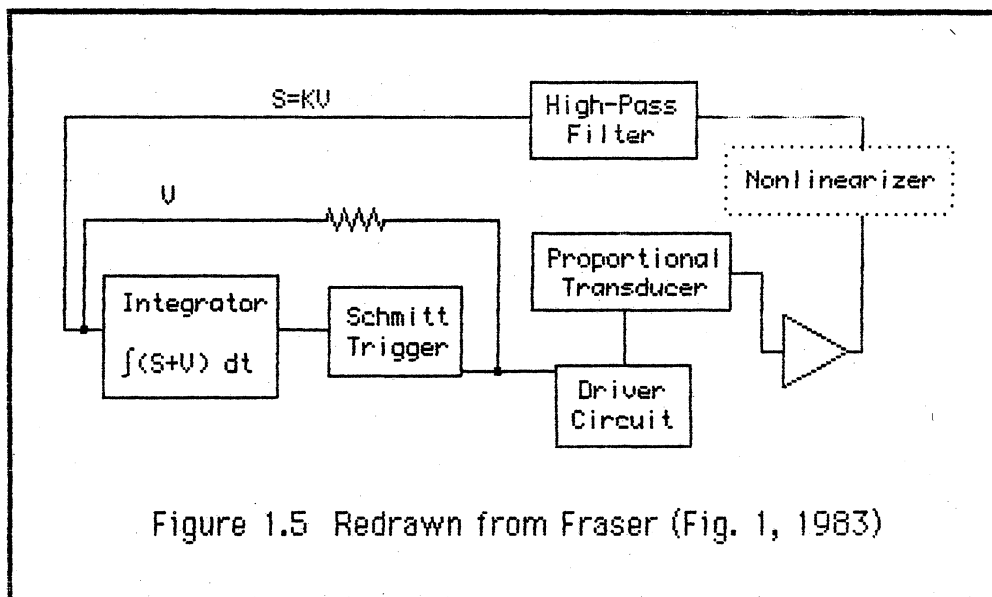shows Fraser's circuit which may be considered oscillogenic.

Figure 1.5 Redrawn from Fraser (Fig. 1, 1983)

Specific mention was made of 'proportional' transducers, indicating linear interactions. No dynamics of the chemical system were considered. This is different from the present work, which considers non-linear effects (electrochemical reaction rates) and a dynamic system (electrochemical reaction/diffusion problems). Fraser considered only the case of an integrator coupled with a Schmitt trigger. As will be seen in chapters 3 and 5, the present work considers oscillogenic feedback circuits which do not require integrators, and proposes feedback nonlinearities that do not involve Schmitt triggers. Fraser seemed to consider the invention as a better way to excite and read sensors. No data on the performance of the oscillators was presented in the patent. A search was conducted for other publications related to Fraser's work. The sources consulted were: Science Citation Indices, Chemical Abstracts, Engineering Index, Physics Abstracts Government Reports, Physics Abstracts, Computer and Control

Abstracts, and the IEEE Index. Two references to Fraser were found, but neither was concerned with oscillators.

Both of these oscillogenic systems were uncovered <u>after</u> the theoretical and experimental part of this work had been completed. The present work must be considered an independent discovery of the oscillogenic instrument, with original contributions in the analysis of the general concept and in the specific implementation of an oscillogenic electrochemical instrument which will be considered in subsequent chapters.

What possible advantages might one expect to accrue from oscillogenic techniques (that is, a general chemical oscillator)? Time is one of the easiest quantities to measure, and is acknowledged to be the most precise measurement possible today. Thus, the measurement of a frequency can be expected to be very precise. Periodic signals lend themselves to many techniques: signal averaging, waveform sampling, Fourier transforms, perturbation analysis, high temporal resolution digitization... Finally, one should not limit oneself to one's own imagination; the invention of a chemical oscillator, which can be applied to many different systems, may open chemistry to different perspectives and thus new insight.

The remainder of this dissertation will be concerned with the development of oscillogenic techniques. The most familiar oscillators are physical oscillators (not involved with chemical change). The design of physical oscillators, especially electronic, is well known (Chapter 2). Why did chemists have difficulty accepting chemical oscillators and by

what process did two chemical engineers (Mercure and Rony) devise a method for a general chemical oscillator? (Chapter 3). Given the idea for a generalized oscillator using chemical systems, can one demonstrate the feasiblity of its construction (Chapters 4 and 5)? Can one demonstrate a reduction of the theory to practice by the construction of such a device for a class of reactions (Chapter 6)? Since the device is designed to extract information from a chemical system, can one apply techniques for extracting information that are not known by chemists (or chemical engineers) in general? Chapter 7 is a tutorial on an extremely powerful class of methods for extracting information from dynamic systems. Finally, what other potential reductions to practice and directions for future work exist (Chapter 8)?

Chapter 2. Physical Oscillators

*"It was the painted figure of Time as he is commonly
represented, save that, in lieu of a scythe, he held what, at a
casual glance, I supposed to be the pictured image of a huge
pendulum, such as we see on antique clocks... While I gazed
directly upward at it I fancied that I saw it in motion"*
                                        Edgar Allan Poe,
                                        The Pit and the Pendulum"

A *physical oscillator* is an oscillating system where the time
varying quantities are the result of non-chemical interactions. Familiar
physical oscillators include pendulums, weights suspended on springs, and
the electronic oscillators common in radios. The dynamics are determined
by large body interactions or by electronic interactions, rather than by
intermolecular interactions.

The situation with the much older field of physical oscillators is
quite a contrast with the field of chemical oscillators. As early as the
sixteenth century, it is said that Galileo timed, by his pulse, the motions
of a swinging lamp in the cathedral of Pisa and established the periodicity
of the motion. *Periodicity* is defined for a function, f(t), such that if
the function has period, L, then:

$$f(t+L) = f(t) \text{ for all } t \qquad\qquad [2.1]$$

Some of the simplest periodic functions are the trigonometric
functions, for example:

$$f(t) = \sin(\omega t) \qquad\qquad [2.2]$$

where

$$L = 2\pi/\omega \qquad\qquad [2.3]$$

Expressions containing various sine and cosine terms are known as harmonics; therefore, oscillators which generate these waveforms are called *harmonic oscillators*. One such oscillator is the simplest ideal electrical oscillator; an *LC oscillator* comprised of an inductor and a capacitor connected together. Figure 2.1 shows the circuit for this oscillator.

Figure 2.1 Inductor-Capacitor 'Oscillator'
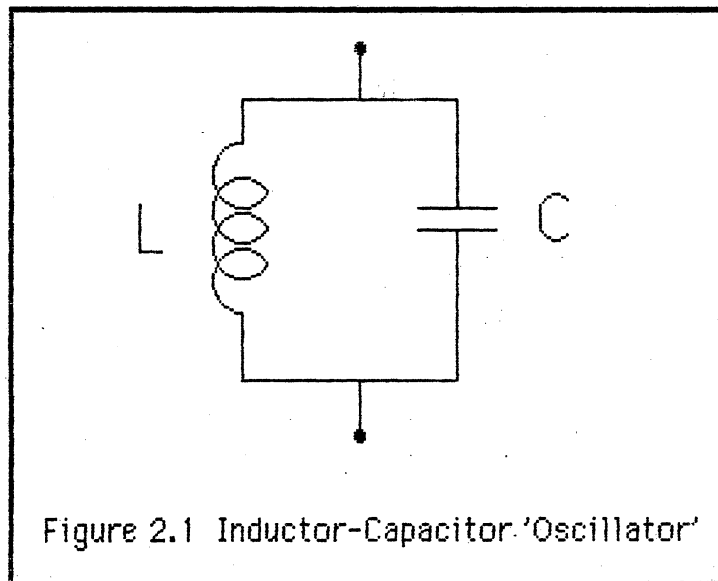
This combination can be described by the differential equation,

$$\ddot{x} + \omega^2 x = 0 \qquad\qquad [2.4]$$

where,

     x is the electrical potential (voltage) across the terminals

     $\omega^2 = 1/LC$, the square of the oscillator frequency
            in units of (radians per unit time) squared   [2.5]

and,

$$\ddot{x} = \frac{d^2x}{dt^2} \qquad\qquad [2.6]$$

This equation also approximates the behavior of a mass attached to a spring and that of a pendulum with a small displacement from vertical. The quantity, x, represents different physical quantities in each case: angular displacement for the pendulum, translational displacement for the mass on a spring, and the potential across the capacitor for the capacitor/inductor pair. Actually, the full solution for this equation can be expressed as,

$$x(t) = a \cos(\omega t) + b \sin(\omega t) \qquad\qquad [2.7]$$

or as,

$$x(t) = r_1 \exp(-j\omega t) + r_2 \exp(+j\omega t) \qquad\qquad [2.8]$$

Equation 2.7 shows why this oscillator is called a harmonic oscillator; the function is composed of trigonometric sine and cosine terms.

The differential equation 2.4 can be expressed as a *block diagram* as shown in Figure 2.2. These diagrams are useful in visually expressing the inter-relationships in a dynamic system. Some common symbols used in a block diagram are defined in Figure 2.3. Each block has one or more
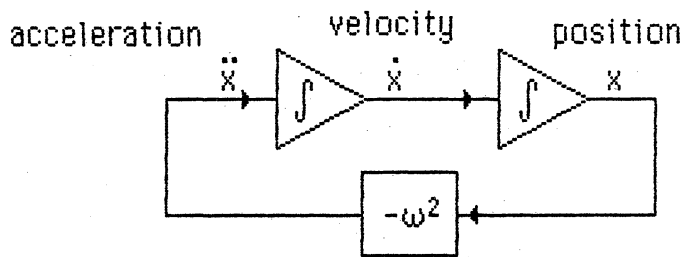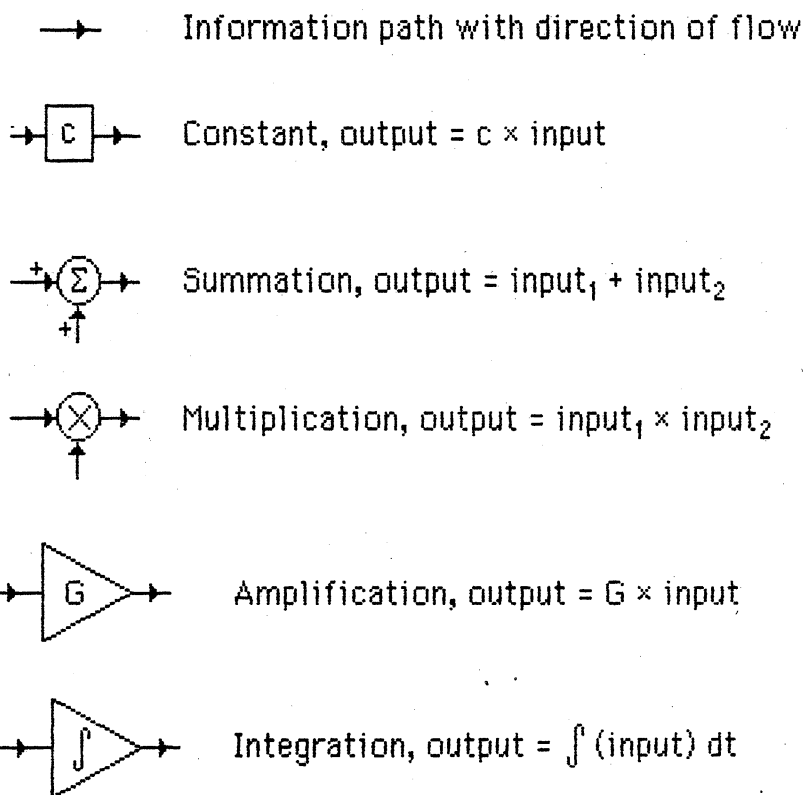
Figure 2.2 Harmonic oscillator

Figure 2.3 Common symbols used in block diagrams

inputs and <u>one and only one</u> output; the direction of information flow is denoted by the arrow heads. Note the long connection from the output of the second integration block in figure 2.2 to the input to the first integration block. This connection feeds information from an output of the system back to an input and is called *feedback*. A feedback element is <u>required</u> for the operation of any oscillator, and will be found somewhere in the differential equations or the block diagrams describing the system! Figure 2.4 shows a generic illustration of *feedback*.

Feedback

Figure 2.4 Feedback path linking output
to input of a system
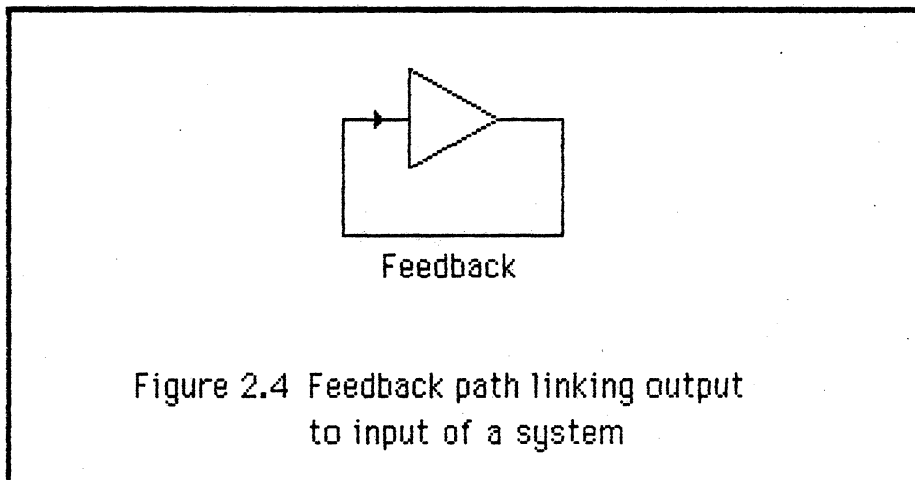
Because the differential equation 2.4 is second order, there are two undetermined constants, which depend on the initial conditions. In this case, the constant pair a and b or the pair r_1 and r_2 are these undetermined constants. The values of these constants can be determined from the values of $x(t)$ and d $x(t)$/dt at a particular time. Such values define the *state of the system* at a particular point in time; the state at any other time can be predicted from this information about the system

at a single time plus the differential equation. The *state* of a dynamic system can be likened to the *thermodynamic state* of a system in chemical equilibrium. The mathematical description of the relationship between composition, temperature, pressure, and volume, is known as an *equation of state* [Castellan 1964, Prausnitz 1969]. For a chemical system at equilibrium, the *equation of state* and the values of the *variables* in that equation completely determine the condition of the chemical system. It is quite natural to refer to the variables in such an equation as *state variables*.

The idea of state variables can be generalized for any system of ordinary differential equations: (a) there is a set of variables that fully describe the state of the system, and (b) if the values of these variables are known at any particular time and the input to the system is known, then the state of the system can be predicted at all times. A more formal definition of the term *state variable* is:

> The state of a system is a mathematical structure containing a set of n variables $x_1(t)$, $x_2(t)$,.... $x_i(t)$,...., $x_n(t)$, called the *state variables*, such that the initial values $x_i(t_0)$ of this set and the future system inputs $u_j(t)$ are sufficient to uniquely describe the systems's future response for $t \geq t_0$.

> [Kalman 1963, D'Azzo & Houpis 1975]

Since the values of x_1 and x_2 at a point in time define the coefficients in equations 2.7 and 2.8 and allow prediction of the state at any other time, these variables are state variables for this system. For the physical

systems described, the state variables may have physical meanings. For example, for the pendulum or the mass/spring, x_1 corresponds to position and x_2 to velocity, while for the capacitor/inductor, x_1 corresponds to the voltage across the capacitor and x_2 to the current flowing in the system. For the electrical example of figure 2.1, two exactly equivalent descriptions are possible:

$$\ddot{i} + CL\, i = 0 \qquad\qquad\qquad [2.9]$$

or

$$\ddot{e} + CL\, e = 0 \qquad\qquad\qquad [2.10]$$

note that
$$i = C\, \dot{e} \qquad\qquad\qquad [2.11]$$
and
$$e = L\, \dot{i} \qquad\qquad\qquad [2.12]$$
where

      i is the current across an element

      e is the potential across an element

      C is the capacitance

and   L is the inductance

Just as equations 2.9 and 2.10 both fully describe the electrical system of figure 2.1, in general there are many different state variable representations of a system. That is, the choice of the state variables is not unique! This should not cloud the issue if one compares the state variables of a dynamic system with the more familiar state variables in a

thermodynamic equation of state. The simplest thermodynamic equation
of state is the ideal gas law:

$$p V = n R T \qquad [2.13]$$

where
      p is the pressure
      V is the volume
      n is the number of moles of gas
      R is the universal gas constant
and   T is the absolute temperature

This equation can be rewritten in this form:

$$p = \rho R T \qquad [2.14]$$

where
      ρ is the molar density (moles per unit volume)

In one case, the state variables are pressure, volume, and temperature (p
V T), while in the second case the state variables are pressure, density,
and temperature (p ρ T). Even in thermodynamics the choice of state
variables is not unique. However, there are specific relationships between
different sets of state variables both in system dynamics and in
thermodynamics. The choice of state variables is set by considerations
such as: communicating a physical "feeling" for the system, ease of
computation, simplicity of representation, etc. The state variables used
in this work are usually chosen for their close connection to the physical
system under study.

      The state of the system at any time can also be visualized on a

graph in what is known as the *phase plane*; the first derivative of a variable (with respect to time) is graphed as a function of that variable. This is a convenient method of displaying the behavior of a system without the explict use of a time variable. Dynamic systems with only two state variables are particularly well illustrated by this technique, because any possible combination of state variables can be depicted. The mechanical systems previously discussed (the spring and mass combination and the pendulum) have graphs in the phase plane which are graphs of velocity versus position. Velocity and position are two easily understood state variables. Figure 2.5, on the next page, shows such a phase plane graph for the system described by equation 2.4. The periodic nature of the oscillator can be seen in the phase plane as a closed curve. This curve displays a *trajectory* of the system, that is, the path which describes how the variables change as time progresses. Arrow heads are customarily used to denote the direction of travel of a system along a trajectory. Closed curves on a phase plane graph are periodic solutions to the system equations. For a harmonic oscillator, the trajectories are ellipses; the size of the ellipse is determined by the initial conditions of the state variables.

The differential equation for the harmonic oscillator can be written as a system of first order differential equations:

$$\dot{x}_1 = x_2 \qquad\qquad\qquad [2.15]$$

and

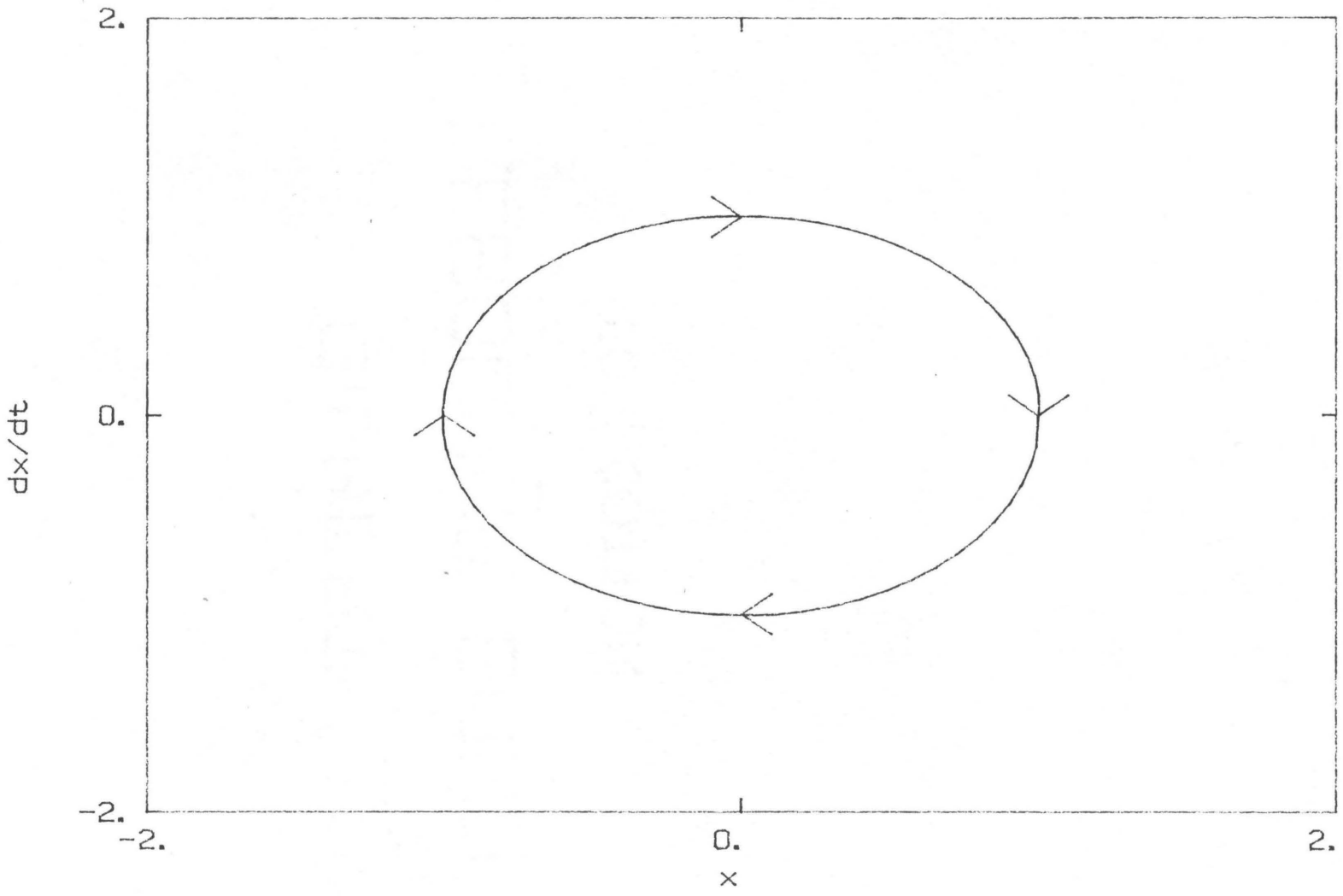$$\dot{x}_2 = -\omega^2\, x_1 \qquad\qquad\qquad [2.16]$$

Figure 2.5   Phase Plane Graph of the Harmonic Oscillator

where

$$\dot{x} = \frac{dx}{dt}$$, the dot ($\dot{}$) means the first derivative with respect to time

$$x_1 = x$$

and

$$x_2 = \dot{x}$$

Note that these differential equations may be put in a more standard form,

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} \qquad [2.17]$$

where the **bold** typeface indicates a matrix or vector quantity.

In this case (compare to equations 2.15 and 2.16),

$$\mathbf{A} = \begin{bmatrix} 0 & 1 \\ -\omega^2 & 0 \end{bmatrix} \qquad [2.18]$$

Since we have passed over into the matrix notation of linear algebra, let us take a short diversion into one of the principles of that subject. One of the fundamental sets of quantities related to a matrix, such as **A** above, is the set of associated *eigen values* and *eigen vectors*. The adjective *eigen* comes from the German word that means: inherent, intrinsic, belonging, specific, particular, or characteristic. These, then, are quantities specific to a given matrix. The defining relationship is:

$$\mathbf{A}\,\Lambda_i = \lambda_i \Lambda_i \qquad [2.19]$$

where Λ is a vector

λ is a scalar

and, i may range from 1 to the dimension of the matrix **A**

Consider a specific numeric example:

$$A = \begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix}$$ [2.20]

Now find the specific values, the eigen values, for the matrix in equation 2.20. If equation 2.19 holds, then the following must be true:

$$(A - \lambda_i I)\Lambda_i = 0$$ [2.21]

where **I** is the *identity matrix*,

$$I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$ [2.22]

If the eigen vector, Λ, is to be nonzero then:

determinant $(A - \lambda_i I) = 0$ [2.23]

Expanding the determinant:

$$\det (A - \lambda_i I) = \det \left[\begin{bmatrix} -1 & -1 \\ 1 & -1 \end{bmatrix} - \begin{bmatrix} \lambda_i & 0 \\ 0 & \lambda_i \end{bmatrix} \right]$$ [2.24]

$$= \det \left[\begin{bmatrix} -1-\lambda_i & -1 \\ 1 & -1-\lambda_i \end{bmatrix} \right]$$ [2.25]

$$= (-1-\lambda_i)(-1-\lambda_i) + 1$$ [2.26]

$$0 = \lambda_i^2 - 2\lambda_i + 2$$ [2.27]

The solutions to this quadratic equation, and the eigen values of the matrix (equation 2.20), are:

$$\lambda = 1 \pm j \quad \text{where } j = (-1)^{0.5} \qquad [2.28]$$

Note that in general the eigen values are complex numbers, but that either the real or the complex parts might be zero. There is a specific relationship between the eigenvalues and the nature of the dynamic system.

Since the term *linear* is an important part of this chapter, let us review the concept of linearity. *Homogeneity* and *additivity* are two salient characteristics of linear equations [Kirk 1970]:

f is a *linear* function of **q** if and only if it satisfies the *principle of homogeneity*

$$f(\alpha q) = \alpha \, f(q) \qquad [2.29]$$

... and the *principle of additivity*

$$f(q_1 + q_2) = f(q_1) + f(q_2) \qquad [2.30]$$

A linear *system* is a a dynamic system that can be described by a coupled set of linear differential equations. Any linear system described by ordinary differential equations can be expressed in the form of a set of first order differential equations and will be described by a matrix equation of the form,

$$\dot{x} = Ax + b \qquad [2.31]$$

where **b** represents a vector of inputs to the system.

The eigenvalues of the matrix, **A**, characterize the nature of the dynamic system.

Returning to the matrix of equation 2.18, note that the eigenvalues of this matrix are $+j\omega$ and $-j\omega$; the real component is zero. The purely imaginary eigenvalues correspond to a solution to this system of equations that is composed strictly of sine and cosine terms.

As mentioned before, the second-order differential equation describing the motion of a pendulum or the displacement of a mass on a spring is only an approximation to the physical situation. The primary deviation from this approximation is the presence of friction. Any real system has some frictional component. Friction is a term in the differential equation, proportional to the velocity of the system, that acts in opposition to the acceleration term in the system. Figure 2.6 shows the simple L-C oscillator with friction (a resistor, R) added.
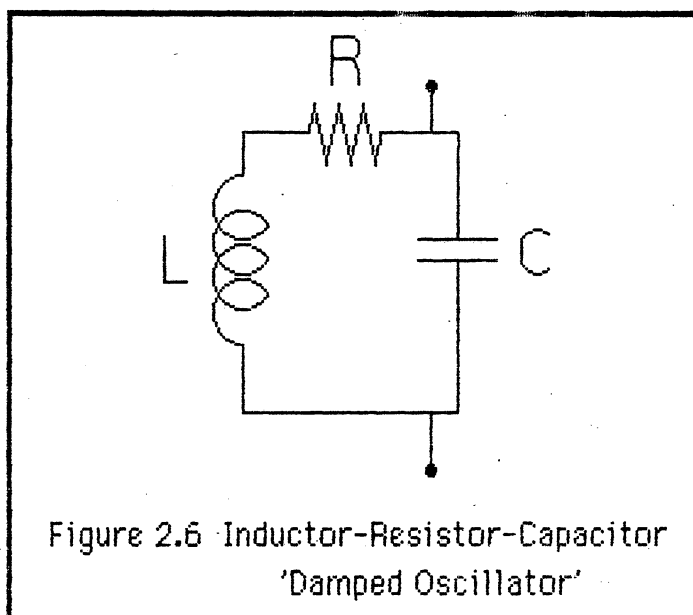
Figure 2.6 Inductor-Resistor-Capacitor 'Damped Oscillator'

In this example, the resulting differential equation is:

$$\ddot{x} + f\dot{x} + \omega^2 x = 0 \qquad [2.32]$$

where

$$f = R/L \qquad [2.33]$$

which is greater than zero and represents a coefficient of friction.

In state variables,

$$\dot{x} = Ax \qquad [2.17]$$

and

$$A = \begin{bmatrix} 0 & 1 \\ -\omega^2 & -f \end{bmatrix} \qquad [2.34]$$

The general solution to equation 2.32 can be shown to be:

$$x(t) = \exp(-ft/2) [r_1 \exp(\Omega t) + r_2 \exp(-\Omega t)] \qquad [2.35]$$

where $\Omega^2 = \dfrac{[f^2 - 4\omega^2]^{0.5}}{2}$ $\qquad [2.36]$

Note that the eigenvalues of the matrix, **A**, are: $(-f/2 - \Omega)$ and $(f/2 + \Omega)$.

Examine the individual terms of equation 2.35, and note that the first exponential term guarantees that the solution goes to zero at long times for f>0. <u>A stable oscillator can be defined as a system that produces a constant amplitude and constant period.</u> Thus the R-L-C example of Figure 2.2 is not a stable oscillator, and is not periodic in the strict sense defined above in equation 2.1. Since all real macroscopic

physical systems have friction of some type, <u>there are no macroscopic</u> <u>real systems described by the harmonic oscillator equation</u> (equation 2.4), and therefore <u>harmonic oscillators cannot be physically realized</u>, except in approximation.

Compare the eigenvalues of the matrix in equation 2.18 to those of the matrix in equation 2.34. These two systems differ only in the presence of friction (equation 2.31) in one of the systems. If the friction term goes to zero, then the eigenvalues of the two systems become identical. By varying the value of the friction one can change the qualitative nature of the behavior of the frictional system. For example, if the friction, f, is sufficiently large, then the Ω term in equation 2.35 is purely real (zero imaginary components), and the system will behave as in figure 2.7 below.

Figure 2.7 High dissipation second order system
Eigenvalues: imaginary part is zero
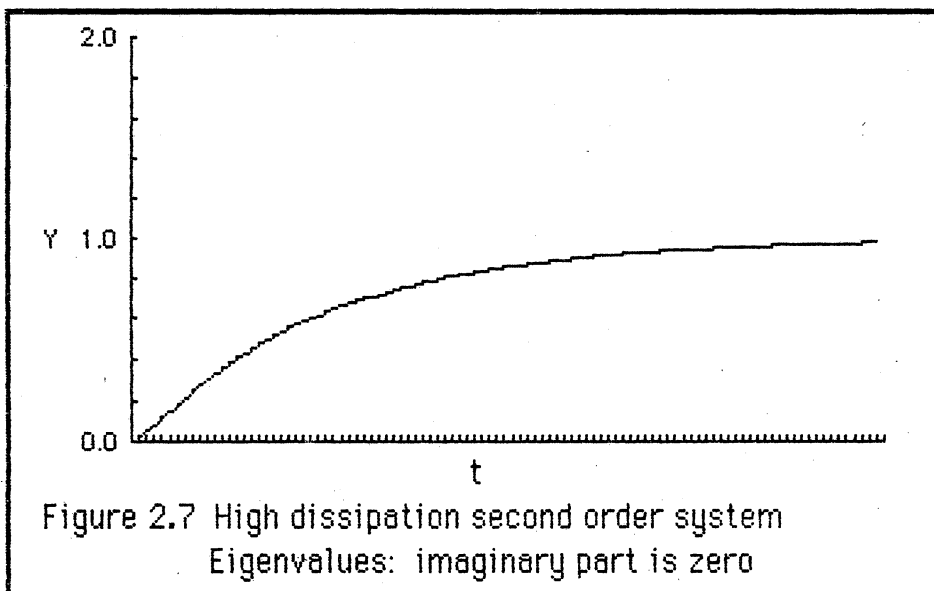
There is no evidence of oscillation in this case. If the value of the

friction drops to the point where the imaginary component of the Ω term of equation 2.36 becomes non-zero, then the response of the system will be similar to Figure 2.8 below; this is the onset of oscillator-like behavior.

Figure 2.8 Medium dissipation second order system
Eigenvalues: real and imaginary parts

Figure 2.9 Zero dissipation second order system
Eigenvalues: real part zero

36
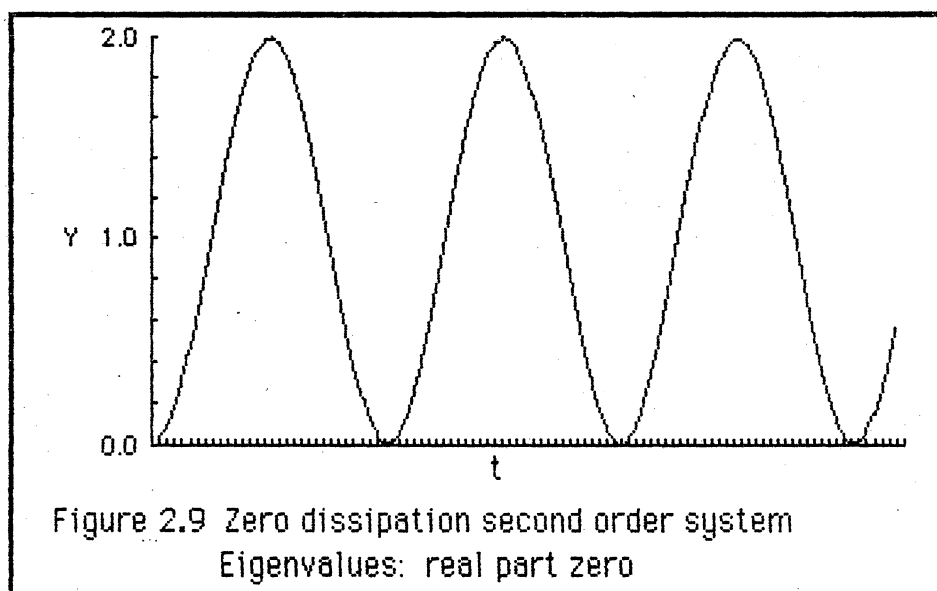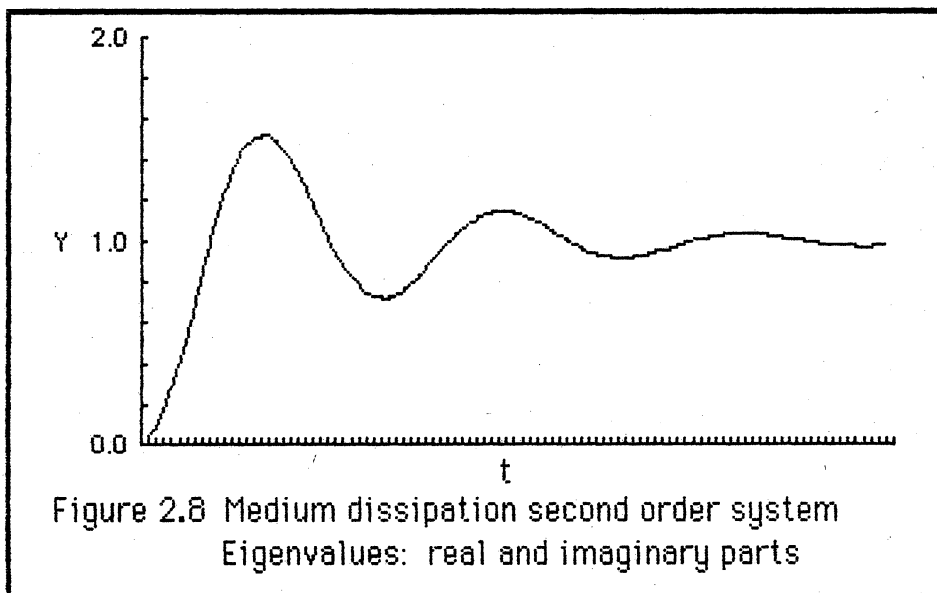
Finally, as has been described above, if the value of friction goes to zero, then the value of Ω in equation 2.33 becomes purely imaginary (real part is zero). At this point, the system response is oscillatory as in Figure 2.9. Thus <u>the nature of the eigenvalues predicts the response of linear systems</u>.

To compensate for the friction and force the eigenvalues to purely imaginary values, suppose we had a way to add a *negative friction* to the system,

$$\ddot{x} + (f - G)\,\dot{x} + \omega^2 x = 0 \tag{2.37}$$

If we could adjust the constant, G, to exactly match the unknown frictional element, f, then the oscillator might be stable. Essentially, the negative friction is an energy source to replace the energy dissipated by friction. And this energy is added by a velocity feedback in the system.

Figure 2.10 Inductor-Resistor-Amplifier-Capacitor
'Oscillator'
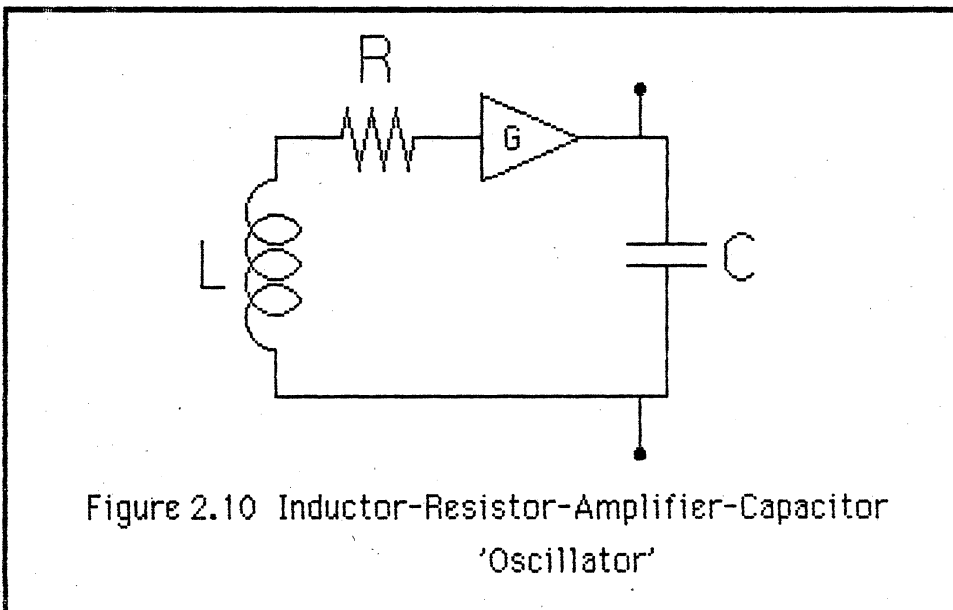
Figure 2.10 shows the circuit of just such a hypothetical oscillator with G as the gain of an amplifier; figure 2.11 shows a block diagram for this oscillator.

Figure 2.11 Harmonic oscillator with friction
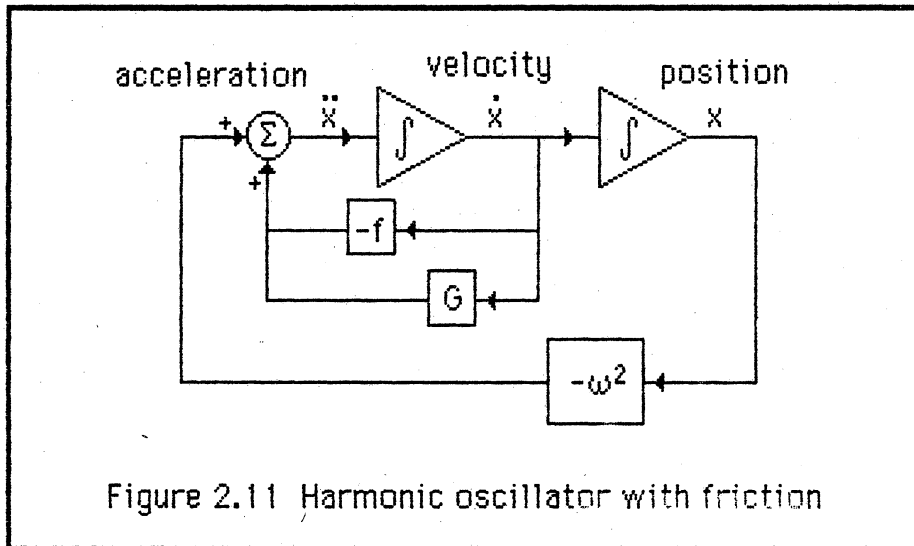
There are several problems with this approach. In general, the value of the constant, G, must be found experimentally. Slight differences between the actual friction, f, and the value, G, will cause the oscillator to either decay to zero amplitude or to grow exponentially, as shown in figure 2.12, until some system bound is reached.

too low   exact   too high

x(t)

38
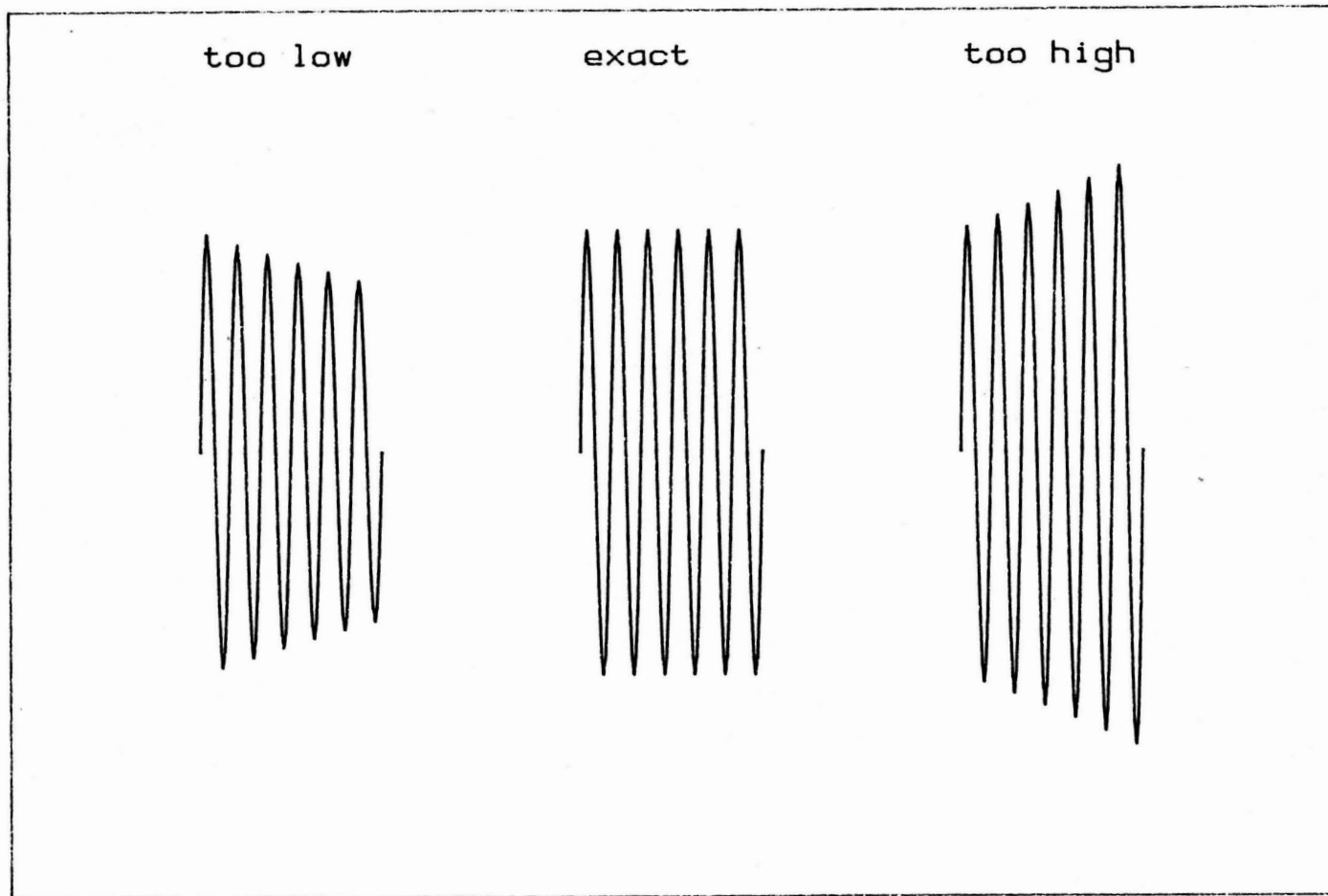
t

Figure 2.12 The effect of small changes in gain on a linear 'oscillator'

If the gain is too small then the oscillation decays away to zero. If the gain is too large then the oscillation expands exponentially without limit. There is only one exact value of the gain (for a given value of the friction, f) that gives a stable oscillation. The same behavior can be seen in figure 2.13 which is a plot in the phase plane. Gains less than the exact value needed to match the friction will give the solid trajectory depicted spiraling towards the center of the plane. Gains greater than the exact value needed to match the friction will give the trajectory expanding out into the plane. The only stable oscillator is depicted as the dotted line, which is the trajectory for only one, exact, value for the gain.

In practice, the "frictional" components vary slightly and in an unknown fashion with time; therefore, some provision must be made to vary the constant, G, with time. There are also random disturbance inputs in any real system that directly affect the state variables. Thus the amplitude and the frequency will be random variables. What is needed is some automatic method of adjusting the negative friction, G, to compensate for small system changes. This technique may be seen as *amplitude control*, modifying the gain to stabilize the non-ideal realization of a harmonic oscillator.

Before leaving this simple phase plane demonstration of stability, this system can be used to illustrate a similarity between physical and chemical oscillators. *Equilibrium* is defined as the condition of no change in the system with time; the velocities must all go to zero. For the systems shown in the phase plane of figure 2.13, the equilibrium position is at $dx/dt=0$ and $x=0$.
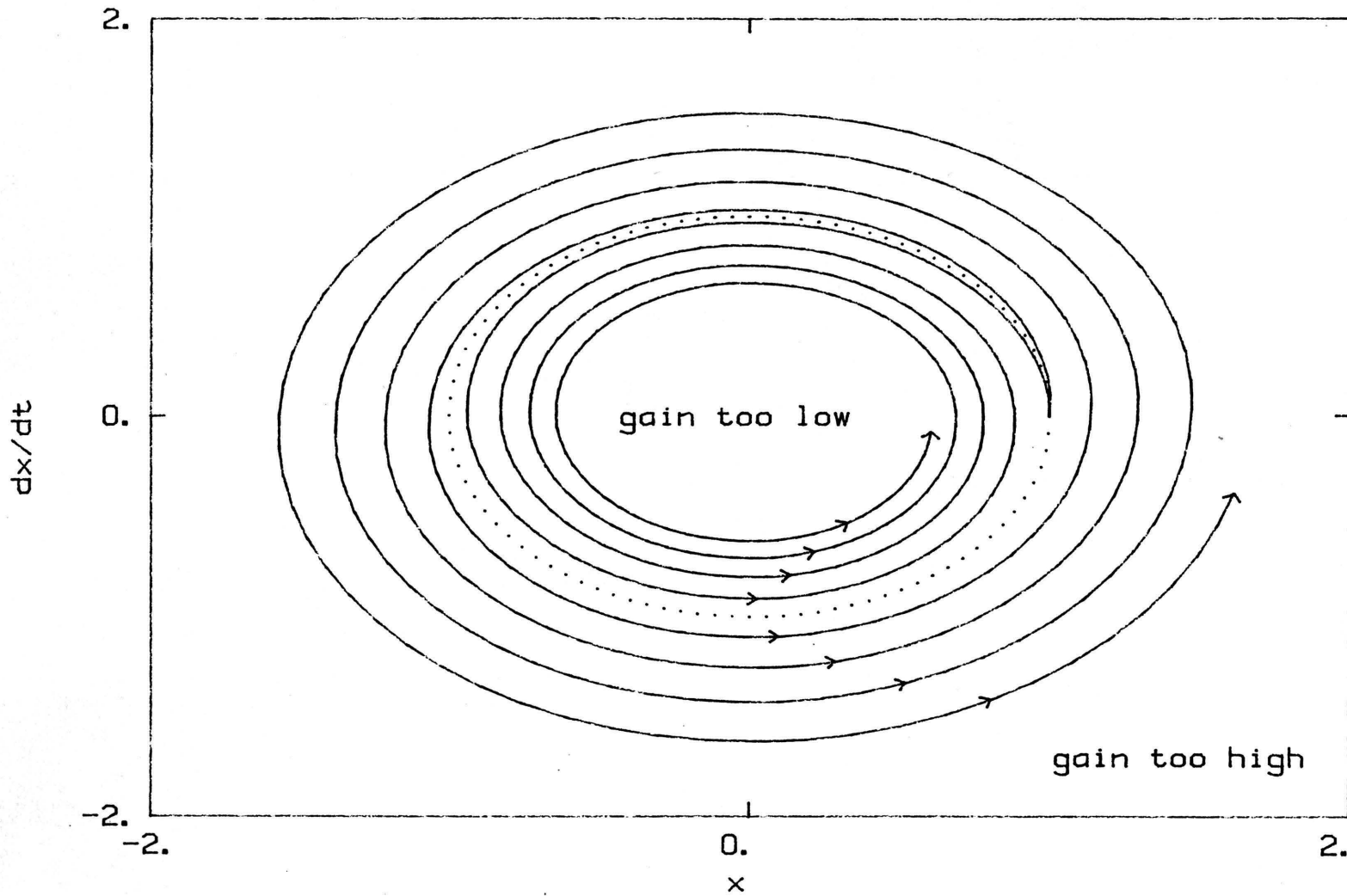
Figure 2.13   Phase Plane Plot of Linear 'Oscillator'

Note that the state variables never pass <u>through</u> the equilibrium point [Graham 1961, Nayfeh 1979]. If the system reaches equilibrium, it stays there! The fact that closed or periodic trajectories can never pass through an equilibrium point is true of all systems -- linear or not. This is true of chemical oscillators as well, and it is only far from equilibrium that chemical oscillators are possible [Prigogine 1967]. However this was a source of confusion for early workers in the field of chemical oscillators. For example, Field [1977] stated: *"Thus an oscillatory trajectory in a closed chemical system cannot pass through the point where all coordinates simultaneously have their ultimate equilibrium value"*. This is true enough, but Field went on to say: *"the way the trajectories of mechanical and electrical oscillators may"*, which is <u>not</u> true. If one looks only at one state variable at a time, it may indeed pass through the equilibrium value, but a single variable <u>does not</u> completely characterize the system! The use of a state variable approach clarifies the problem to the extent that this similarity between physical and chemical oscillators is apparent.

The 'oscillators' shown in figures 2.6 and 2.10 are the simplest cases of linear systems that exhibit fluctuations which are nearly periodic. As shown above, these simple linear systems cannot be used for practical oscillators because of the inescapable presence of friction, noise, and time-varying parameters. Perhaps a more complicated linear system might offer some hope for producing a linear oscillator. It is certainly easier to analyze and predict the behavior of linear systems. Accepting the presence of friction, is there another way to make an

oscillator described by a linear equation -- a *linear oscillator*?

In control systems theory and practice, the question of stability takes another form. Control systems should approach some desired value, known as the *setpoint*, in a smooth yet rapid fashion. Systems that diverge from the setpoint are known as *unstable*, and systems that oscillate with fixed amplitude about the setpoint are known as *marginally stable*. The same tools that can be used to analyze control system stability can be used for the analysis of linear oscillatory systems. This analysis of stability suggests that the simplest linear frictional system capable of a periodic response is a series of first-order systems. Each first order system is of the form,

$$\dot{x} = -ax + bu \qquad\qquad [2.38]$$

where a, b are constants

and u is an input to the system

For example, the three coupled first-order equations could be:

$$\dot{x}_1 = -x_1 + x_2 \qquad\qquad [2.39]$$
$$\dot{x}_2 = -x_2 + x_3 \qquad\qquad [2.40]$$
$$\dot{x}_3 = -x_3 + k\,x_1 \qquad\qquad [2.41]$$

Figure 2.14 shows this system, called a *phase lag oscillator,* in terms of an electrical realization, a circuit of resistors, capacitors, and an amplifier.

Figure 2.14 Phase Lag Oscillator
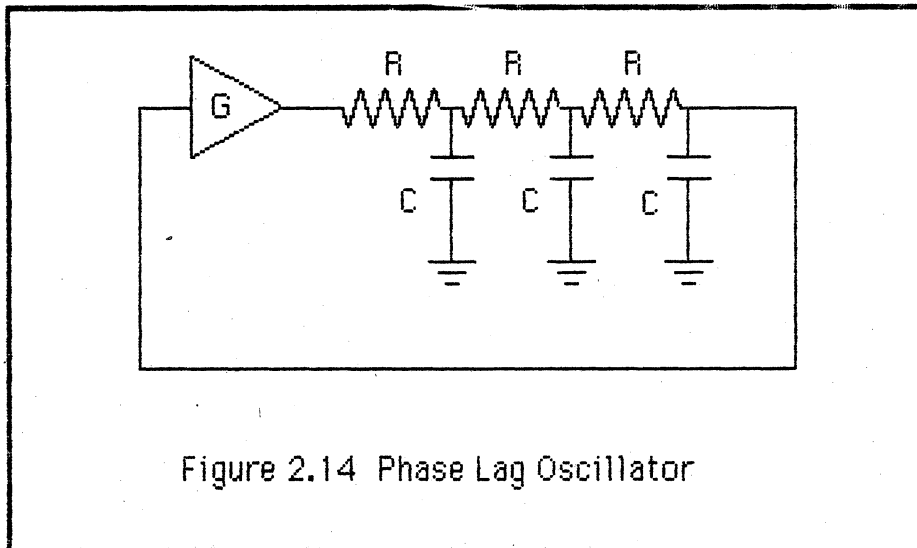
Expressing equations 2.39, 2.40, and 2.41 in standard form,

$$\dot{x} = A\,x \qquad\qquad [2.17]$$

where

$$A = \begin{bmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ G & 0 & -1 \end{bmatrix} \qquad\qquad [2.42]$$

An analysis of the eigenvalues of this matrix provides the most concise way to investigate this system as an oscillator. No matter how complicated, any linear system can be expressed in terms of a system of first order differential equations, and thus in terms of a system matrix, A, as in equation 2.17. In the simpler cases described earlier, system matrices 2.18 and 2.34 showed periodic behavior when the eigenvalues had exactly zero real parts and finite imaginary parts. So, what condition will give the more complicated systems purely imaginary eigenvalues? Without going into the details of the solution, the eigenvalues can be

calculated as a function of the gain constant, G.   Purely imaginary eigenvalues appear only when the gain,

$$G = -8 \qquad [2.43]$$

then the eigenvalues of this matrix are:

$$-3$$

and
$$+j\ 3^{0.5}$$
$$-j\ 3^{0.5}$$

The pair of purely imaginary eigenvalues indicates that the system in equation 2.17, with **A** as in equation 2.42, will have sine/cosine terms as its solution. The eigenvalue equal to -3 expresses itself as a transient term which quickly dies out. At _exactly_ a gain of -8, the system is periodic. But just as the system of equation 2.37 illustrated in figures 2.12 and 2.13, if the gain (G) is slightly high or slightly low the oscillations will either diverge or die out. No linear system will differ in detail in this matter. Small system imperfections or noise will destroy the perfect balance needed for a stable oscillator, causing the size or amplitude of the waveform to either grow without bound or to decay away to nothing. Since it is exactly this change of amplitude that is the manifestation of the imperfections of the linear 'oscillator', some means must be found to measure this amplitude and use this measurement to control the gain (G) in the system of figure 2.10, or indeed any linear system.

We have concluded that some means of measuring the amplitude is needed to control the amount of velocity feedback or the amount of gain.

One might think of the measurement of the amplitude in terms of an *amplitude functional*. Any periodic function has an amplitude, defined as the difference between the maximum and minimum values. Since *amplitude* is a measure of a function, the term *functional* is used. Kirk (1970) defines a *functional* as:

> A rule of correspondence that assigns to each function **x** in a certain class Ω a unique real number. Ω is called the *domain* of the functional, and the set of real numbers associated with the functions in Ω is called the *range* of the functional.

The amplitude functional has the domain of all periodic functions, and the range of positive real numbers. For the purposes of analyzing 'near' oscillators, one can stretch the point to include near-periodic functions in the domain. With a near-periodic waveform or periodic function as an input, the amplitude functional returns a value for the amplitude. Based on the measured value of the amplitude, the system can then be adjusted for stability. This is exactly the case in many electrical oscillators [Hofer 1979, Meyer-Ebrecht 1972, Oliver 1960, Reich 1961, Vannerson 1974 and 1975, Zimmerman 1959].

The essential fact is that <u>the amplitude functional is not a linear functional, and amplitude control cannot be achieved by a linear system</u>. The nonlinearity of an amplitude functional can be demonstrated in the following manner. The amplitude of a sine curve is twice the multiplier of the sine:

$$f(t) = W \sin(wt) \qquad\qquad [2.44]$$

$$\text{amplitude}(f) = 2W \qquad\qquad\qquad [2.45]$$

$$g(t) = V \sin(vt) \qquad\qquad\qquad [2.46]$$

$$\text{amplitude}(g) = 2V \qquad\qquad\qquad [2.47]$$

If the amplitude functional is a linear functional, then the additivity principle (equation 2.30) applies and:

$$f(t) + g(t) = W \sin(wt) + V \sin(vt) \qquad\qquad [2.48]$$

$$\text{amplitude}(f+g) = \text{amplitude}(f) + \text{amplitude}(g) \qquad [2.49]$$

$$= 2W + 2V \qquad\qquad\qquad [2.50]$$

Equations 2.49 and 2.50 are inconsistent, as can be seen by a simple example. Assume the following:

$$f(t) = \sin(t) \qquad\qquad\qquad [2.51]$$

$$g(t) = 0.5 \sin(2t) \qquad\qquad\qquad [2.52]$$

The amplitude of function, f, is undoubtedly 2.0, and the amplitude of function, g, is 1.0. The sum of the two functions is,

$$f(t) + g(t) = \sin(t) + 0.5 \sin(2t) \qquad\qquad [2.53]$$

The amplitude of this combined function can be found by the usual method of determining the value of the function at the point where the first derivative is zero and the second derivative is not zero.

$$(f+g)' = \cos(t) + \cos(2t) = 0 \qquad [2.54]$$

$$(f+g)'' = -\sin(t) - 2\sin(2t) \qquad [2.55]$$

Through the use of trigonometric identities, equation 2.54 becomes:

$$\cos(t) + \cos^2(t) - \sin^2(t) = 0 \qquad [2.56]$$

$$\cos(t) + 2\cos^2(t) - 1 = 0 \qquad [2.57]$$

The solution to this quadratic equation is:

$$\cos(t) = 1/2 \text{ or } -1 \qquad [2.58]$$

$$\sin(t) = 3^{0.5}/2 \text{ or } 0 \qquad [2.59]$$

If $\sin(t)=0$ then $(f+g)''=0$, so the extrema are at the points where $\cos(t) = 1/2$. The value of the amplitude is therefore:

$$\text{amplitude}(f+g) = 2 * [\sin(60 \text{ deg}) + 0.5 \sin(120 \text{ deg})] \qquad [2.60]$$

$$amplitude(f+g) = 3^{1.5}/ 2 \qquad\qquad [2.61]$$

$$amplitude(f+g) \approx 2.58 \qquad\qquad [2.62]$$

but,

$$amplitude(f) + amplitude(g) = 3.0 \qquad\qquad [2.63]$$

Thus, superposition does not hold, and the amplitude functional is not a linear functional. No combination of linear operations will yield a value of the amplitude.

Therefore, since a nonlinear functional is needed to measure the amplitude in order to control that amplitude, and any oscillator in the real world must have some means of amplitude control, then a practical oscillator must have some nonlinear element(s). The mathematical description of the oscillator must be nonlinear and the oscillator is then a non-linear oscillator. The only stable oscillators are nonlinear oscillators.

One can examine various candidate functions that give a measure of the amplitude of a waveform. An example nonlinear functional that provides a simple measure of the amplitude is the average of the square of the function. The analysis of equation 2.37, above, showed that some means of adjusting a so-called 'negative' friction would stabilize an oscillator. Let the multiplier for the velocity (dx/dt) term in equation 2.37 be:

$$e(1-x)^2 \qquad\qquad [2.64]$$

where e is a constant

If the average value of x is less than 1, then this term will tend to be greater than zero, and will cause the amplitude to increase. Conversely, if the average value of x is greater than 1, then the nonlinear term (2.64) will tend to be less than zero, and will cause the amplitude to decrease. This is the self correcting nature of an amplitude control system. With the term in 2.64 as the multiplier to the velocity term in equation 2.37, the resulting differential equation is known as the van der Pol equation [Graham 1961, Nayfeh 1979, van der Pol 1926],

$$\ddot{x} + e\,(1-x\,)^2\,\dot{x} + \omega^2\,x = 0 \qquad\qquad [2.65]$$

Figure 2.15 illustrates this equation in block diagram form.

Figure 2.15 Block diagram of Van der Pol oscillator

e= 0.1

e= 3

e= 6

e=10

e=40

normalized time
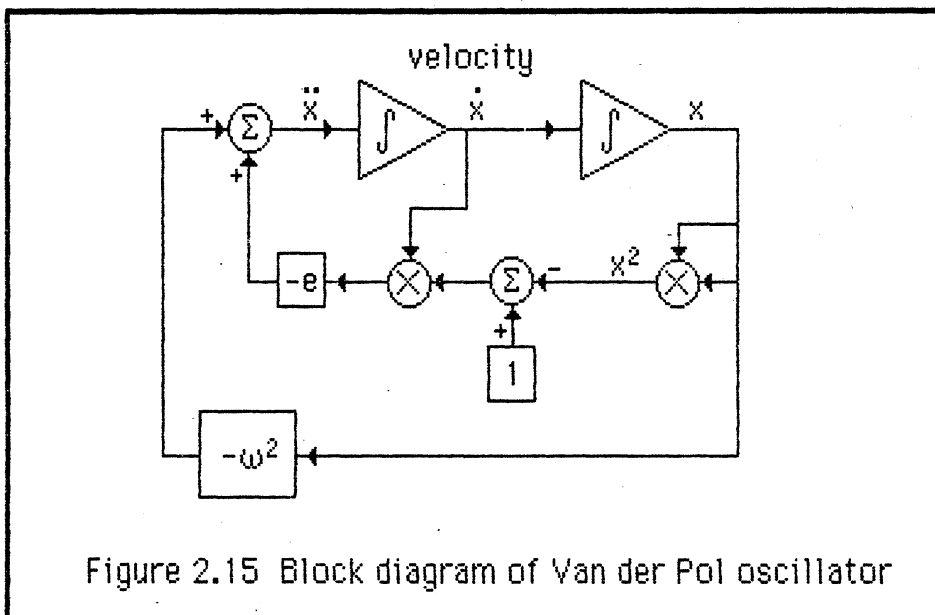
Figure 2.16   Solution of the Van der Pol equation
for various degrees of nonlinearity
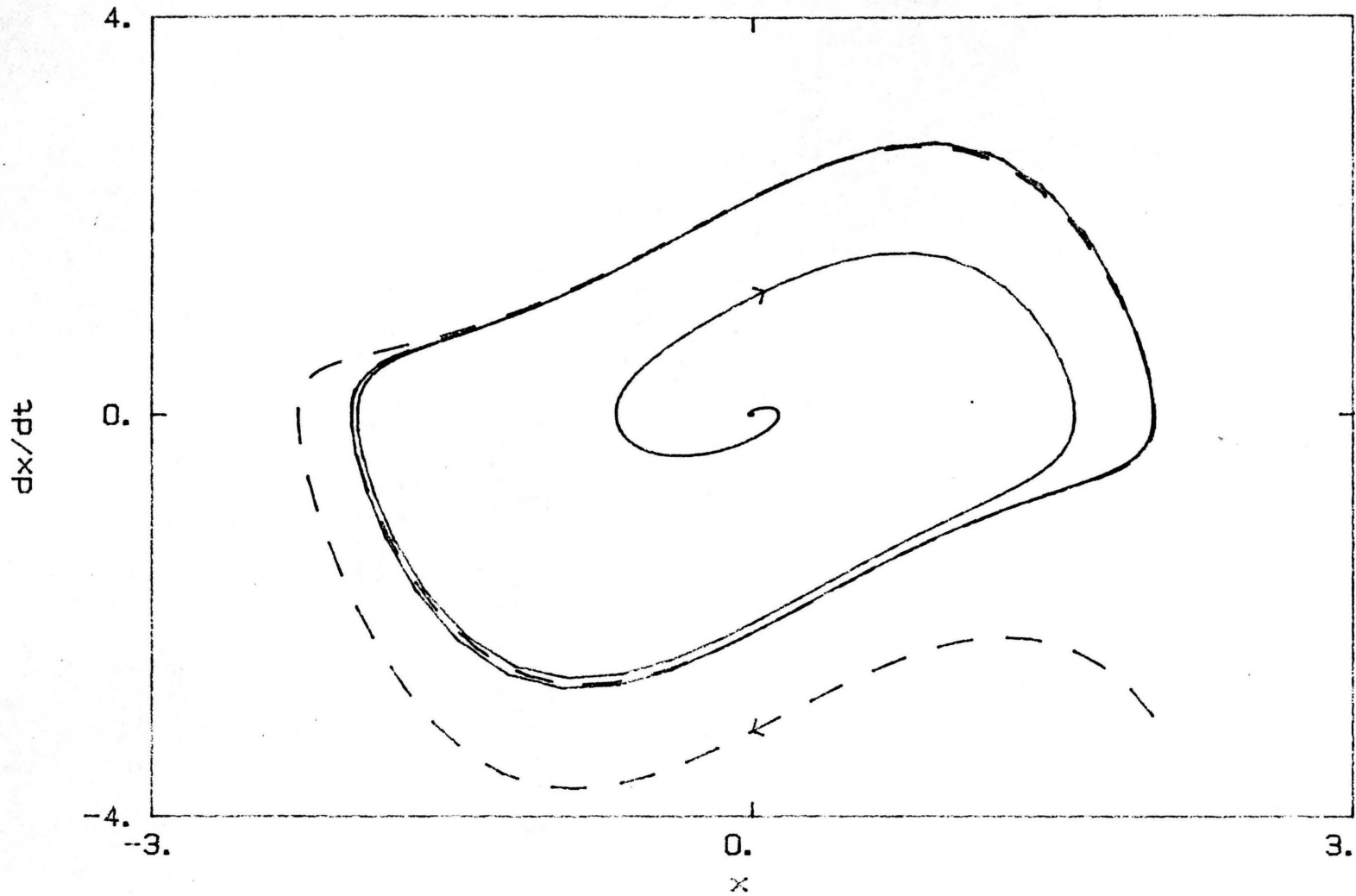
Figure 2.17   Phase Plane Graph of Van der Pol Oscillator (e=1)

In the limit as the constant, e, approaches zero, the equation reduces to the familiar harmonic oscillator. Note in the accompanying graph (Figure 2.16), when the value of e is about 0.1, the wave is very close to a sine function. As the value of e increases, the periodic waveform becomes less and less sinusoidal. At large values of e, greater than about 10.0, the wave is almost a square wave. Figure 2.17 is the phase plane depiction of the van der Pol oscillator with the value of e=1. There are two trajectories illustrated. The solid line shows the system starting at x=0 and dx/dt=0; this trajectory grows out until it reaches a stable periodic trajectory. The dashed line shows the system starting at x=2 and dx/dt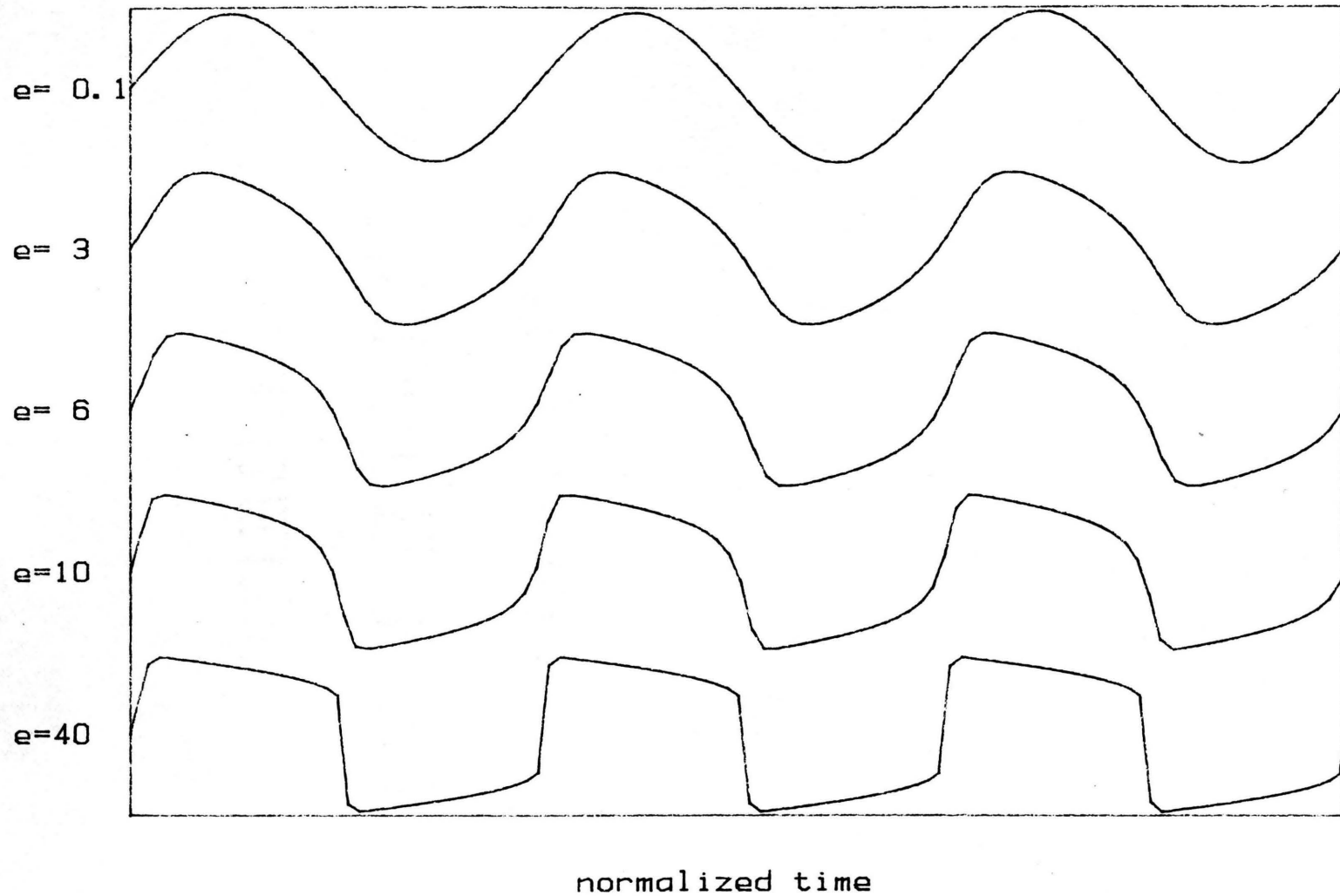=-3; this trajectory moves in until it too reaches the stable periodic trajectory. This periodic trajectory is known as a *limit cycle*. The final amplitude is stable and small perturbations in the system will cause only transient departures from the stable oscillation -- this is a practical oscillator.

At the larger values of the constant, e, the abrupt switches in the wave are characteristic of a type of oscillator known as a *relaxation oscillator* (also known in German as a *Kippschwingungen* or "tipping oscillator" [Brockhaus 1983, Grassman 1971]). The system slowly changes, or relaxes, until it has reached a certain point, and then it rapidly changes ("tips over") to re-establish a new condition. This is a common oscillator in electronics. A widely used integrated circuit, known as a 555 timer, which is often used as an oscillator [Lancaster 1974, Rony 1979], is a relaxation oscillator. The integrated circuit is shown in figure 2.18, and figure 2.19 shows a simple oscillator based on this device.

53

Figure 2.18 555 timer integrated circuit

Figure 2.19 555 Oscillator
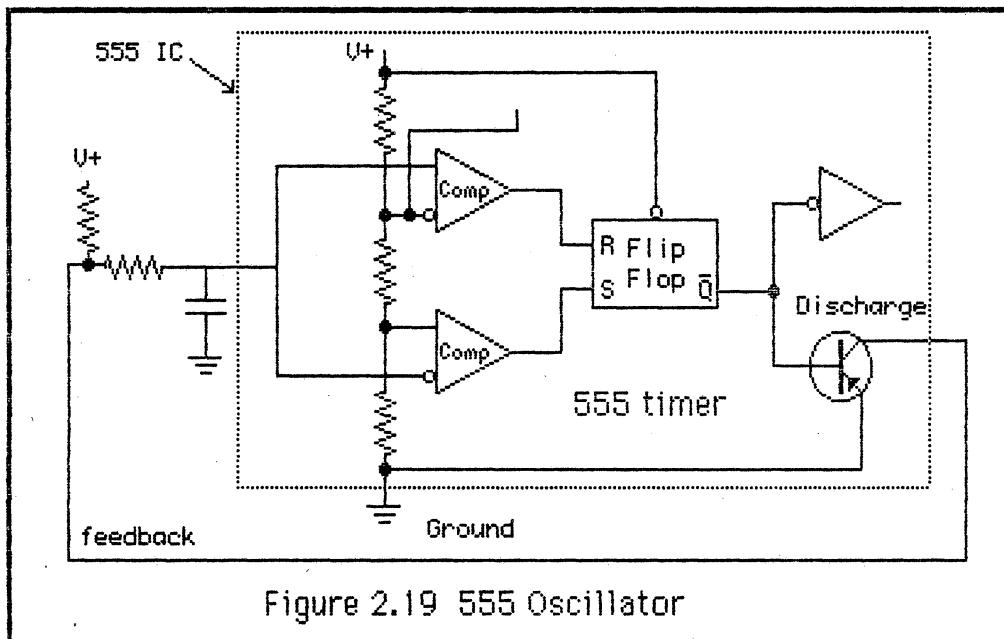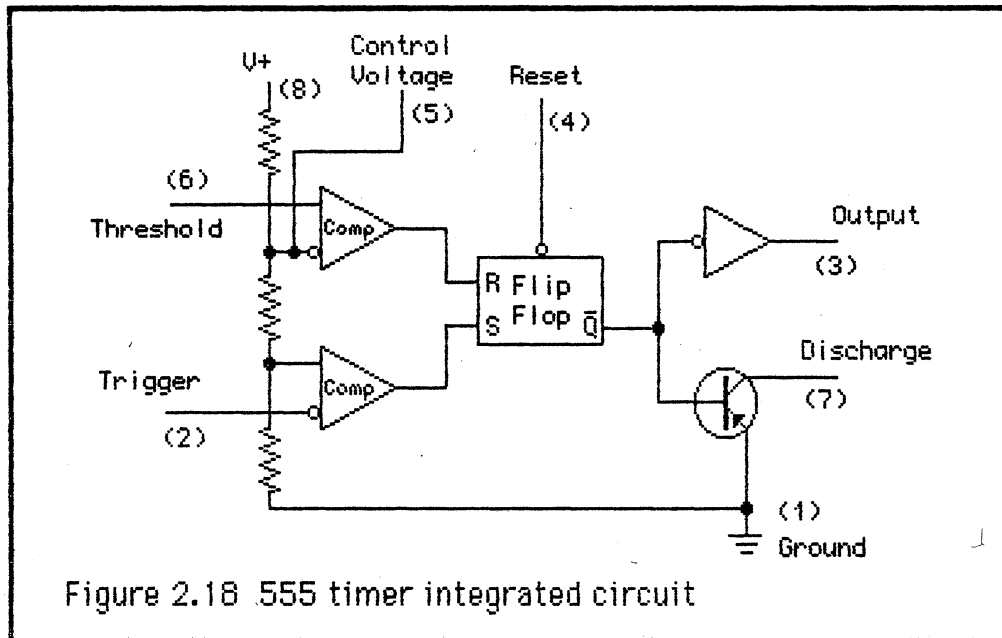
The waveform of the oscillator measured at the capacitor is shown in figure 2.20 below; the phase plane plot at the capacitor is shown in figure 2.21 below.

Figure 2.20 Waveform of 555 Oscillator

Figure 2.21 Phase plane plot for 555 oscillator
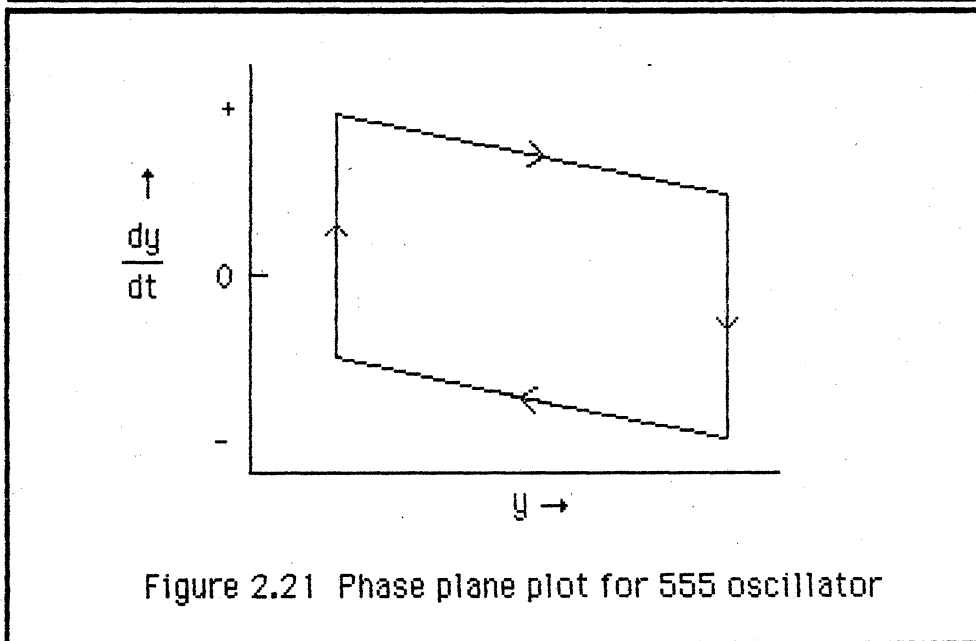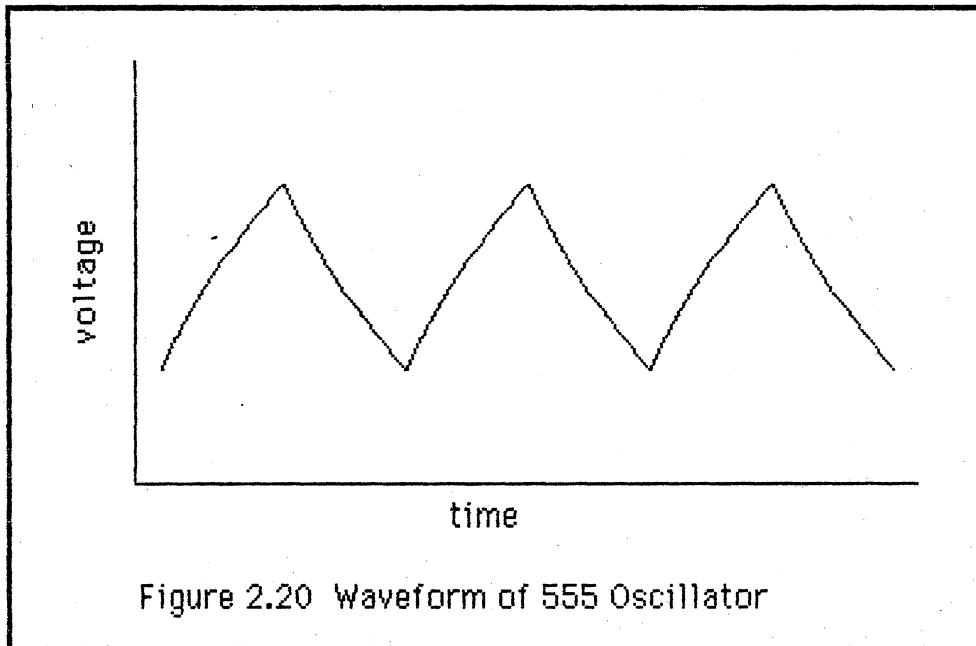
Note the discontinuities, which are characteristic of a relaxation oscillator.

As seen in the van der Pol oscillator (figure 2.16), there is a continuum between the harmonic oscillator and the relaxation oscillator.

The ideal sine wave and the ideal square wave are two extremes in oscillation, one characterized by smooth functions, and the other characterized by discontinuous or near discontinuous periodic functions. The smooth sinusoid extreme is observed in systems where the stabilizing nonlinearity is a minor contributor to the dynamics, while the abrupt relaxation oscillator extreme is seen in systems where nonlinearities are the dominant elements determining the system dynamics.

In summary, three requirements have been identified for a stable physical oscillator: 1) there must be some sort of feedback; a dynamic variable must be coupled to itself in such a way that it affects its own rate of change; 2) there must be some source of energy to counteract the losses due to "friction", and 3) there must be some non-linear element that acts to control the amplitude of the waveform and thus control the stability of the oscillator.

Chapter 3. The development of a generalized chemical oscillator.

*"If we have a correct theory but merely prate about it,
pigeonhole it and do not put it into practice, then that theory,
however good, is of no significance"*

*Mao Tse-tung "On Practice"
Selected Works, 1, p 304
(July 1937)*

If one examines the chemical oscillators in the literature, one is immediately struck by the degree of chance involved in their discovery. As discussed in Chapter 2, until recently one could well refer to them as "found" oscillators. Even the Brandeis group's synthesis of chemical oscillators relies on "found" kinetics. All of the oscillating systems designed by the Brandeis group have halogen-oxygen chemical species as an intrinsic part of the recipe. Because of the uniqueness of the kinetics of such systems, there is not much hope that oscillators based on these species will allow coupling to many other different chemical species. Because of this, the present work has tried to devise a more general system.

The history of this project on the oscillogenic instrument represents the fortunate convergence of long-standing interest in chemical oscillators by two individuals.

On 29 February 1963, Peter R. Rony, while a graduate student at the University of California at Berkeley, asked himself the question, "what is a chemical oscillator?" [Rony (1963)] Reasoning by analogy to electrical and mechanical oscillators, he knew that the simplest way to make a harmonic oscillator required two energy storage systems, and a means to

pass energy between them. In mechanics, these quantities are the potential energy stored in a stretched or compressed spring and the kinetic energy of a moving mass. In electronics, the two energy storage systems are frequently the potential energy stored in a capacitor and the kinetic energy stored in the magnetic field of an inductor. Rony asked, what are the equivalents to these quantities in chemistry? What is chemical inductance, chemical capacitance, or chemical conductance? Similarly, what are chemical resistors, chemical capacitors, chemical inductors, chemical diodes, chemical amplifiers, chemical integrators? [Rony (1963)] What new chemical transducers could be conceived and reduced to practice as practical devices as a part of this analogy. By analogy to photons and phonons, could a quantity called a chemon exist? These questions, though unanswered at the time, continued with Rony for almost two decades and were one reason why he decided in the mid-1970's to pursue electronics and computers more deeply.

In December 1970, on the way to a final examination in biochemistry, Peter Kip Mercure was struck by the 'simultaneity' of the biochemical reactions which he had been studying. The well known charts of metabolic pathways have many reactions which are progressing more or less at the same time; some pathways are turned on and some turned off at any particular time. He reasoned that the dynamics of biological systems must be related to inter-relationships between the various reactions. With this background, he by chance encountered two articles on chemical oscillators in the spring of 1972 [Degn(1972), Field(1972)]. Here the thought was presented that the Belousov-Zhabotinskii reaction was

formed of two different reaction pathways with dynamics such that first one and then the other would become active. Because of the similarity to the biochemical pathways, this thought fell on prepared ground in Mercure's mind. In the winter of 1974, Mercure prepared a lecture-demonstration on the Belousov-Zhabotinskii reaction for the chemistry research department of Chrysler Corporation, where he was employed. Meanwhile, his interest in dynamic systems continued. When inexpensive integrated circuit operational amplifiers first became widely available, Mercure made some experiments using small analog computers for simulations. And when microprocessors were introduced, he saw them as a tool for simulation.

The major reason for entering the Virginia Polytechnic Institute and State University was Peter Rony's work there with microcomputers. After entering graduate school, one of the items which he discussed with P. R. Rony was the work and interest in chemical oscillators. At VPI, he received a formal introduction to dynamic systems via course work in chemical engineering, electrical engineering, engineering science and mechanics, mathematics, and computer science. With Mr. Mercure's new knowledge of engineering techniques, he mentioned in 1981 the possibility of studying chemical oscillations in continuous stirred tank reactors (CSTR's). It later turned out that other investigators were already doing just that, but it was a start to the engineering of chemical oscillators.

Meanwhile Rony was still concerned with the conflict of why it appeared to be so easy to make a physical oscillator and yet so difficult to make a chemical oscillator. As engineers, Rony and Mercure were

familiar with the stability analysis of process control systems. Specifically, one must go to a great deal of trouble to avoid oscillations in process control systems. One key element fell into place when he realized that chemical control systems can <u>be</u> oscillators by the use of external feedback. That is, by the use of feedback outside the chemical system. Peter R. Rony proposed (6 March 1982) that a general chemical oscillator could be created by using an external feedback system with gain. Using linear stability analysis, Rony and Mercure expected the resulting waveforms to be sinusoidal.

Controller & Process

| Electronic System | Input | Physical System | Output |

feedback

Figure 3.1 Relationship of Process Input to Output
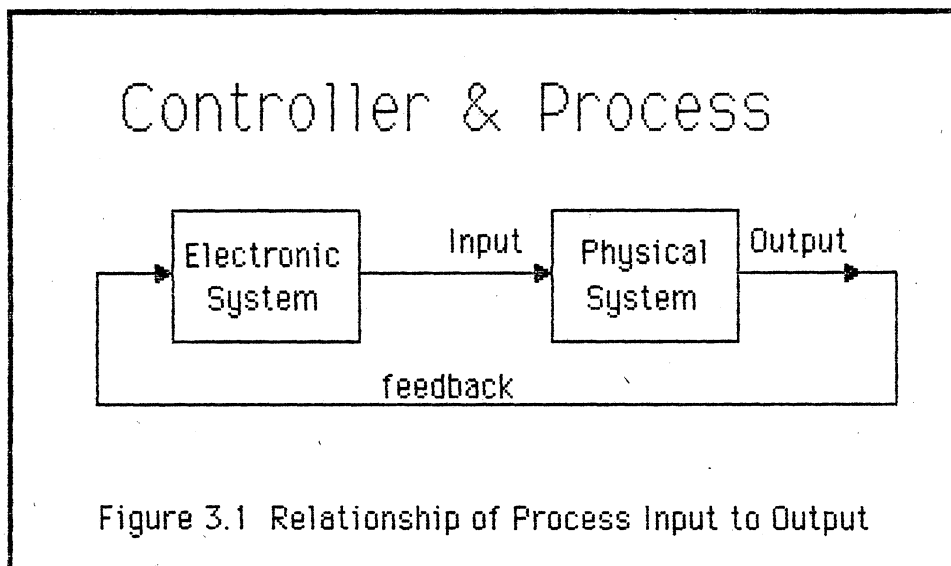
Adding external electronic feedback to a physical system is coupling the chemical system to some electronics in much the same way as coupling two different reactions. On the other hand, the electronic system may be considered a control system, and the physical system as the process. As in control systems, one speaks of an input variable which is manipulated, and an output variable that is sensed, see Figure 3.1. The

lines with arrows in this figure represent the *flow of information* in the system. *Feedback* is the return of <u>information</u> from the latter stages of a system to the earlier stages. Rather than the transfer of mass, energy, or momentum, control systems transfer information. In control systems, feedback involves a sensor, that can measure some physical quantity. The sensed quantity must be related in some way to the parameter which the instrument is designed to measure. For example, the concentration of a component, A, can be measured in an instrument which senses another component, B, if the concentration of A is coupled in some way to the concentration of B. Candidates for sensed quantities in a <u>chemical</u> oscillogenic instrument are: the concentration of a component, the emission of light by a reaction, the absorption of light by a component, and the current flowing in an electrochemical cell. In addition, the external feedback would require an output, a device which could control some chemical quantity. Candidates for manipulated or input variables in a <u>chemical</u> oscillogenic instrument are: addition of chemical reagents to a system, adding photons to a system, varying the temperature of a system, and changing the voltage applied to an electrochemical cell. The chemical system must have an input (manipulated variable) and an output (sensed variable), and the oscillogenic feedback connects these two variables in such a way that the system oscillates. In homogeneous chemical oscillators, there is some chemical path internal to the system which serves as feedback; it is possible to substitute *information feedback* and produce a more general chemical oscillator.

An information feedback method and a chemical system can combine

to make an oscillator. As a demonstration or 'reduction to practice' of this idea, a model system was sought for a chemical oscillogenic instrument. The sensed and manipulated variables were the first to be considered.

Because the experimental system was expected to be connected to a digital computer for the experiments, and the external feedback could easily be handled electronically, it was natural to think of using an electrochemical system for a model oscillogenic instrument. Additionally, it was hoped that an electrochemical system could respond rapidly enough to give frequencies in the audio range. The current homogeneous chemical oscillators, such as those designed by the Brandeis group, operate with periods on the order of minutes. If the oscillator operates with a period of milliseconds, then the measurements of frequency can be made much more quickly. A quantum jump in speed of measurement was also desired to dramatically illustrate the differences between the present approach and the existing approaches.

After the model system was chosen, the nature of the external feedback was next to be defined. Mercure and Rony were thinking in terms of control theory and linear systems analysis, and the feedback was considered to be a pure gain or perhaps a linear system adding some time lag. For example, chapter 2 of this work showed linear systems with a gain element which critically controlled the oscillation.

The methods of *frequency response analysis* can be used to predict the performance of nearly linear oscillators. For example, an oscillator of the previous chapter (Figure 2.7, equations 2.36 – 2.38) can

be examined by breaking the feedback loop and observing the response of the circuit to a sine wave input. If one compares the input and output of a system, the results will be similar to Figure 3.2.

Figure 3.2 Frequency response analysis
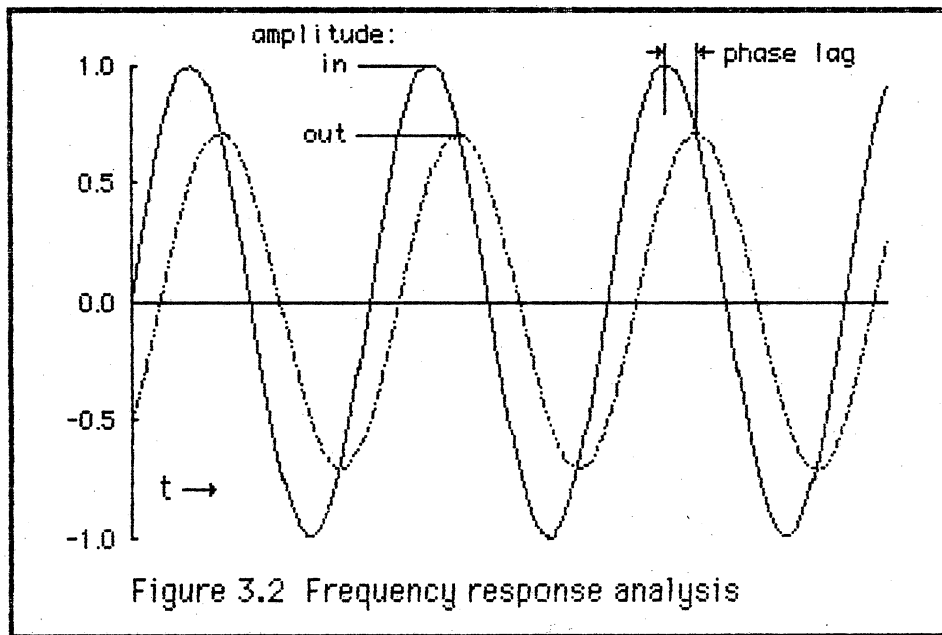
The output of this linear system will be a sine wave of different amplitude than the input and shifted somewhat. The change in amplitude is characterized by the ratio of the two amplitudes: the *amplitude ratio*. The shift is called a *phase* difference, and is characterized by a *phase angle*. A full cycle of the sine wave is considered 360 degrees, and the difference between the two waves is expressed in angular units. Because the output wave in Figure 3.2 peaks somewhat later in time than the input, the phase of the output is said to *lag*. It is the *phase lag* of the circuit of Figure 2.7 which allows the oscillation to occur, thus this circuit is known as a *phase lag oscillator*. Graphs of the logarithim of the amplitude ratio and the phase angle as functions of the input

frequency, known as Bode plots, are used to display the behavior of a system. The Bode plots for the phase lag circuit of Figure 2.7 are: 1) the gain plot in Figure 3.3 and 2) the phase plot in Figure 3.4.

One of the results of frequency response theory states that for a stable periodic system the net gain around the feedback loop will be unity and the phase angle will be -180 degrees. In Figure 3.4 the phase angle condition is met at a frequency (ω) of 1.73 Hz. The amplitude ratio at this frequency is 0.125 (log(amplitude ratio) = -0.9 in Figure 3.3). Note that the product of this ratio and the predicted gain for a stable oscillator (k = 8, equation [2.40]) is unity as required by frequency response theory.

Applying frequency response methods to electrochemical systems suggests that some variation in the gain and phase lag of an electrochemical system would be the controlling factor in the system oscillation. The frequency response of electrochemical reactions has been well studied. The usual mathematical approach has been to calculate the frequency response for approximations of electrochemical behavior [Macdonald(1977)]. For the first and simplest approximation, the electrochemical reaction is assumed to be an extremely fast reaction (essentially infinite rate). The Bode plots for this case are shown in figures 3.5 and 3.6. The phase angle of a relatively rapid reaction (called reversible by the electrochemists) is independent of the frequency (Figure 3.6), and it is this phase angle which controls an oscillator with linear feedback.

Clearly, the frequency of an electrochemical oscillator with a linear feedback loop would be unresponsive to the chemistry of this "reversible"

Figure 3.3 Bode gain plot for phase lag oscillator

Figure 3.4 Bode phase plot for phase lag oscillator

65

Figure 3.5 Bode gain plot,
reversible electrochemical reaction

Figure 3.6 Bode phase plot,
reversible electrochemical reaction
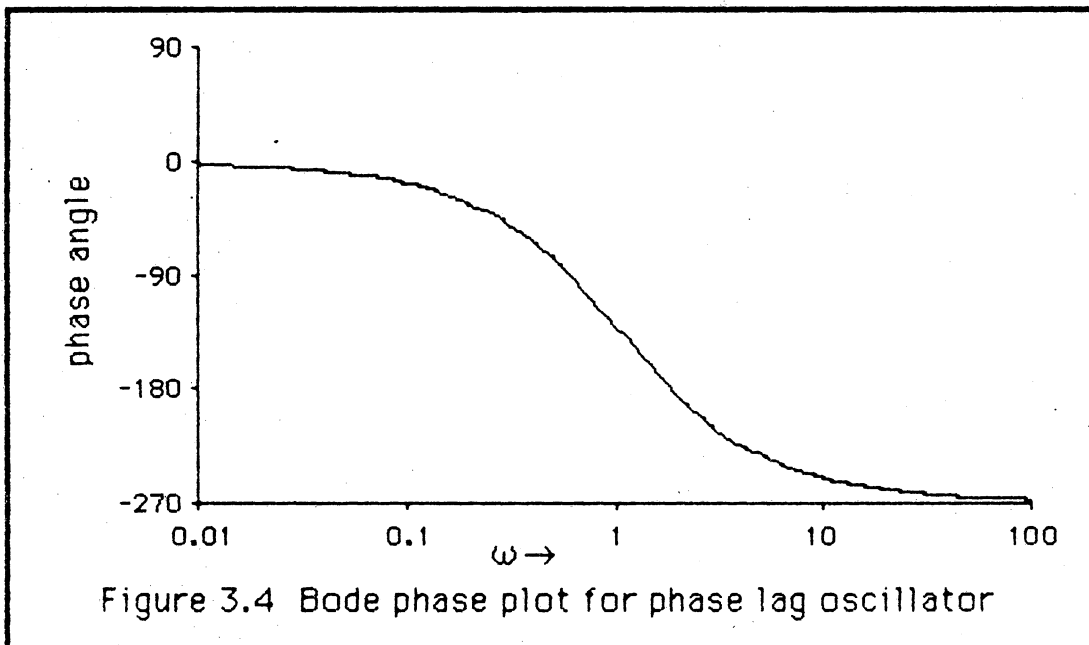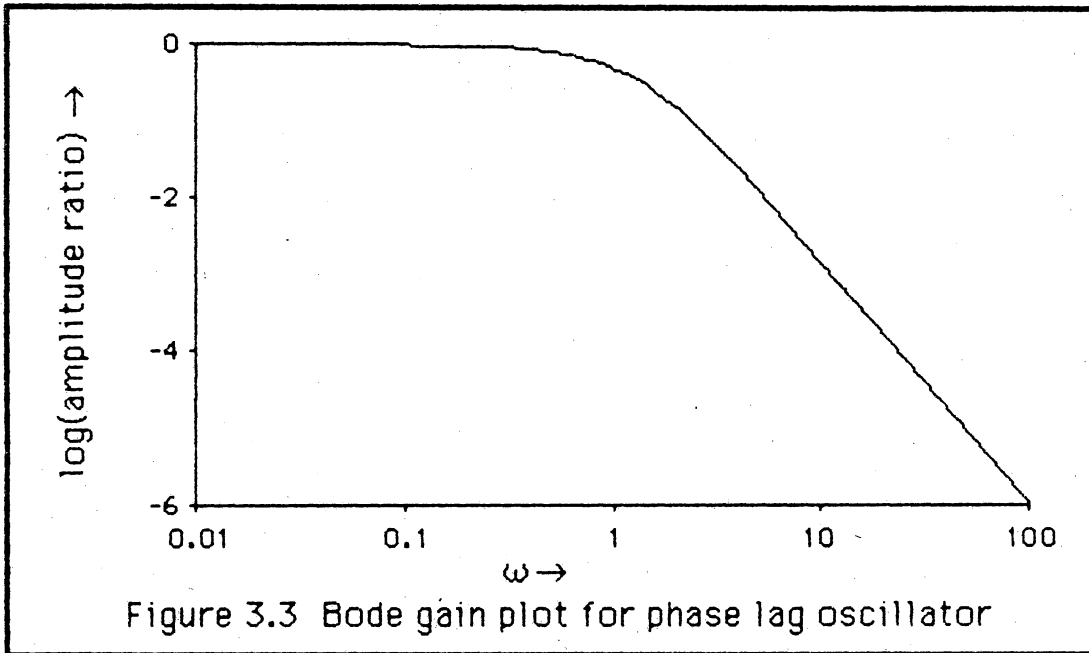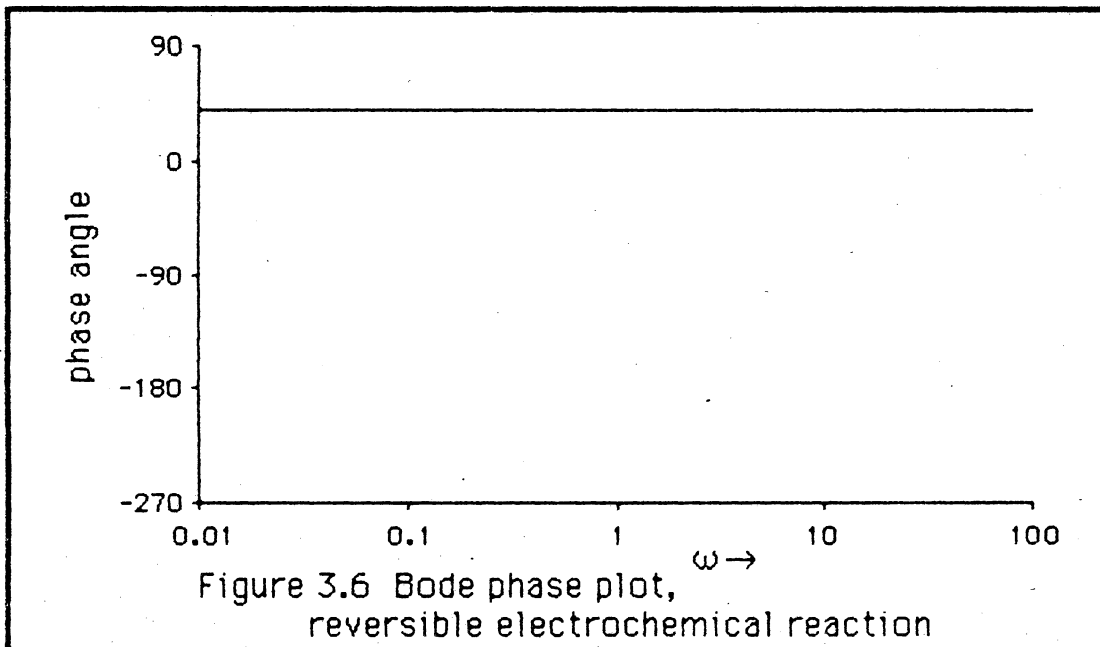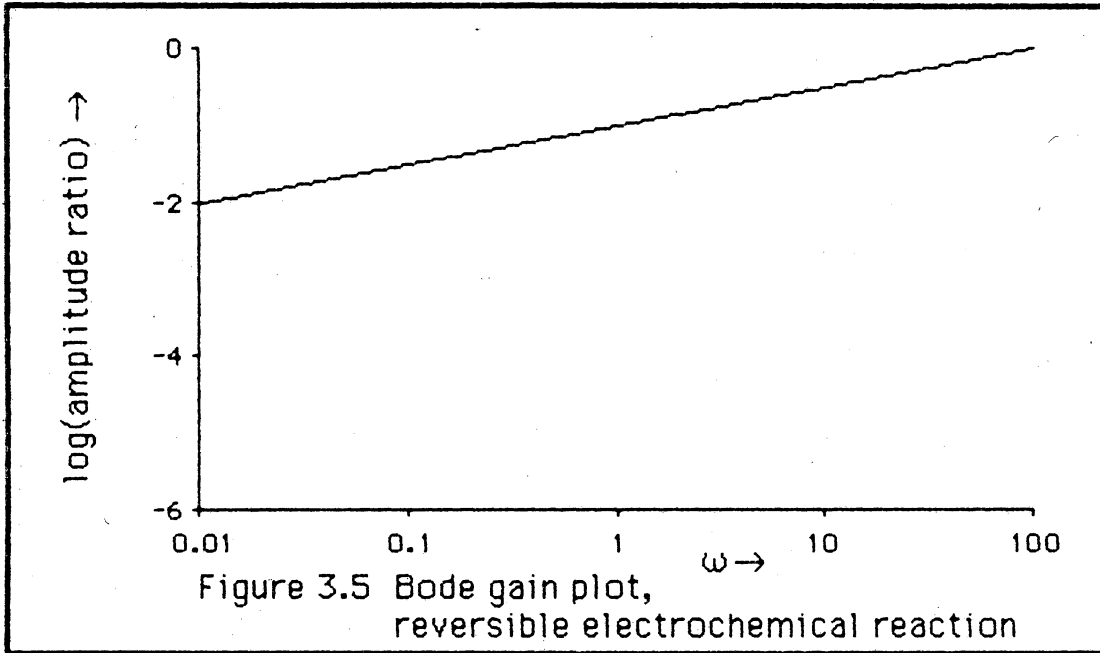
reaction. A more realistic electrochemical approximation requires all reactions to have a finite rate. This type of reaction is called *quasi-reversible* by the electrochemists. Figures 3.7 and 3.8 are the Bode plots for such a system.

The phase angle of the quasi-reversible electrochemical system shows a <u>very</u> weak dependence on the frequency. An oscillator with a linear feedback path would <u>not</u> be very sensitive to changes in such a chemical system. The idea of an electrochemical oscillogenic instrument was not abandoned. But, the work on a mathematical model of the electrochemistry was commenced without a clear idea of how the oscillator would be formed.

A study of electronic oscillators [Oliver (1960), Gaver (1966), Meyer-Ebrecht (1972), Vannerson & Smith (1974, 1975), and Hofer (1979)] showed that some automatic gain control was needed for a stable oscillator. This led to the ideas of non-linear electronics to control the amplitude. Not only must an oscillator have a stable amplitude but the oscillator must be started in some way. The system must somehow make the transition from non-oscillatory to oscillatory. A system which can start from zero amplitude and progress to oscillation is *self-starting*. This is a desirable property for an oscillator and is related to amplitude control. An example of a *self-starting* oscillator is the Van der Pol oscillator discussed at the end of chapter 2. Non-linear elements of the external feedback might make the oscillogenic instrument both self-starting as well as stable.

Figure 3.7 Bode gain plot,
         quasi-reversible electrochemical reaction

Figure 3.8 Bode phase plot,
         quasi-reversible electrochemical reaction
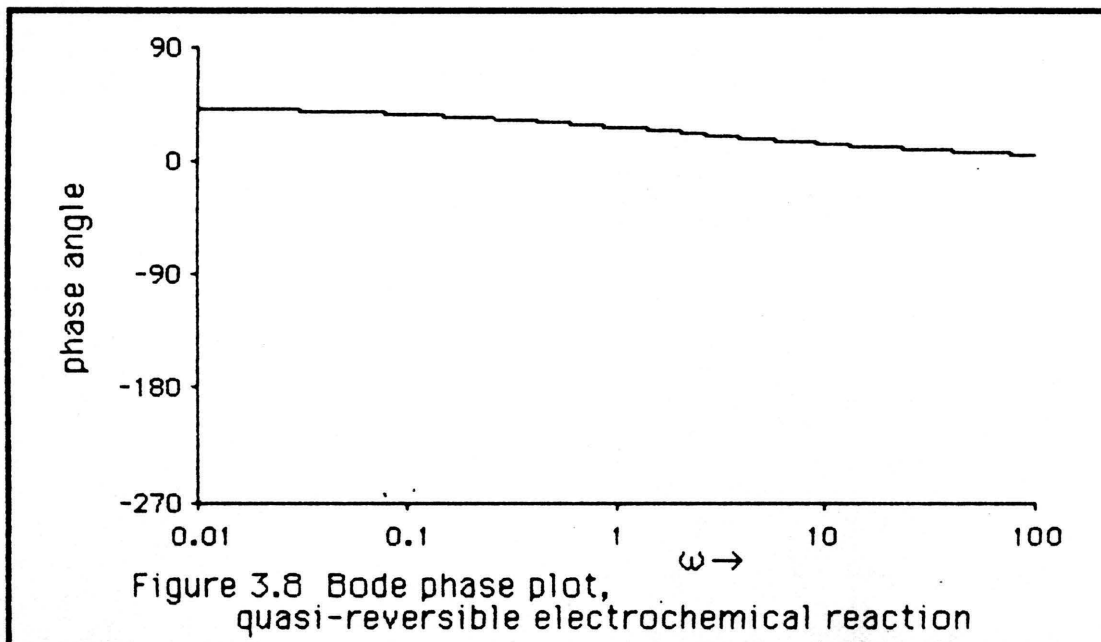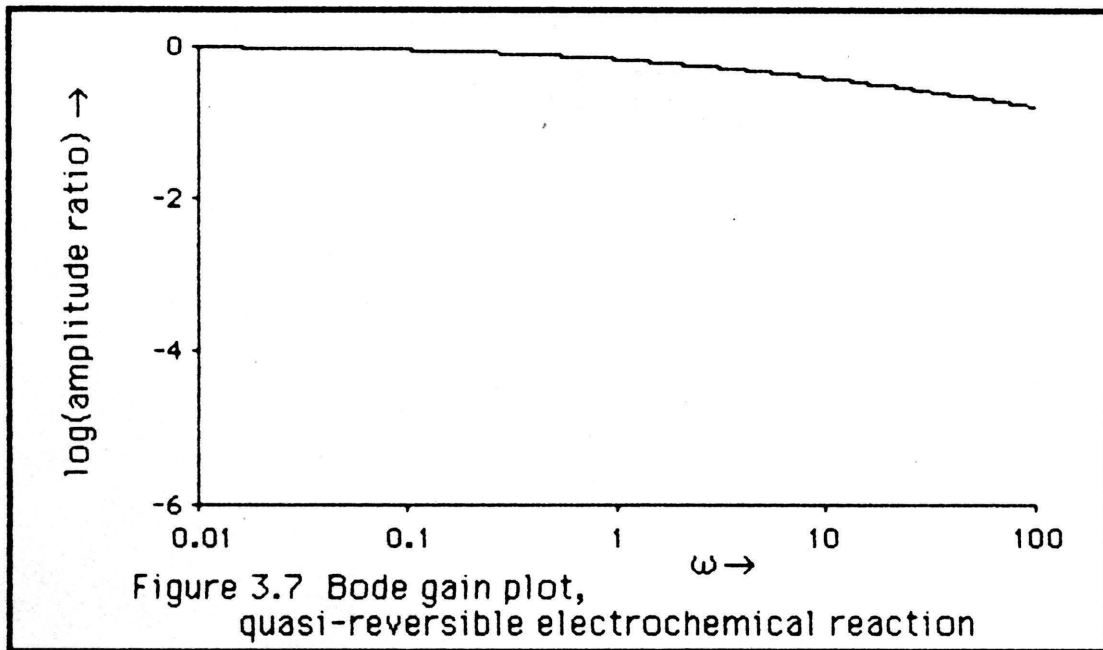
At this point, it seemed to Mercure that an electrochemical oscillator could not rely on the same oscillation mechanism as the 'linear' or phase lag oscillators. But, the exact nature of an alternate mechanism was still not apparent.

The other shoe dropped, so to speak, on the 29th of August 1983 while Mercure was studying results of the Brandeis group [Epstein 1983a,b, De Kepper 1981a,b]. If one examines the waveforms of the customary chemical oscillators, the shape seems to be discontinuous in nature. Figure 3.9 shows the waveforms for a diverse series of oscillators (these plots were redrawn from the original literature reports): 1) harmonic, 2) oxidation in a flame [Griffiths 1985], 3) Belousov-Zhabotinskii [Tyson 1985], 4) electrochemical [Wojtowicz 1972], 5) designed oscillator from the Brandeis group [de Kepper 1981a], and 6) van der Pol (see the end of chapter 2 of the present work). The waveforms can be compared to the harmonic (or linear) oscillator, and the van der Pol relaxation (non-linear) oscillator. Chemical oscillators are relaxation oscillators! Here is the reason for the differences between physical and chemical oscillators: the nonlinearities dominate chemical oscillators while they can be minimized in physical oscillators. Nonlinearity is an integral part of chemistry, from activation energy and the effect of temperature on reaction rate to the common nonlinear expressions for reaction rates. Thus, in the continuum from harmonic to relaxation oscillators, chemical oscillators tend toward the relaxation end. This prompted the thought that a generalized chemical oscillator could be a relaxation oscillator.

amplitude
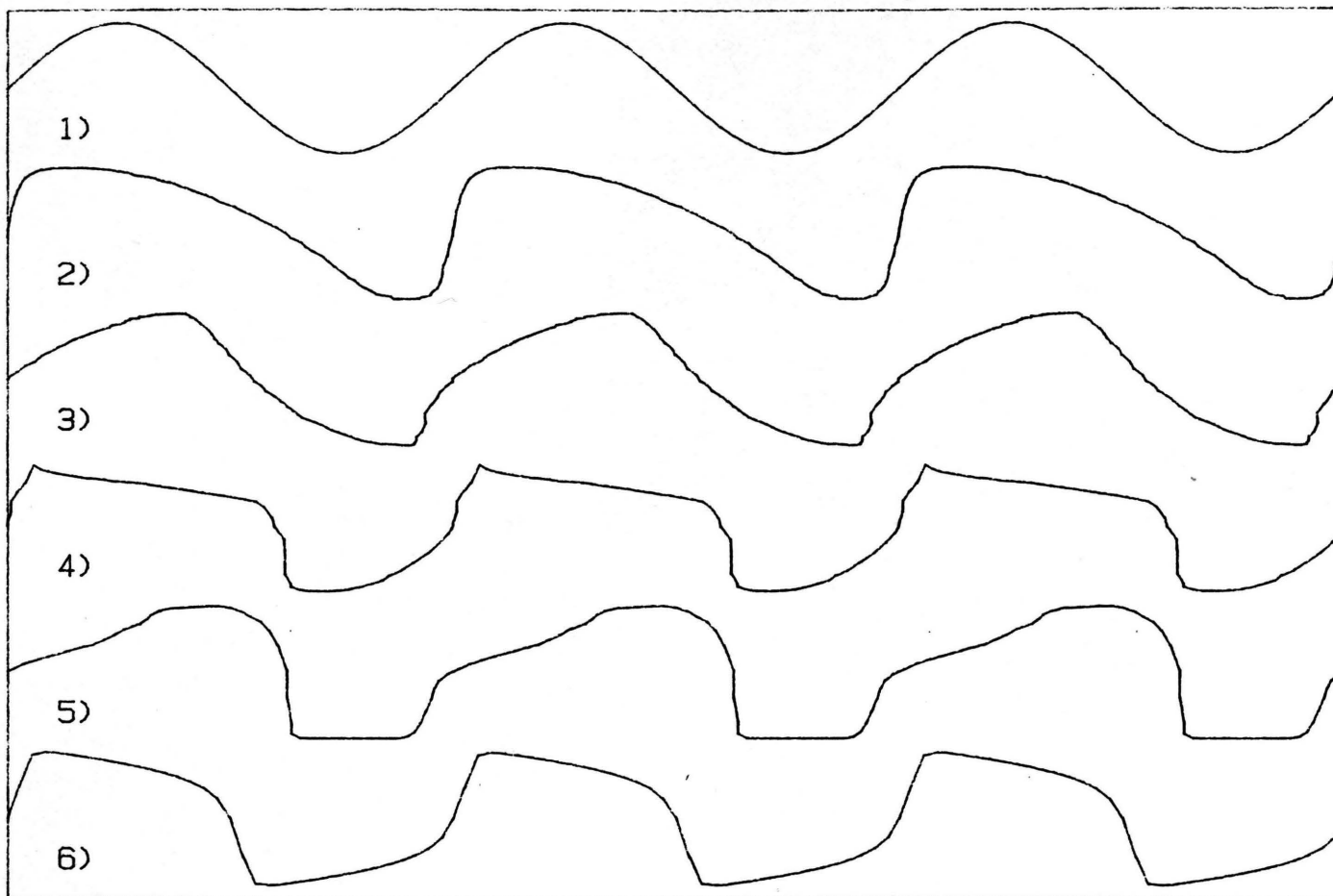
time

Figure 3.9   A range of oscillators: 1) Linear, 2) Flame
3) B-Z, 4) Electrochemical, 5) Brandeis, 6) van der Pol

What simple system could one conceive that would allow many different types of chemical systems to oscillate? Taking an intuitive approach to oscillator design, what is the nature of oscillators? The feedback in oscillating systems must act counter to the velocity at times. The velocity is the rate of change of a quantity with time, dx/dt. In an oscillator, there must be forces which, at times increase a velocity, and at times decrease that velocity. For example, after passing through the bottom of its swing, the velocity of a pendulum is slowed by the action of gravity. Similarly, a chemical system must act such that a product of the system will affect the rate of generation, dc/dt, of that product. This is the basic principle of "feedback". In a generalized chemical oscillator, one of the "products" of the system is the output of the sensor used in the external feedback. Thus this "product" must indirectly affect its own rate of change. This means, of course, to change the input to the chemical system via the controlled variable or final control element in the external feedback loop.

To illustrate these intuitive principles of oscillators, consider a circuit that is perhaps the simplest practical electrical oscillator [Goldblatt (1984)]: an integrator coupled to a hysteresis element. This circuit is illustrated in Figure 3.10.

Figure 3.10 'Simplest' oscillator
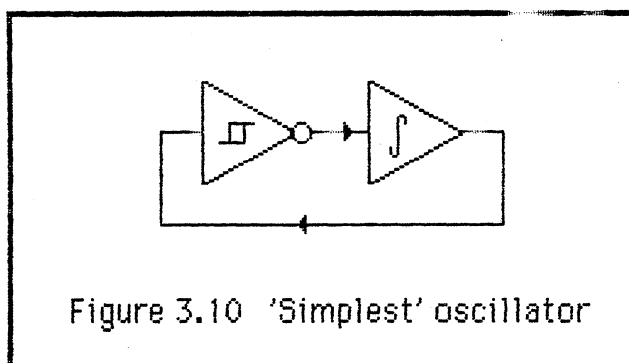
The 'integrator' element is quite straight forward: the output is the integral with respect to time of the input. The 'hysteresis' element is a circuit known as an *inverting Schmitt trigger*. The output of the Schmitt trigger depends not only on the present state of the input, but also on the last state of the output. The Schmitt trigger is a circuit with a precisely defined hysteresis. It is a digital device in the sense that the output of the device is discrete and can take on only one of two values either low or high, 0 or 1, minus or plus, etc. As the name "trigger" might imply, the output state changes when the input voltage level reaches a certain value known as a switching potential. The hysteresis of a Schmitt trigger is dependent on the fact that the switching potential is a function of the output state of the circuit. The behavior of this element can be presented as a graph (Figure 3.11)

Figure 3.11 Input/Output Diagram
of the Inverting Schmitt Trigger
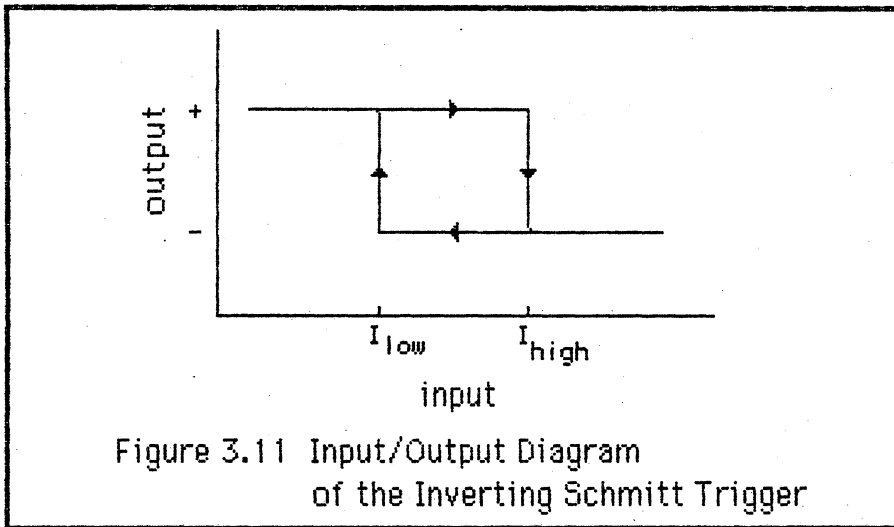
or in tabular form:

Previous Output	Input	Next Output
Low	$<I_{low}$	High
Low	$\geq I_{low}$	Low
High	$>I_{high}$	High
High	$\leq I_{high}$	Low

For the inverting Schmitt trigger, the switching potential (call it I_{high}) when the output state is high is higher than the switching potential (call it I_{low}) when the output state is low. Let's trace the operation of a Schmitt trigger over an input cycle. Starting at a low input voltage, the output will be high. As the input voltage is raised, the output will stay low as I_{low} is passed and at I_{high} it will switch to low. If the input is then lowered, the output will stay high as I_{high} is passed and only switch when I_{low} is reached. The difference between I_{low} and I_{high} is the hysteresis.

In the oscillator of Figure 3.9, there are only two variables: the

output of the integrator, and the output of the hysteresis element. Figure 3.12 is plot of these two variables.

Figure 3.12 Outputs of 'simplest' oscillator
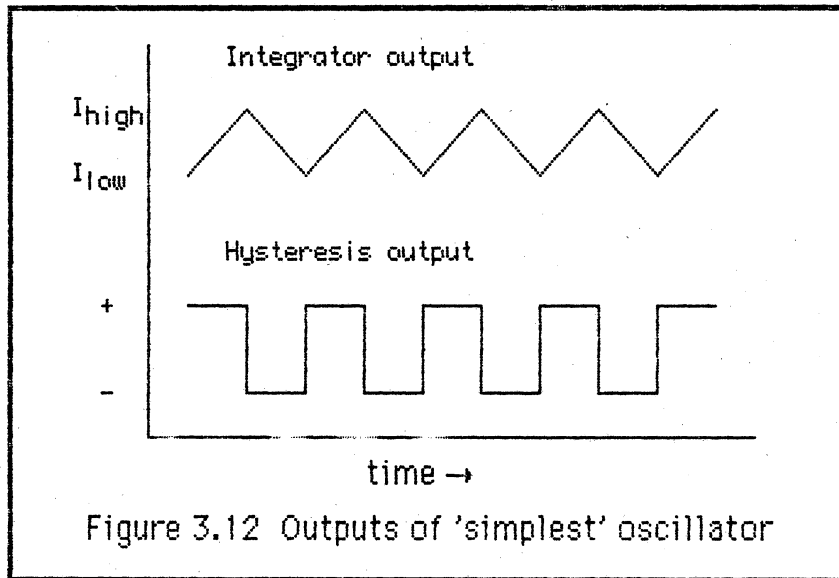
Note how the output of the integrator affects itself:

1) The integrator's output enters the hysteresis element.

2) The history of the hysteresis input determines its output.

3) The hysteresis output enters the integrator.

4) The input of the integrator affects the output of the integrator.

5) Thus, overall, the integrator's input is also a function of its output.

One might ask what elements other than integrators might function in this circuit to allow it to oscillate? Any element which has a 'transient response' can be used in this oscillator. Transient response can be

/

defined in terms of the functional relationship between input and output:

$$\frac{\partial \; Output}{\partial \; Input} = f(t)$$ [3.1]

A system has a transient response to an input if the partial derivative of the output with respect to that input has an explicit dependence on time.

If there is a functional dependence on time and there are different responses to different levels of stimulus, then the system can be made to oscillate.

If a chemical system has a transient response, then one can use a straight forward design technique to produce an oscillator. Choose two levels of stimulus or input to the chemical system that produce different, possibly opposite, responses. One can label the two levels of stimulus as low and high, and the resulting response of the chemical system as low and high. For example the high response is the response due to the high stimulus. Figure 3.13 schematically illustrates such a system.

Figure 3.13 Chemical System with Transient Response
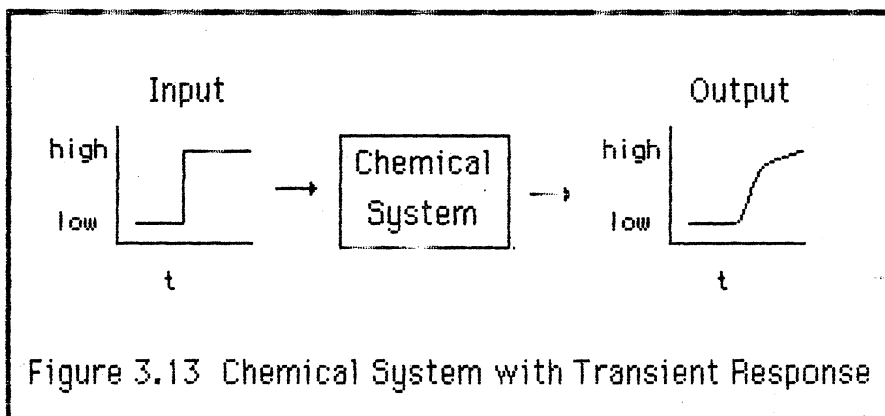
The output of the system, the response, is then used as an input to an

inverting function with hysteresis. And, the output of the hysteresis element is connected to the input of the chemical system. Figure 3.14 illustrates such a connection.

Figure 3.14 Combined Hysteresis and Chemical System
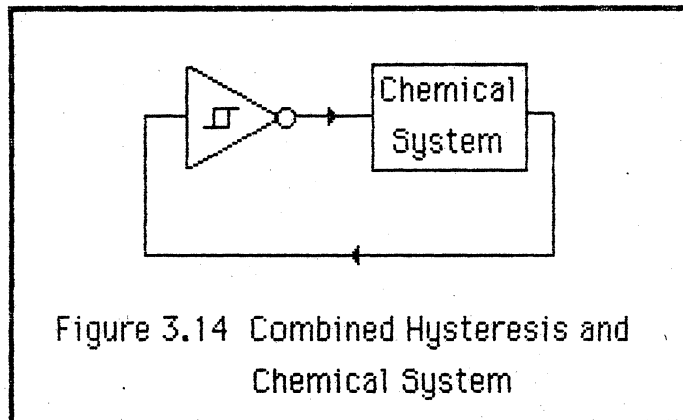
If the response is low, then the feedback will tend to force the stimulus to a high state, and vice versa. But the hysteresis controls the switching between the stimuli such that for the first part of the transient response to a low stimuli, the stimuli will remain low. Then, at some level of the low response, the function will change the stimulus to the high state. Thus there will be a periodic switching between alternate states as illustrated in Figure 3.15.
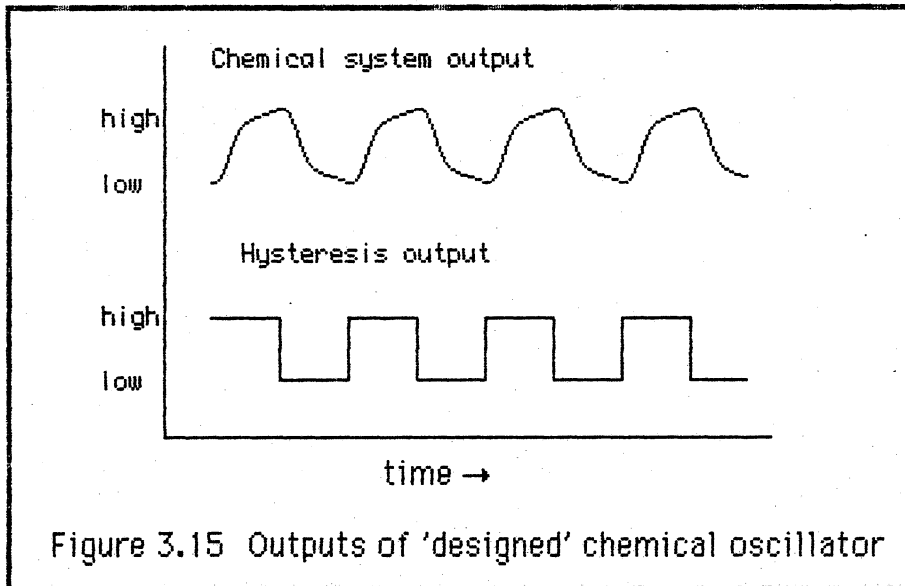
Figure 3.15 Outputs of 'designed' chemical oscillator

The nature of the hysteresis circuit depends on the type of response of the chemical system. One can distinguish two different types of responses to the stimulus. With the first, the response tends towards a steady state that is a completely different value for each level of stimulus. One might call this a *cumulative* response. The second type of action is such that the stimulus perturbs the system from an "equilibrium" state and the time dependence is such that the system relaxes towards the same "equilibrium" state independent of the size or direction of the stimulus. One might call this a *resilient* response. As an example of these two different types of response, consider a vessel with a constant flow of water entering the top, and a valve at the bottom that lets water out (Figure 3.16). The stimulus to the system is the partial closing of the outlet valve.

Figure 3.16 Response of a Flow System to Stimulus
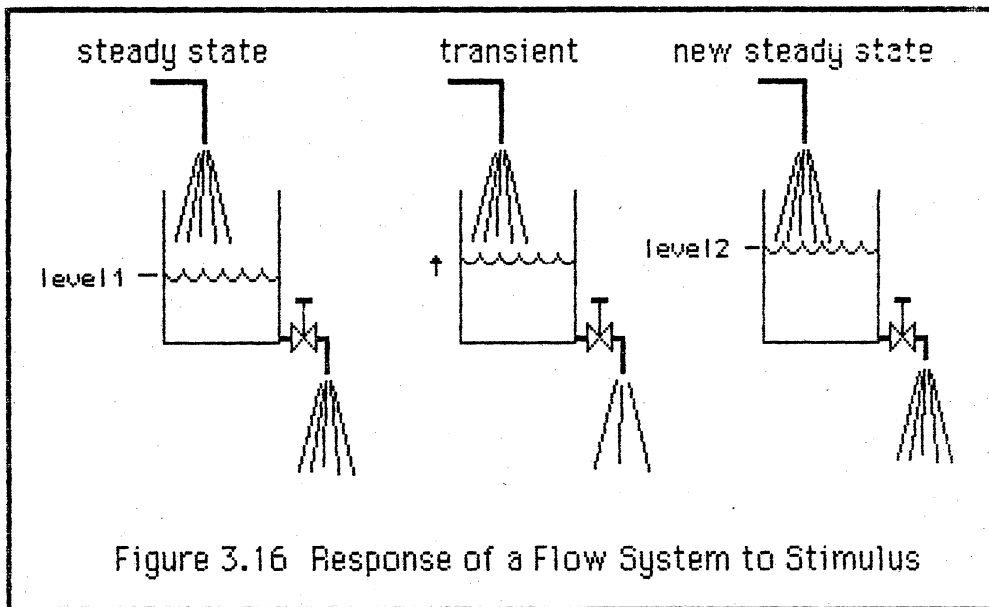
If one measures the <u>level</u> of the water in the vessel, then the level will rise to a new steady state level. This is an example of a response of the second type, a *cumulative* response (Figure 3.17).

Figure 3.17 Response of level to stimulus (cumulative)
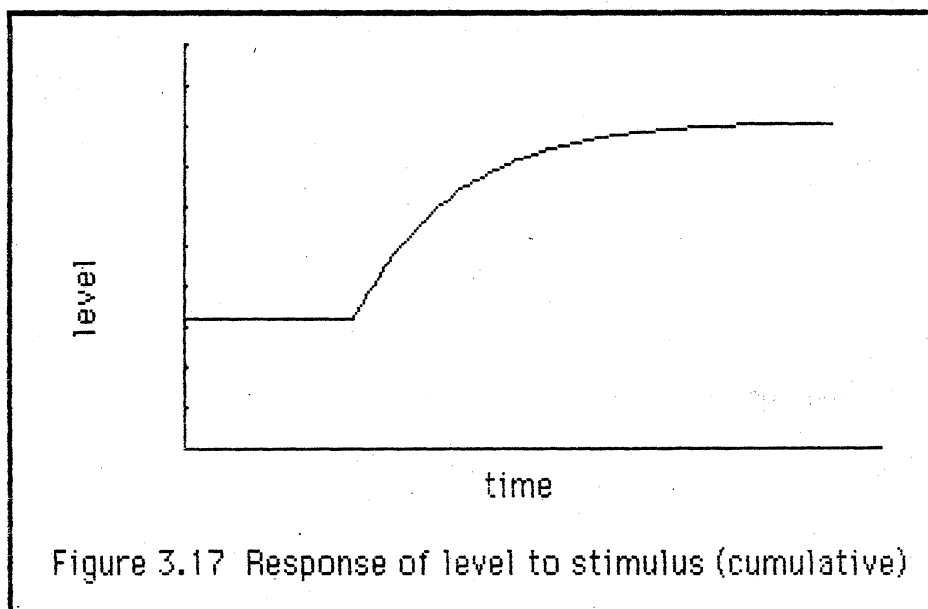
Given a cumulative response, what type of external feedback would allow such a system to oscillate? The integrator of Figure 3.10 is a cumulative responding system, and the appropriate feedback element for this system is a Schmitt trigger (Figure 3.11) as explained previously. As a cumulative response rises, the feedback element should sense the crossing of a threshold value (I_{high} in Figure 3.11) and switch the input to the system. The feedback element must then wait for the output of the cumulative system to reach a different threshold. The output of the cumulative system will periodically move <u>between</u> the two thresholds.

Returning to the system of Figure 3.16, if one measures the <u>flow</u> of water out of the vessel, then the flow will first decrease and then increase to match the incoming flow again. This is an example of a response of the first kind, a *resilient* response (Figure 3.17).

Figure 3.18 Response of flow to stimulus (resilient)
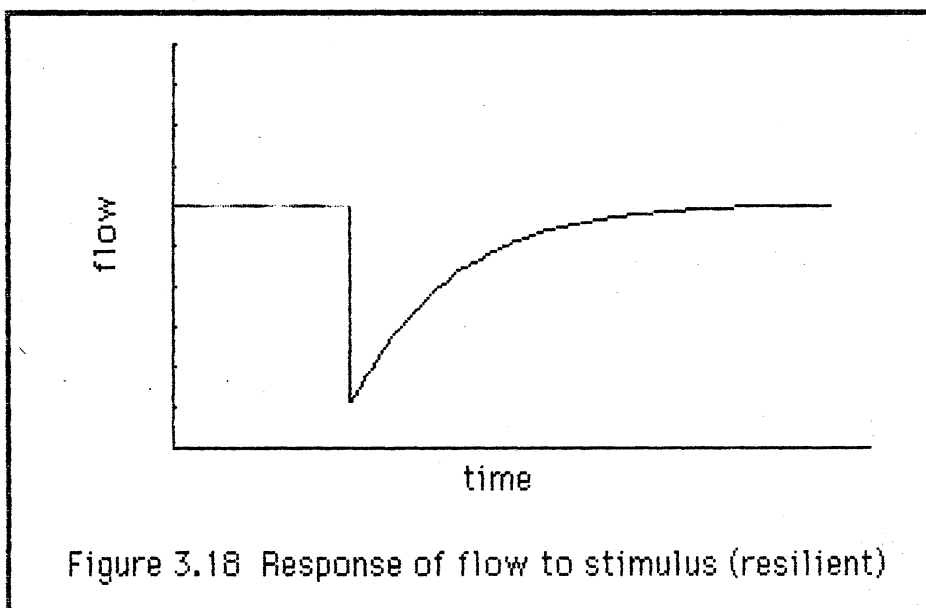
The type of external feedback, required to allow a resiliently responding system to oscillate, is different than the Schmitt trigger. The initial response is to jump to a new state and then the state will decay towards equilibrium. Before the state reaches equilibrium, the feedback element must switch the state. There must be a kind of "pre-action" response where the function measures the approach to equilibrium and switches the output at some fixed point before equilibrium is reached. This can be described graphically:

Figure 3.19 Input/Output Diagram
of the Inverting 'pre-action' Trigger
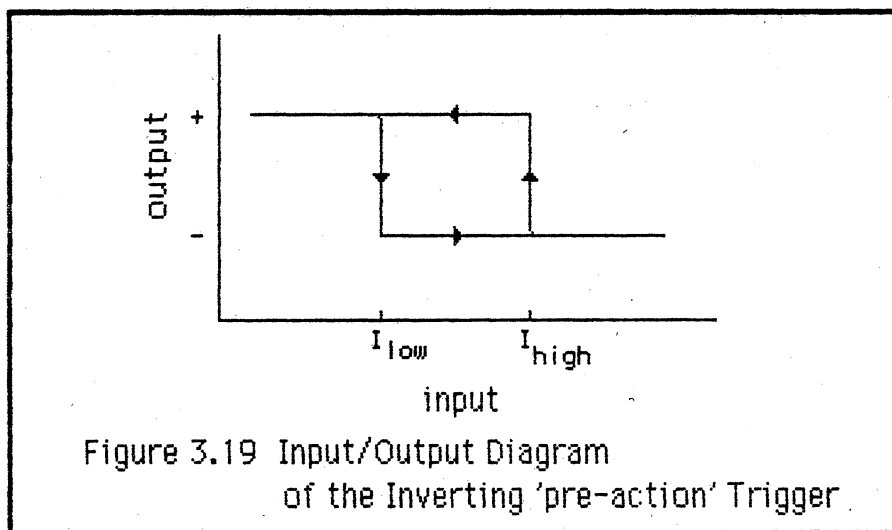
Or it may be described by a table:

Previous Output	Input transition	Next Output
Low	from $>I_{high}$ to $<I_{high}$	High
High	from $<I_{low}$ to $>I_{low}$	Low

This function may still be considered a hysteresis function since the output of the function during a change of input from low to high does not retrace the output resulting from a change of input from high to low. This

hysteresis function is a kind of "pre-action" function. As in the case of the Schmitt trigger previously described, one can follow a cycle of the input voltage. Starting at a low input voltage and a high output, as the input voltage is raised, the output will switch at I_{low} (before reaching I_{high}). The output will then stay low as I_{high} is passed. As the voltage is lowered, the output will switch at I_{high} and remain high as I_{low} is passed. If one follows a cycle of the resilient system oscillator: 1) the input to the resilient system will switch and after the initial discontinuity, the output of the resilient system will decay towards the equilibrium value, 2) at some point before reaching equilibrium, the output will reach a threshold and the external feedback will switch states and 3) the cycle will continue. The output of the resilient system will move outside of the thresholds on either side of equilibrium.

This "pre-action" hysteresis function is related to the Schmitt trigger, however it may not be physically realizable as an electronic circuit. For a *resilient* response, the 'pre-action' hysteresis is needed. For a *cumulative* response, the kind of hysteresis needed is exactly a Schmitt trigger.

If the type of pre-action hysteresis function required by a *resilient* response is not physically realizable, it may be possible to modify the behavior of the system. Judging from the example of the water flowing into a vessel (Figure 3.16), changing the variable measured or observed may change the type of response observed from *resilient* to *cumulative*.

Consider the use of an electrochemical system as a model system

for designing an oscillogenic instrument. What is the nature of its response? If an electrical potential (voltage) is applied to an electrochemical system, the current flowing in that system can be measured. This provides the input (potential) and the output (current) needed. The response of this system to a change in potential is of the *resilient* type (see chapters 4 and 5). One must then either devise a 'pre-action' hysteresis circuit, or observe some other quantity. A schematic view of the resulting oscillogenic system is shown in Figure 3.20 below.

Figure 3.20 Oscilliogenic instrument
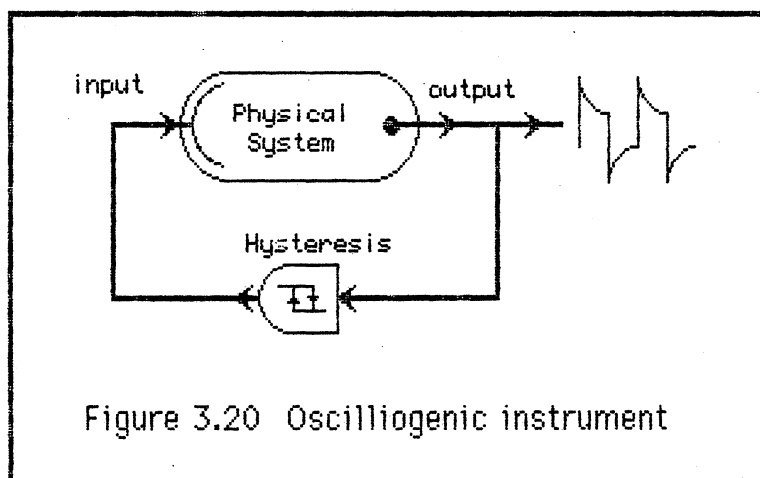
In summary, P. R. Rony and P. K. Mercure took a long standing interest in chemical oscillators and combined that interest with the fundamentals of engineering control theory. P. R. Rony realized that chemical systems can become oscillatory through the use of external feedback and a suitably designed controller. Electrochemistry was chosen as a model system because of the ease of interfacing the experiments to computers. P. K. Mercure realized that using linear feedback with an

electrochemical system would not be suitable, and that nonlinearities dominated the behavior of previously known chemical oscillators. This led P. K. Mercure to suggest the use of "hysteresis" elements (such as Schmitt triggers) in the feedback path. The qualitative properties of dynamic systems coupled to hysteresis elements can be used in a heuristic fashion to propose different feedback systems which might allow a chemical system to oscillate.

The general principles presented in this chapter could be studied with a mathematical model for an oscillogenic electrochemical system. The model would help to answer some fundamental questions about the system. First, how should the hysteresis function operate? How should the hysteresis function be implemented in hardware? Will the oscillator work? Then, can one predict the waveform? Finally, what is the dependence of the frequency on varying parameters?

Chapter 4.

A Mathematical Model for an Electrochemical System,
Numerical Preliminaries

"We shape first our tools, thereafter our tools shape us"

attributed to Buckminster Fuller

The obvious purpose of this model is to answer the following questions, which closed the previous chapter: (1) With a model electrochemical system, will the oscillator work? (2) Can one predict the frequency and waveform? and (3) What parameters affect the frequency and waveform, and to what extent? But there is further motivation in the choice of the method used. The waveshape and the frequency of the oscillator are of interest. Frequency is a single measurement and can only yield a single experimental quantity. Yet when the oscillator is connected to a computer, there can be many more measurements than just the frequency. How can one use these measurements to learn more about the system?

There are entire fields devoted to extracting information from signals. Two such fields are known as *system identification* theory and *parameter estimation* theory. System identification seeks to determine the <u>structure</u> of the dynamical system, while parameter estimation seeks to evaluate the <u>quantities</u> needed to predict the behavior of the system. In chemistry, a good example of these two areas can be seen in the elucidation of reaction mechanisms.

Reaction:

n-Pentane   iso-Pentane

$$A \longrightarrow B$$
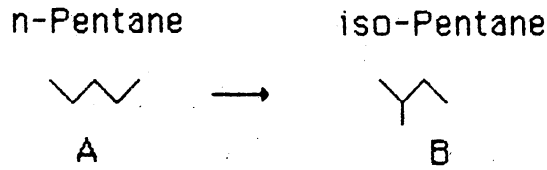
System identification:

$$r = \frac{k_1 P_A - k_2 P_B}{P_{H_2} + k_3 P_B} \quad \text{OR} \quad r = \frac{k_1 P_A - k_2 P_B}{P_{H_2} + k_3 P_A} \quad ?$$

Parameter Estimation:

$$r = \frac{1.8\, P_A - P_B}{P_{H_2} + 6.6\, P_B} \quad \text{OR} \quad r = \frac{1.8\, P_A - P_B}{P_{H_2} + 8.5\, P_B} \quad ?$$

Figure 4.1 System Identification and Parameter Estimation in Chemical Kinetics (from Froment 1979, Hosten 1971)

Figure 4.1, illustrates the issues for the case of chemical kinetics. Establishing the <u>form</u> of the rate expression is equivalent to system identification; evaluation of the rate <u>constants</u> is parameter estimation. Of course, the two processes go hand in hand in a kinetic study.

There are many approaches to parameter estimation, but the particular approaches of interest in this work are techniques for making real time estimates of chemical parameters. This "on line" measurement of chemical parameters would be appropriate in process instrumentation and in flowing systems such as chromatography. The type of estimation used in this case may be referred to as an "observer". One wants to observe a quantity for which one may not have a good or direct measurement. In essence, an "observer" compares the predicted results from an internal model of the system, to quantities that can be directly measured. If there are differences between predicted and measured values, then the parameters (or constants) in the internal model are adjusted.

The subject of parameter estimation will be discussed further in a later chapter. At this point, note the fact that the observer must have an internal model of the nature of the observed system, that is a description of the dynamics of the system being measured. For practical purposes, this dynamic model must be simple and capable of being rapidly evaluated. Since a computer is doing the evaluation, this first requirement translates to a program that takes a short amount of execution time to calculate the changes in the system. Moreover, the application to instrumentation requires a small computer; thus the implementation of the program should

be suitable for microcomputers. This second requirement means that the size of the program and the required data storage should be minimized. The mathematical model of the electrochemical cell was developed with an eye towards embedding it within an observer program.

The simplest model of an electrochemical system is:

$$A + e^- \rightarrow B \qquad\qquad [4.1]$$

An electron, e^-, driven by an electrical potential combines with component A at the surface of an electrode to produce component B (see Figure 4.2).

Figure 4.2 Schematic View of Electrochemistry
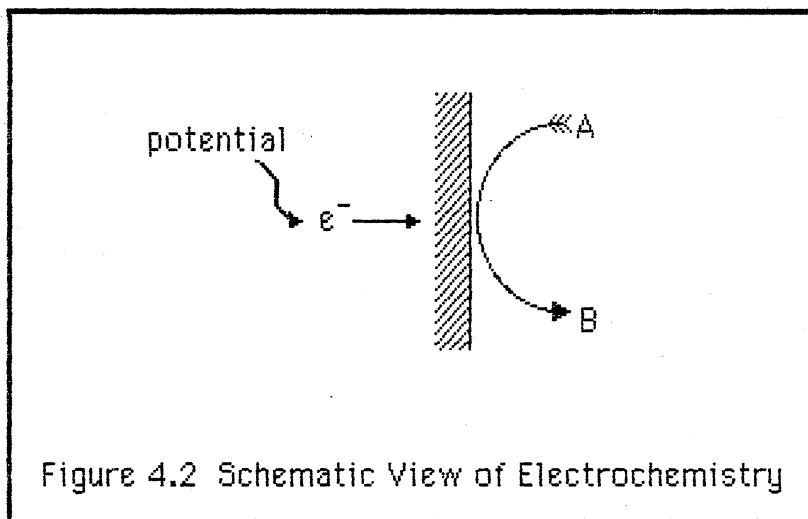
The components, A and B, may diffuse to and from the electrode surface. For dilute electroactive components in an excess of supporting electrolyte [Kelzer and Scherson 1980], the governing transport equation [Bird, Stewart, and Lightfoot 1960] is:

$$\frac{\partial c_A}{\partial t} + v \cdot \nabla c_A = \mathcal{D}_{Am} \nabla^2 c_A + R_A \qquad [4.2]$$

where,

c_A is the concentration of species A (moles/volume)

v is the velocity of convection (distance/time)

∇ is the gradient operator, in this case, of a scalar field, (effectively the first derivative in space of a quantity

$\mathcal{D}_{Am}$ is the diffusivity of species A in the mixture m (distance^2/time)

∇^2 is the Laplacian operator, in this case of a scalar field (effectively the second derivative in space of a quantity)

R_A is the rate of production of A due to chemical reaction (moles/(volume*time))

If the fluid is stagnant (that is, not a rotating disk electrode or other moving electrode, and there is no flow in the system) and there is no reaction in the bulk phase then:

$$v = R_A = 0 \qquad [4.3]$$

and,

$$\frac{\partial c_A}{\partial t} = \mathcal{D}_{Am} \nabla^2 c_A \qquad [4.4]$$

When the electrode is a flat plate, or when the boundary layer in which the concentration differs appreciably from the bulk concentration is small, then the equation can be stated in a single rectangular coordinate:

$$\frac{\partial C_A}{\partial t} = \mathcal{D}_{Am} \frac{\partial^2 C_A}{\partial x^2}$$

[4.5]

Equation 4.5 describes the concentration gradients in the electrolyte solution. The reaction occurring at the surface of the electrode sets a boundary condition for the differential equation. This model accounts only for the current flowing to the electrode as a result of the reaction at the surface; electrochemists refer to this current as the Faradaic component of the total current. There is also a current flow due to redistribution of charges at the surface of the electrode, known by the electrochemists as double-layer charging. If double layer effects were to be incorporated into the model, then some expression for the rate of charging would have to be assumed. This effect could be incorporated into the model, but it will be assumed negligible as electrochemists often do for a first approximation [Brown 1984, Kelzer 1980, Macdonald 1977]. The reaction is assumed to be a first order process consistent with electrochemical practice [Brown 1984, Macdonald 1977]. Adsorption of species on the surface of the electrode is not incorporated into the model, but if required it would be a simple matter to do so. The final result is that the current flowing to the electrode is related to the net rate of reaction at the surface:

$$i = nFA (k_f C_a - k_b C_b)$$

[4.6]

where, i is the current to the electrode (coulombs/time)

A is the area of the electrode (distance^2)

n is the number of electrons exchanged for each ion reacting

F is Faraday's constant (coulombs per mole of electrons)

k_f is the forward rate constant (distance/time)

k_b is the backward rate constant (distance/time)

Activated complex theory predicts that for a simple charge transfer process [Macdonald 1977] the forward and reverse rate constants are:

$$k_f = k_0 \exp\left[-\beta \frac{nF}{RT} (E-E_0) \right] \qquad [4.7]$$

$$k_b = k_0 \exp\left[(1-\beta) \frac{nF}{RT} (E-E_0) \right] \qquad [4.8]$$

where k_0 is a standard rate constant (distance/time)

β is the charge transfer coefficient (dimensionless)

R is the gas constant (electron-volts per mole-Kelvin)

T is the absolute temperature (Kelvin)

and E_0 is the standard potential of the system (volts)

The flux of ions at the electrode surface is related to the current flowing to the electrode such that:

$$i = nFA \, \mathcal{D}_{Am} \left. \frac{\partial C_A}{\partial x} \right|_{x=0} \qquad [4.9]$$

The expression for the current can also be written in terms of the concentration of the product:

$$= -nFA \, \mathcal{D}_{Bm} \left. \frac{\partial C_B}{\partial x} \right|_{x=0} \qquad [4.10]$$

This gives one boundary condition for the differential equation. The other boundary condition is obtained by noting that at large distances from the electrode, the concentrations must approach the bulk concentrations:

$$C_A(x \to \infty) = C_{A,bulk} \qquad [4.11]$$

$$C_B(x \to \infty) = C_{B,bulk} \qquad [4.12]$$

At this point, one can assume that the diffusivities of the species are roughly equivalent:

$$\mathcal{D}_{Am} = \mathcal{D}_{Bm} \qquad [4.13]$$

In addition, if the second component, B, is present in negligible quantities at the beginning of the experiment, then at any point:

$$C_B = C_{A,bulk} - C_A \qquad [4.14]$$

One can define dimensionless quantities in the following way:

$$f = C_A / C_{A,bulk} \qquad [4.15]$$

$$\tau = t\, k_0^2 \,/\, \mathcal{D}_{Am} \qquad [4.16]$$

$$d = x\, k_0 \,/\, \mathcal{D}_{Am} \qquad [4.17]$$

With these substitutions, the dimensionless model equation becomes:

$$\frac{\partial f}{\partial \tau} = \frac{\partial^2 f}{\partial d^2} \qquad [4.18]$$

The initial condition is:

at $\tau=0$, and $d>0$,

$$f = 1 \qquad\qquad [4.19]$$

The first boundary condition is:

at $\tau>0$, $d \to \infty$,

$$f = 1 \qquad\qquad [4.20]$$

And the second boundary condition is:

at $\tau>0$, $d=0$,

$$\frac{\partial f}{\partial d} = R(f,E) \qquad\qquad [4.21]$$

where $R(f,E) = f \exp\left[-\beta \frac{nF}{RT} (E-E_0) \right]$

$$- (1-f) \exp\left[(1-\beta) \frac{nF}{RT} (E-E_0) \right] \qquad\qquad [4.22]$$

The most interesting behavior is due to the kinetic expression that forms the second boundary condition.

When the applied potential (E) is time varying, there are virtually no closed form analytical solutions. There are classic series solutions [Nicholson and Shain 1964 and 1965] and solutions by the use of the Laplace transform [Macdonald 1977]. However, for any realistic case, at some point in the evaluation of a solution, a numerical method is required. For the two solutions just mentioned, the numerical step is typically the evaluation of an integral. There might also be approximate perturbation

solutions for the cases where the potential input function is periodic [Gilsinn 1982, Jaffe 1984]. However, perturbation methods are so complex that the derivation is extremely error prone, and the application of perturbation methods to partial differential equations is not a very well developed technique. Since the customary solution methods require the use of numerical methods, it is appropriate to consider which are the most efficient numerical methods. This work uses more recent advances in numerical analysis [Davis 1984, de Boor 1978] to generate a compact numerical method.

The solution to the differential equation is a continuous function of both time and distance (see Figure 4.3). A numerical technique is a method in which the solution of a problem is expressd in an approximate manner suitable for implementation on a digital computer. Since the digital computer is a discrete device, there must be some discretization in both time and space. That is, the solution must be represented as a finite set of numbers within the computer. The technique employed in this work is known as the method of lines [Davis 1984, Finlayson 1980]. A method is chosen for describing the spatial solution at any one time; then, the time variation of this description is determined. This technique results in a series of coupled, first order (in time), ordinary differential equations. The solution to these equations is then propagated in time. The spatial discretization can be thought of as establishing some points in space, and the temporal propagation of the solution can be thought of as following the solution at these points as a function of time.
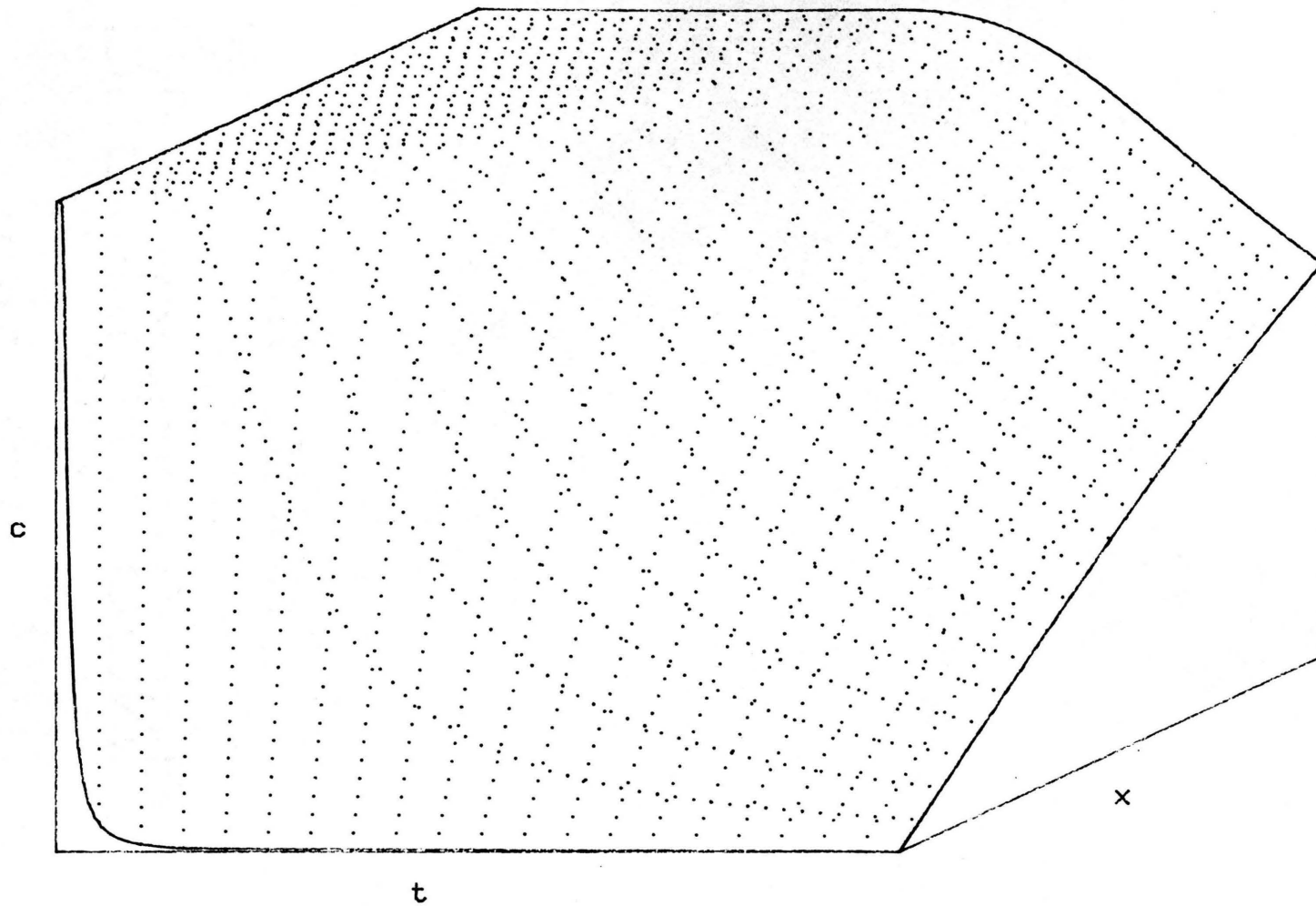
Figure 4.3 Solution to a Differential Equation
Sketched as a Surface Continuous in Time and Space
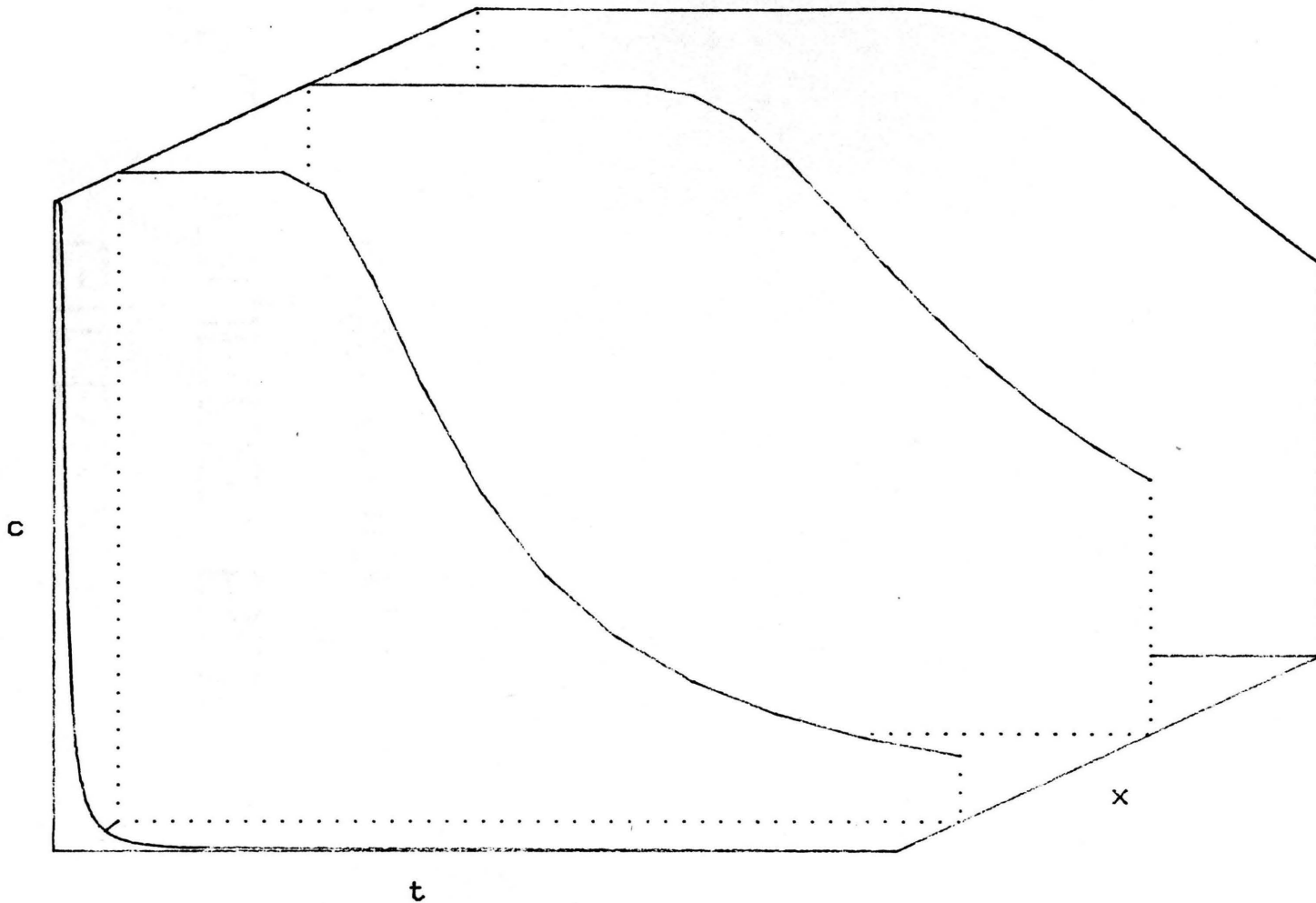
Figure 4.4   Method of Lines, Approximate Solution
of a Differential Equation

As illustrated in Figure 4.4, the temporal solutions at each spatial point are like lines in time.

The simplest spatial discretization is obtained by dividing the spatial dimension into equal sized segments (see Figure 4.5).

Figure 4.5 Uniform Spatial Discretization
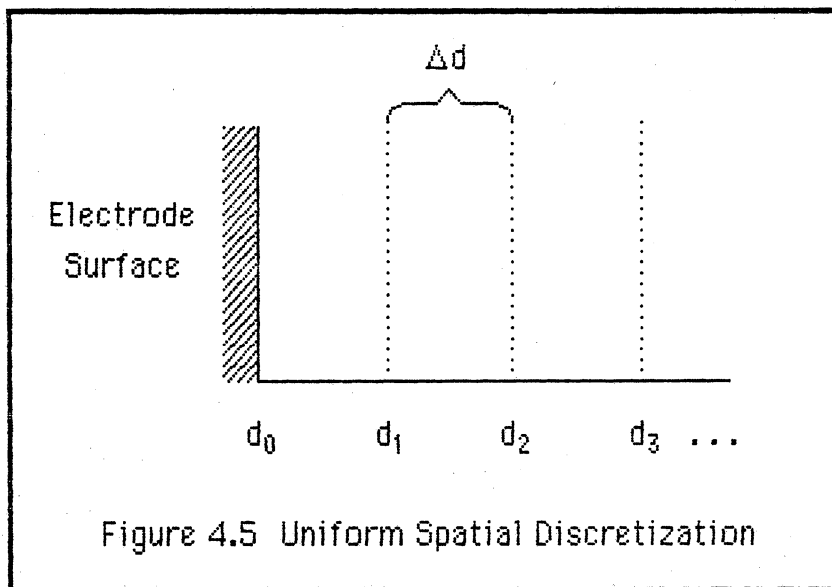
One then has a vector of points in space:

$$\mathbf{d} = [\ d_0\ ,\ d_1\ ...\ d_i\ ...\ d_n\ ] \qquad [4.23]$$

$$\text{let}\quad h = d_i - d_{i-1} \qquad [4.24]$$

Note: Vector and matrix quantities are indicated in this manuscript by the use of **bold** face.

The value, h, is the spatial step-size. Intuitively, if one wants to calculate a derivative numerically, one usually resorts to the difference

formulas that are used in introductory calculus. An example of a difference formula is:

$$\frac{\partial f(\tau, d_i)}{\partial d} \approx \frac{f(\tau, d_i) - f(\tau, d_{i-1})}{h} \qquad [4.25]$$

This intuition can be placed on a more precise basis by examining a Taylor series expansion:

$$f(\tau, d_{i-1}) = f(\tau, d_i) - h \left. \frac{\partial f(\tau, d)}{\partial d} \right|_{d=d_i} + \frac{h^2}{2} \left. \frac{\partial^2 f(\tau, d)}{\partial d^2} \right|_{d=d_i} + \ldots \quad [4.26]$$

Which can be rearranged to:

$$h \left. \frac{\partial f(\tau, d)}{\partial d} \right|_{d=d_i} = f(\tau, d_i) - f(\tau, d_{i-1}) + \mathcal{O}(h^2) \qquad [4.27]$$

where, $\mathcal{O}(h^{p+1})$ indicates truncation errors of order p

Also,

$$\left. \frac{\partial f(\tau, d)}{\partial d} \right|_{d=d_i} = \frac{f(\tau, d_i) - f(\tau, d_{i-1})}{h} + \mathcal{O}(h^1) \qquad [4.28]$$

Thus the *truncation error* varies linearly with the step size. A better expression would be of larger order, and one can be derived from the Taylor series expansion in the same way:

$$f(\tau, d_{i-1}) = f(\tau, d_i) - h \frac{\partial f(\tau, d_i)}{\partial d} + \frac{h^2}{2} \frac{\partial^2 f(\tau, d_i)}{\partial d^2} - \frac{h^3}{3!} \frac{\partial^3 f(\tau, d_i)}{\partial d^3} + \ldots [4.29]$$

$$f(\tau, d_{i+1}) = f(\tau, d_i) + h \frac{\partial f(\tau, d_i)}{\partial d} + \frac{h^2}{2} \frac{\partial^2 f(\tau, d_i)}{\partial d^2} + \frac{h^3}{3!} \frac{\partial^3 f(\tau, d_i)}{\partial d^3} + \dots \quad [4.30]$$

Taking the difference between these two expressions gives:

$$\frac{\partial f(\tau, d_i)}{\partial d} = \frac{f(\tau, d_{i+1}) - f(\tau, d_{i-1})}{2h} + \mathcal{O}(h^2) \quad [4.31]$$

Here the *truncation error* due to the spatial discretization varies as the square of the step-size. Similarly, expressions for the second spatial derivative and the first temporal derivative can be developed:

$$\frac{\partial^2 f(\tau, d_i)}{\partial d^2} = \frac{f(\tau, d_{i+1}) - 2f(\tau, d_i) + f(\tau, d_{i-1})}{h^2} + \mathcal{O}(h^2) \quad [4.32]$$

and

$$\frac{\partial f(\tau_j, d)}{\partial \tau} = \frac{f(\tau_j, d) - f(\tau_{j-1}, d)}{\Delta \tau} + \mathcal{O}(h^1) \quad [4.33]$$

where $\Delta \tau$ is a temporal step size

This type of discretization is known generically as *finite differences*. In electrochemistry, this method was popularized by the work of Feldberg [Feldberg 1969, 1981] and is still used today [Brown et al 1981, Seeber and Stepfani 1981, Magno 1983].

The solution is a vector:

$$f(\tau, d) \approx \mathbf{f}(\tau_j, \mathbf{d})$$

$$\approx [f(\tau_j, d_0), f(\tau_j, d_1) \dots f(\tau_j, d_i) \dots f(\tau_j, d_n)] \qquad [4.34]$$

$$\text{where} \quad \mathbf{d} = [d_0, d_1 \dots d_i \dots d_n] \qquad [4.35]$$

Using finite differences, the continuous solution to the differential equation is approximated at a series of points in space and at a series of points in time.

Instead of just a series of points, one might use polynomial functions to approximate the solution of the differential equation. The solution of the differential equation is based on linear combinations of approximation functions. These functions can be called *basis* functions. The use of approximation functions has the advantage of giving a solution to the problem at any point rather than just at a series of discrete points. Polynomials are also simple in form and easy to evaluate numerically. If the approximating functions are single polynomials each covering the entire spatial region of interest (*global* polynomials), then one of the resulting methods is known as "orthogonal collocation". This was introduced by Villadsen and Stewart [Villadsen1967] and first used in electrochemistry by Whiting and Carr [Whiting1977].

Note the use of the word "approximated". The continuous function, which is the solution to the differential equation, is "approximated" by another function. The problem of expressing the spatial concentration profile is related to problems in function approximation. Recent results in approximation theory [de Boor 1978] suggest better means of approximating the solution. A powerful technique, in current use, is the

use of an *interpolating* function [de Boor 1978]. Interpolating functions
are continuous functions that satisfy constraints <u>exactly</u> at certain
points. If the functions must match certain values at specific points,
then the functions are said to *interpolate* these points, that is, to give
function values at any position between the points. Another constraint
can be the *degree of continuity*. An informal definition will suffice
here: the *degree of continuity* is the number of the highest derivative
such that it and all lower derivatives are equal to each other at every
point along a function. Figure 4.6 shows examples of different degrees of
continuity; each quadrant shows a function with a possible discontinuity
at the midpoint. The degree of continuity is described mathematically [de
Boor 1978] by the notation: C0, C1, C2, etc. In Figure 4.6, the upper left
graph shows no continuity: the function *jumps* at the midpoint. The
upper right graph shows C0 continuity: the function does not jump in
value but <u>does</u> jump in slope. The lower left graph shows C1 continuity:
the function does not jump in value or first derivative at any point.
Finally, the lower right graph of Figure 4.6 shows C2 continuity: the
function does not jump in value, first, or second derivative at any point.
Note that it is difficult to visually discern the difference between C1 and
C2 continuity; this may be taken as a sign that C1 continuity is all that
is required for a 'reasonable' approximation to a function. One final
characteristic is: if the constraints on the approximating function include
matching both the value of a function <u>and</u> one or more derivatives of the
function at specific points, then the resulting function is an *osculatory
interpolation* function ('osculari' means 'to kiss' in Latin).

no continuity

C0, continuity of
the function value

C1, continuity of
1st derivative

C2, continuity of
2nd derivative
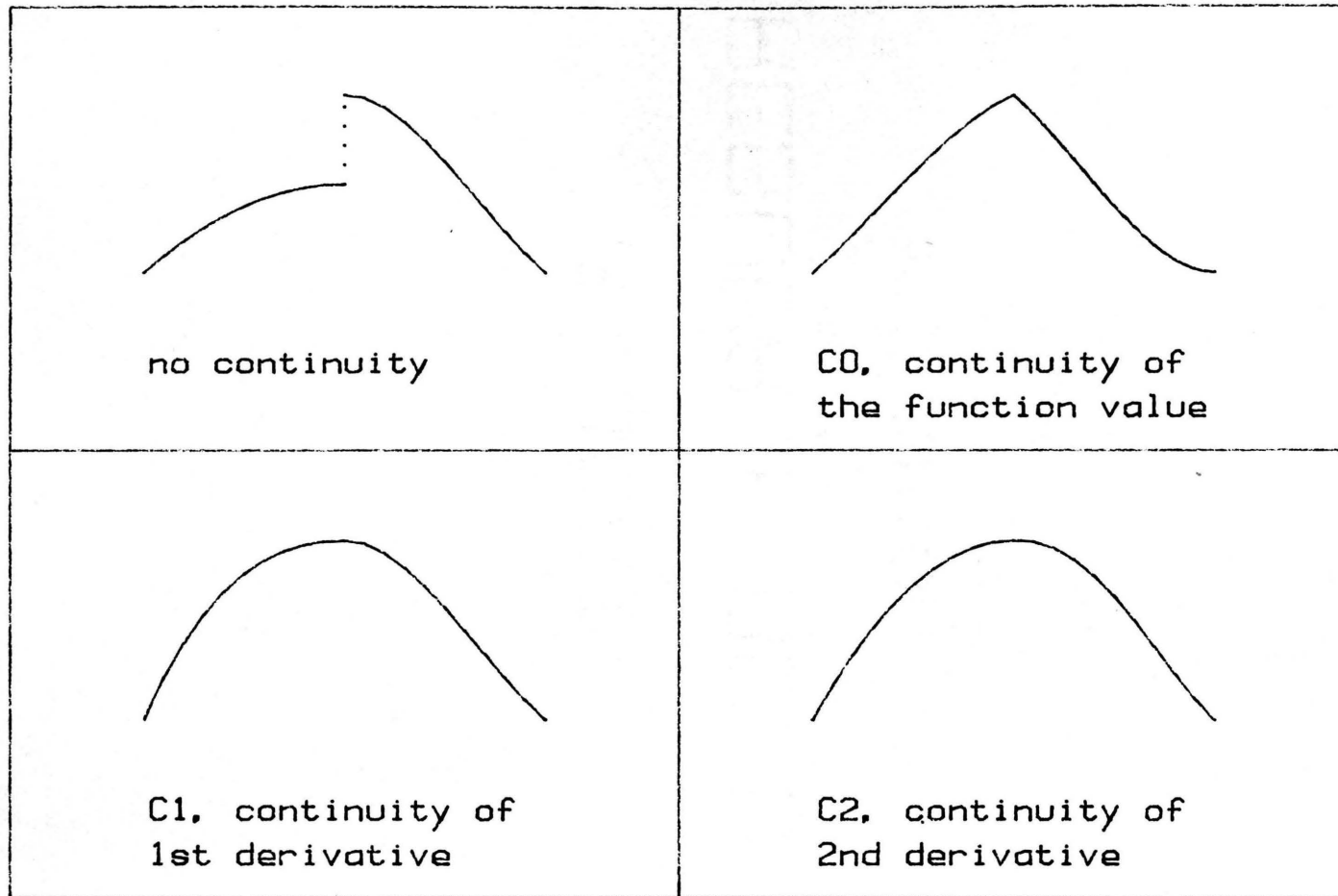
Figure 4.6 Illustration of Four Different Types of Continuity.

One criterion for choosing an interpolating function is that the resulting representation of the solution should be "smooth". If there are no "bumps" in the true solution or the actual physical situation, then there should be no "bumps" in the approximate solution. Anyone who has tried to evaluate a series solution will recall that there are many such "bumps" in the resulting functions. For example, summing only the first few terms of a Fourier series solution to a heat conduction problem will result in a profile of temperature versus distance that "wiggles" up and down substantially! This behavior is shown in Figure 4.7, which compares the Fourier series solution (using the first five terms) to the "true" solution (twenty terms). Similarly, if one fits a high order polynomial to experimental data, the resulting polynomial often shows wild oscillations, particularly outside the interval of experimental measurements. This is illustrated in Figure 4.8. These two examples demonstrate that inappropriate approximation functions can add numerical artifacts to the solution, particularly an oscillation in the approximation error where the approximation "wiggles" above and below the "true" solution.

If one desires to decrease the size of the numerical artifact due to "wiggle", then a precise definition of "wiggle" is required before minimizing the magnitude of the "wiggle". When connecting dots to form a smooth curve, a draftsman may use a thin flexible strip and force the line to pass through certain points; this is known as a *draftsman's spline*.
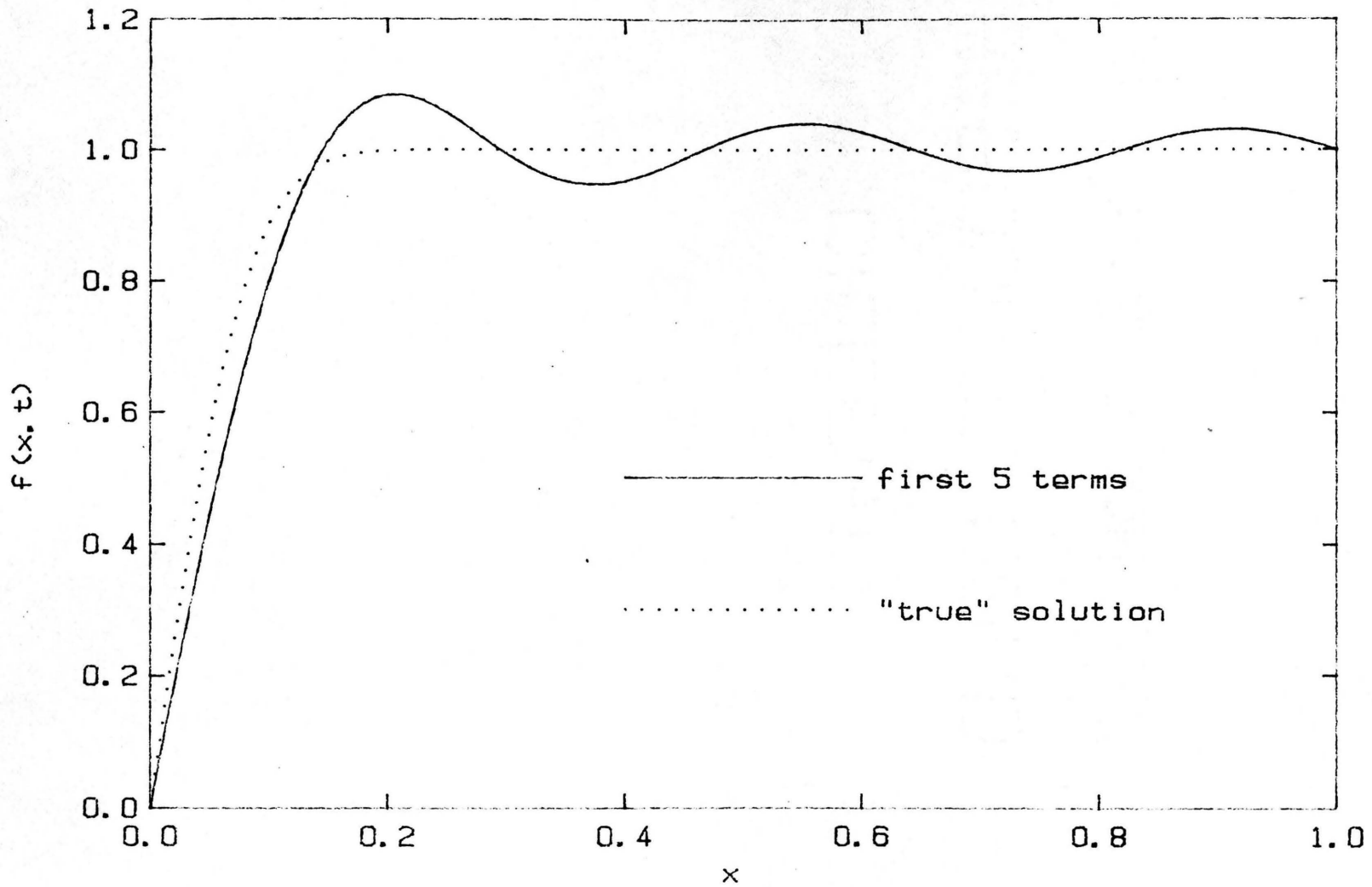
Figure 4.7 Fourier series solution to differential equation
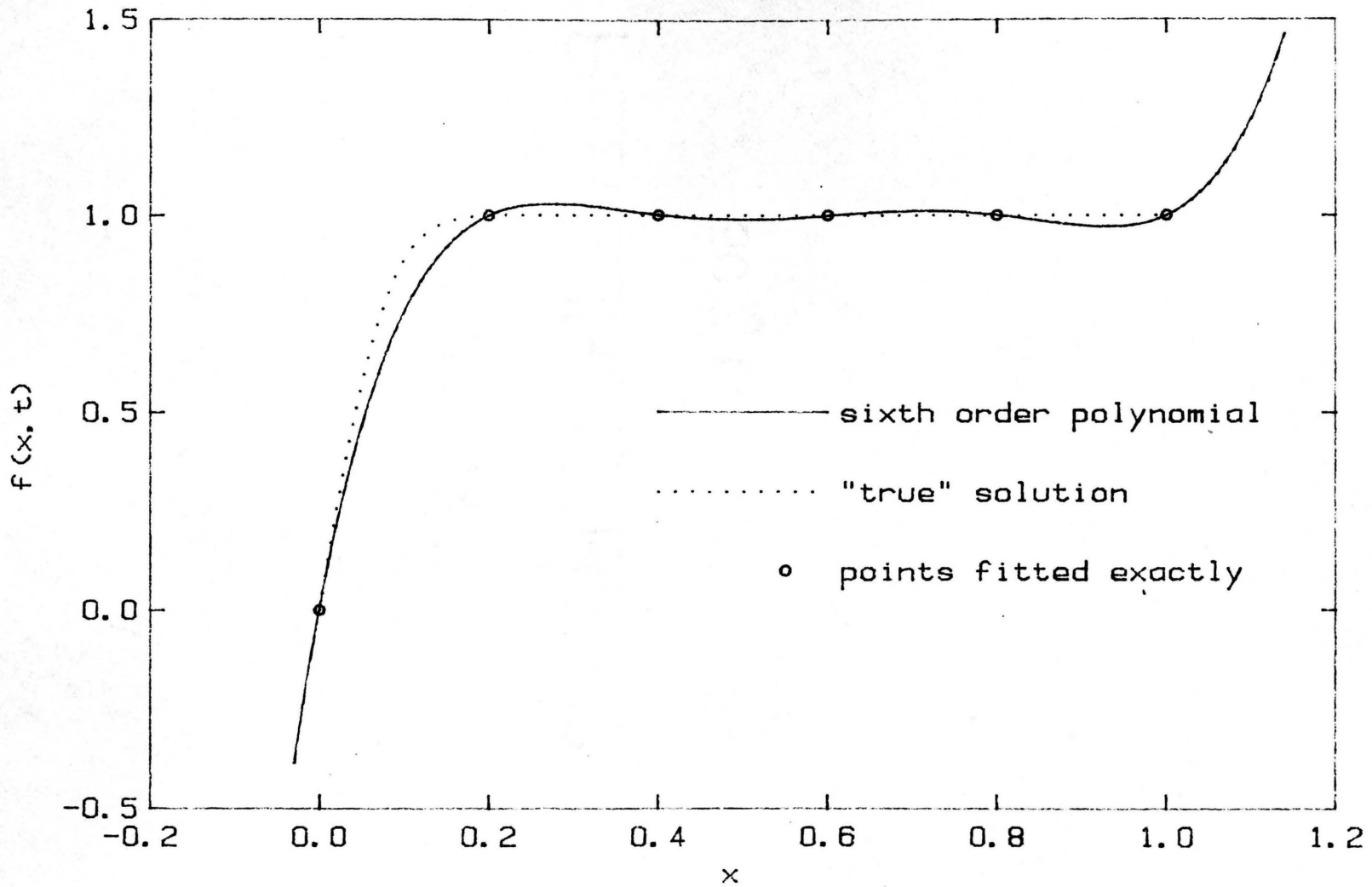
Figure 4.8 Polynomial fit to a series of points

Because the strip is a spring, its shape minimizes the strain energy [de Boor 1981]:

$$strain\ energy \approx \int_b^a \frac{[f''(x)]^2}{[\ 1 + (f'(x))^2\ ]^{5/2}}\ dx \qquad [4.36]$$

where

$$f''(x) = \frac{\partial^2 f(x)}{\partial x^2}$$

and

$$f'(x)) = \frac{\partial f(x)}{\partial x}$$

The strain energy can be approximated by leaving out the first derivative term. The resulting "wiggle" function can be represented as an integral over the space of interest:

$$wiggle(f) = \int_b^a \left[\frac{\partial^2 f(x)}{\partial x^2} \right]^2\ dx \qquad [4.37]$$

Here the square of the second derivative is used to measure the magnitude of the curvature only, and not its sign. Figure 4.9 shows two functions approximating a step function (the dotted line in the figure), along with the value of the "wiggle" as defined in equation 4.37. Certain functions can be found that minimize the "wiggle" over an interval, and thus approximately minimize the "strain energy" of the function. These functions are called *spline functions* [Schoenberg 1946].

'wiggle' = 1.00   'wiggle' = 0.25
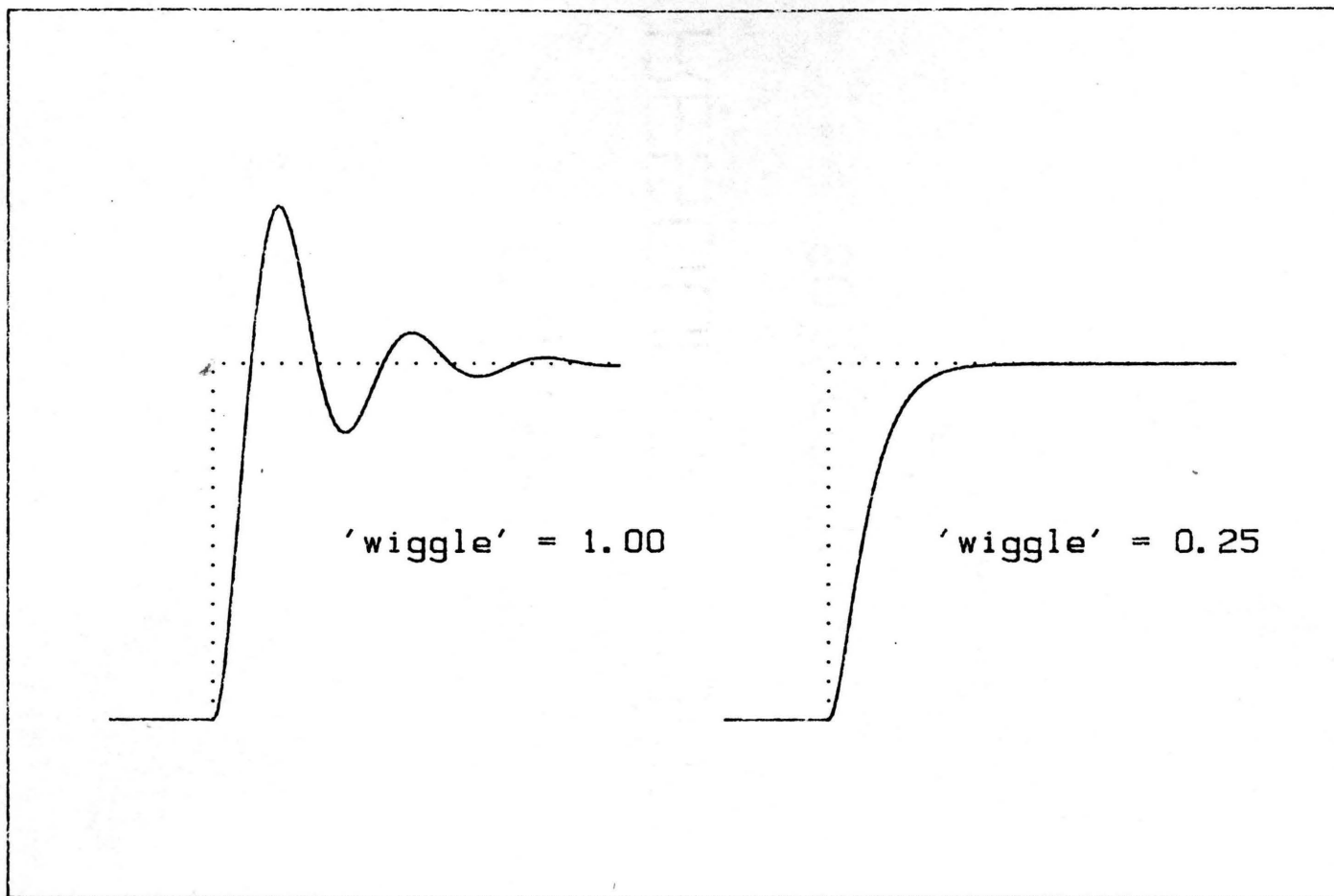
Figure 4.9   Examples of 'wiggle' evaluations.

These spline functions can be constructed of *piecewise polynomials* (the construction of such functions will be covered later in this chapter). A *piecewise polynomial function* is a function that is divided into pieces or intervals along a line, and the value of the function over each interval is given by a polynomial expression. A sketch of two piecewise polynomials is shown in Figure 4.10 (the dotted lines show the divisions for the intervals).

Figure 4.10 Piecewise Polynomials
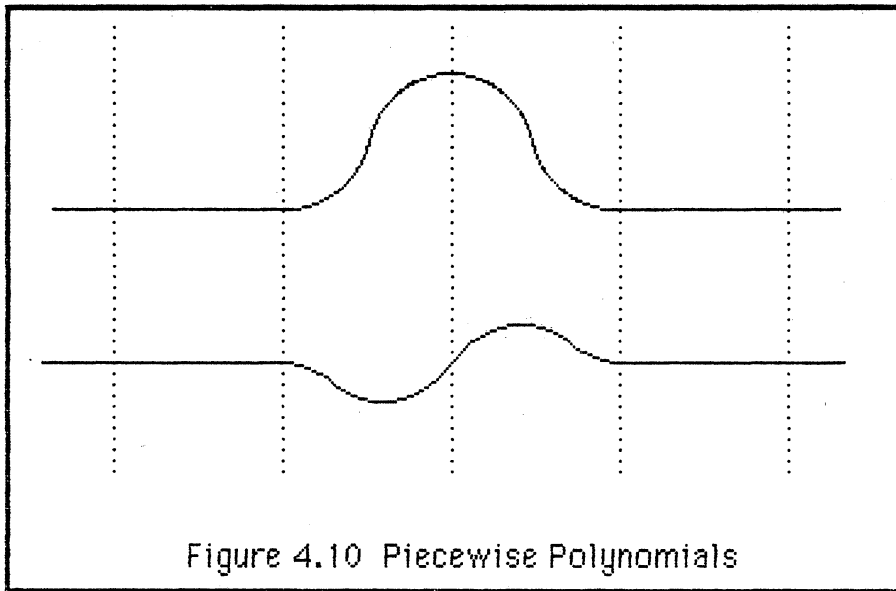
Much theoretical work has been done on approximation theory in recent years. In particular, the classic work by de Boor [de Boor 1978] develops the conditions for optimal interpolation functions using equation 4.37 as a measure of "wiggle". One can paraphrase theorem XIII.5 from de Boor in the following way:

One desires to interpolate a function, g, by using another function that exactly matches g and some of the derivatives of g at a series of points. There is a unique function that minimizes the measure:

$$\text{wiggle}(f) = \int_b^a \left[\frac{\partial^m f(x)}{\partial x^m} \right]^2 dx \qquad [4.38]$$

where m indicates the m th derivative

The theorem states that this optimal interpolating function is the "complete spline interpolant" to the function g. The order of this "smoothest" approximate is 2m, and the spline functions can be expressed as a piecewise polynomial of degree 2m-1.

A spline functions constructed of piecewise polynomials possess: (1) the computational simplicity of polynomials, (2) a minimal value for the numerical artifact of "wiggle", and (3) the ability to exactly match the solution function at specific points (osculatory interpolation).

Numerical techniques using spline functions for solving differential equations are extensively described by de Boor [1978], Dixon [1980], and Davis [1984]. The use of piecewise polynomials (but not splines as defined by de Boor) in electrochemistry was first discussed in a series of papers co-authored by S. Pons [Pons and Schmidt 1980, Pons et al 1982, Speiser et al 1982, Pons 1984]. In discussing the use of piecewise polynomials in electrochemistry, Pons and Schmidt [1980] suggested using "splines", but used a piecewise polynomial of order six over each interval.

The high order polynomial used does not fit the optimality condition that a true spline function would. This PhD dissertation is the first application of splines, as defined by mathematicians [Schoenburg 1946, de Boor 1978], to electrochemistry. Moreover, only one work with Pons [Speiser et al 1982] dealt with time-varying boundary conditions such as found in scanning voltammetry; Speiser et al used global (not piecewise) polynomials. Thus the present work is also the first use of spline collocation for electrochemical situations involving a time varying applied potential.

Note that the result of de Boor's theorem XIII.5 (see equation 4.38) is that ANY other interpolating function with m piecewise continuous derivatives will have more wiggle than a spline function. So, increasing the order of the approximating polynomial beyond that required for continuity increases the "wiggle" in the solution. This dissertation makes use of the optimal character of splines to improve the function approximation for the spatial discretization. This optimal function approximation property and the calculation economy provided are the basis of the choice of Hermite cubic splines for the spatial discretization employed here.

Before considering the choice of spline functions, there is one more class of approximating functions that can be considered. These are non-polynomial (typically transcendental) functions that are defined by a single expression over the entire spatial region. Proposed by Sørensen and Stewart [Sørensen 1982], the name "natural basis functions" was used to describe them. The functions are derived from analytical solutions of a

linear differential equation that is similar to that of the problem being solved.

Equation 4.39, taken from Sørensen [1982], defines a series of *natural basis functions*:

$$\varphi_j(x) = \frac{(1-x)^S}{2} \exp\left[\alpha_j\,(x-1) \right] + \frac{(1+x)^S}{2} \exp\left[-\alpha_j\,(x+1) \right] \quad [4.39]$$

where φ_j is the jth basis function
x is the spatial coordinate
α and s are constants determined for each j

Being non-polynomial, the functions are more difficult to evaluate, but they intrinsically satisfy some boundary conditions and require only a small number of functions to approximate the solution. This minimization of the number of describing functions also may minimize the amount of numerical computation needed to develop a solution to the problem. However, the initial derivation of the approximating functions is not simple and not easily automated on the computer. While minimizing the computational effort is important for parameter estimation, automated solution techniques may be more expedient. Therefore, in this work, polynomial expressions were considered optimal.

The physical basis of the problem described by equations 4.18 to 4.22 requires only continuity in the concentration and flux of ions in solution. That is, there must be no drastic jumps in either the concentration or the rate of motion of the ions because there is no physical reason for these jumps. One cannot say anything about the higher

derivatives on a physical basis. Thus, one seeks a basis function that will be optimal in some fashion with the constraint of zero and first derivative continuity. Since continuity is demanded for the function and its first derivative, the second derivative is the lowest derivative over which there is an arbitrary choice. The basis functions must provide C1 continuity (see Figure 4.6). This means that the value of m, in equation 4.38, has a value of 2 in this case. The second derivative is the measure of "wiggle" for this problem! If m=2, then the order of the interpolatory polynomial is 2m, that is, the order is 4 (the degree of the polynomial is 3). Thus the basis functions can be constructed of cubic polynomials. The actual functions used are one class of cubic polynomials known as the *Hermite cubic polynomials.*

The defining characteristics of piecewise polynomial basis functions are: (1) the function is non-zero only over a finite region, (2) this region is defined by values along the spatial axis called *breakpoints,* (3) within this region the defining function is a simple (in this case, cubic) polynomial, and (4) outside this region the value is identically zero. The Hermite cubic splines are comprised of two different piecewise polynomials; one piecewise polynomial is used to approximate the *value* of a solution function and the other is used to approximate the *slope*. Thinking in terms of approximating value and slope gives an intuitive feeling for the use of a spline function. The polynomials define the spline on an interval covering three breakpoints and are normalized in such a way that certain conditions always hold. For the *value function*, the defining polynomial evaluated at breakpoints one and three is equal to

zero; the defining polynomial evaluated at breakpoint two is equal to one; Figure 4.11 illustrates the value function, with the breakpoints indicated as circles. The first derivative of the value function evaluated at each of the three breakpoints is equal to zero. For the *slope function* (graphed in Figure 4.12 with the breakpoints marked with circles), the defining polynomial evaluated at each of the three breakpoints is equal to zero. The first derivative of the slope function evaluated at the first and third breakpoints is equal to zero. The first derivative of the slope function evaluated at the second, or middle, breakpoint is equal to zero. An algebraic representation of these functions is:

$$
\varphi_{2i-1}(x) = \begin{cases}
\dfrac{-2(x - b_{i-1})^3}{(h_{i-1})^3} + \dfrac{3(x - b_{i-1})^2}{(h_{i-1})^2} & \text{if } b_{i-1} \leq x < b_i \\[3ex]
1 + \dfrac{2(x - b_{i-1})^3}{(h_{i-1})^3} - \dfrac{3(x - b_{i-1})^2}{(h_{i-1})^2} & \text{if } b_i \leq x < b_{i+1} \\[3ex]
0 & \text{if } x < b_{i-1} \text{ or } b_{i+1} < x
\end{cases}
$$

[4.40]

and,

$$
\varphi_{2i}(x) = \begin{cases}
\dfrac{(x - b_{i-1})^2 \, (x - b_i)}{(h_{i-1})^2} & \text{if } b_{i-1} \leq x < b_i \\[3ex]
\dfrac{(x - b_i)(b_{i+1} - x)^2}{(h_i)^2} & \text{if } b_i \leq x < b_{i+1} \\[3ex]
0 & \text{if } x < b_{i-1} \text{ or } b_{i+1} < x
\end{cases}
$$

[4.41]
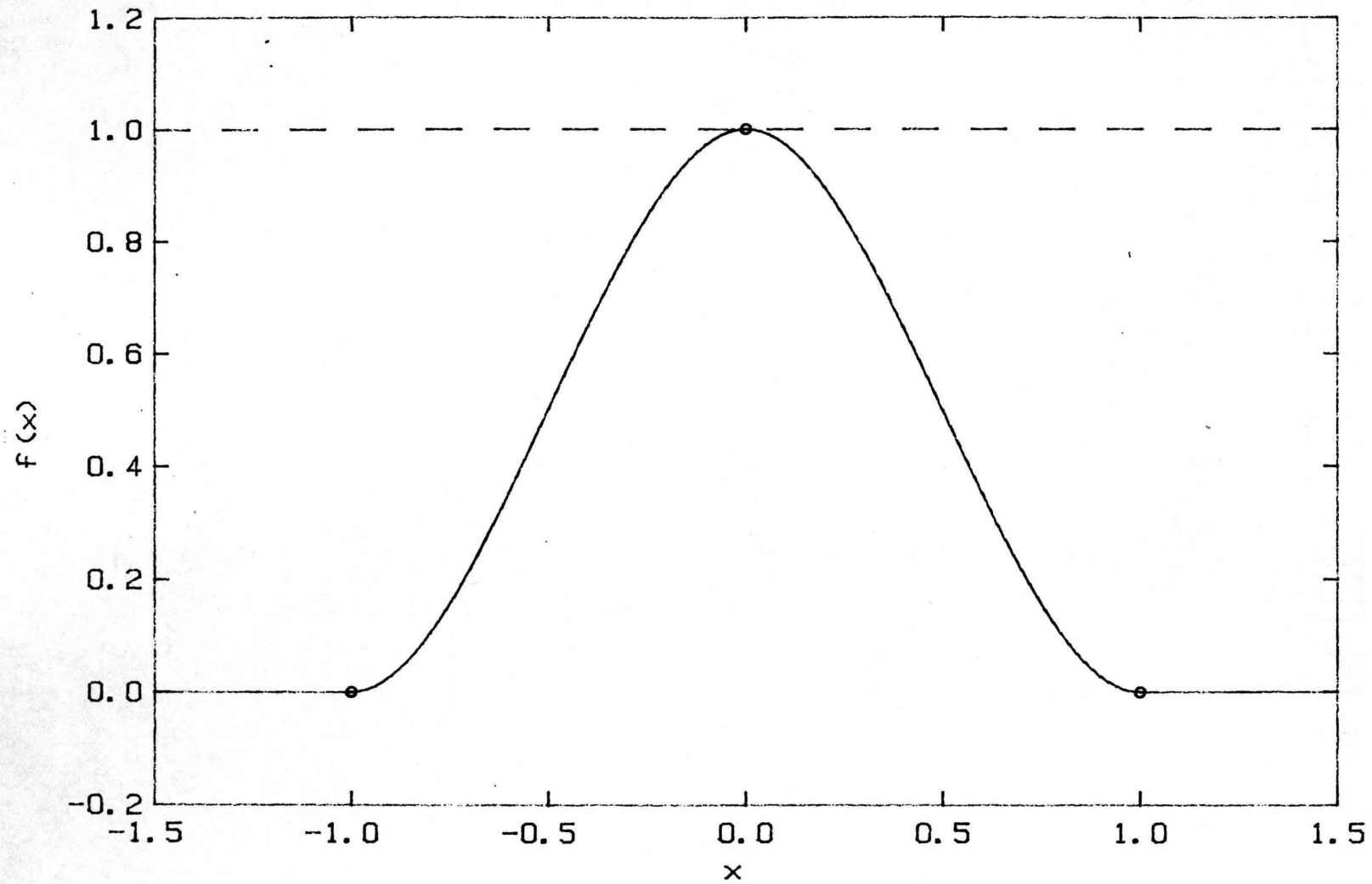
Figure 4.11   Hermite Cubic Value Function
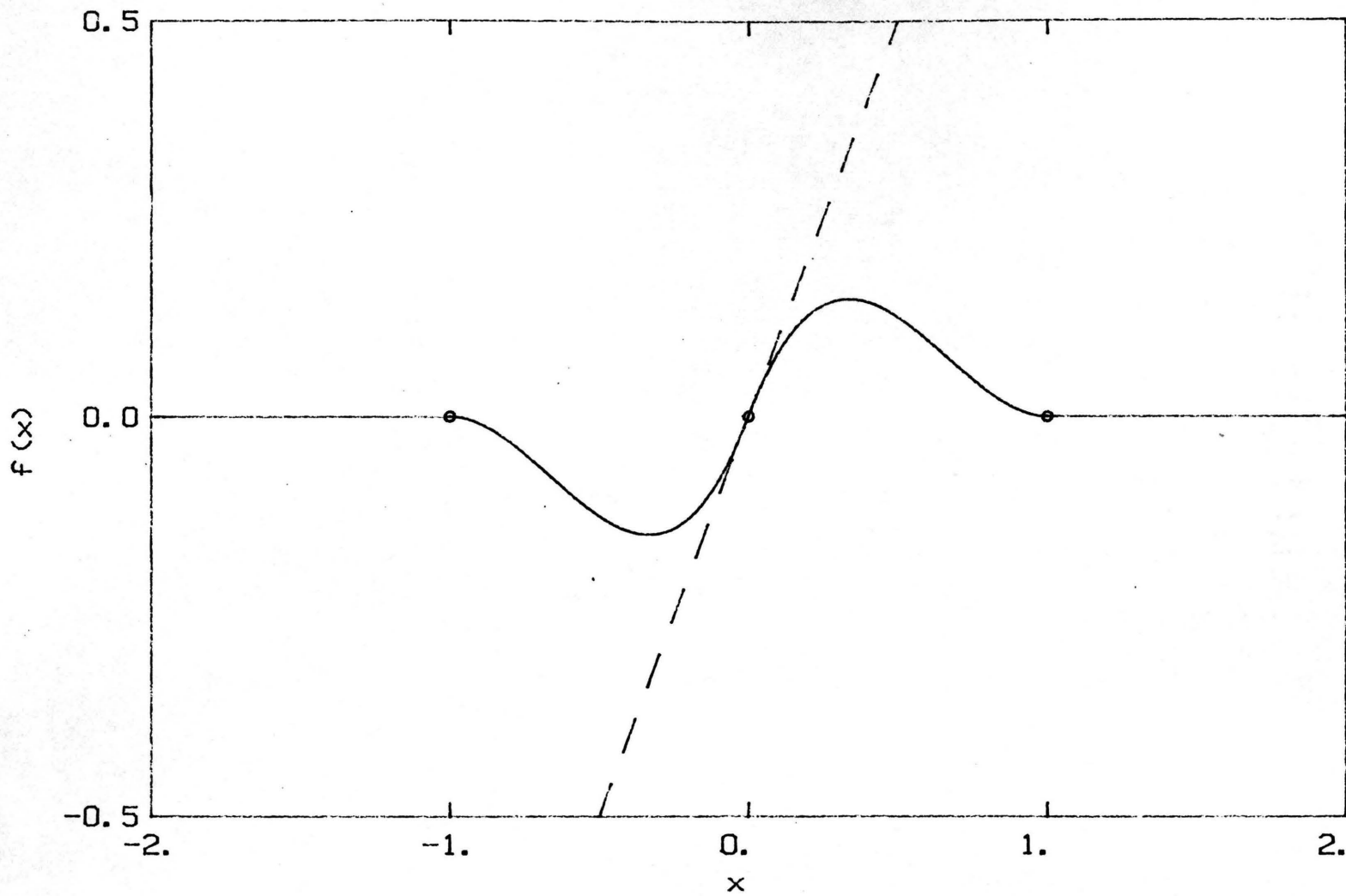
Figure 4.12 Slope Function Using Hermite Cubic Splines

113

Note that the values of the step-size (h_i, the distance between breakpoints) are indexed. Thus, each step can be different and the breakpoints may be distributed in an arbitrary manner. Suitable choices for breakpoint distribution may allow fewer breakpoints, and thus less computational effort for the same error, than an even distribution. This reduction in computational effort can be achieved by clustering the breakpoints in regions of greater importance to the solution. An example of an uneven distribution of breakpoints is given in Figure 4.13.

Figure 4.13 Non-Uniform Spatial Discretization
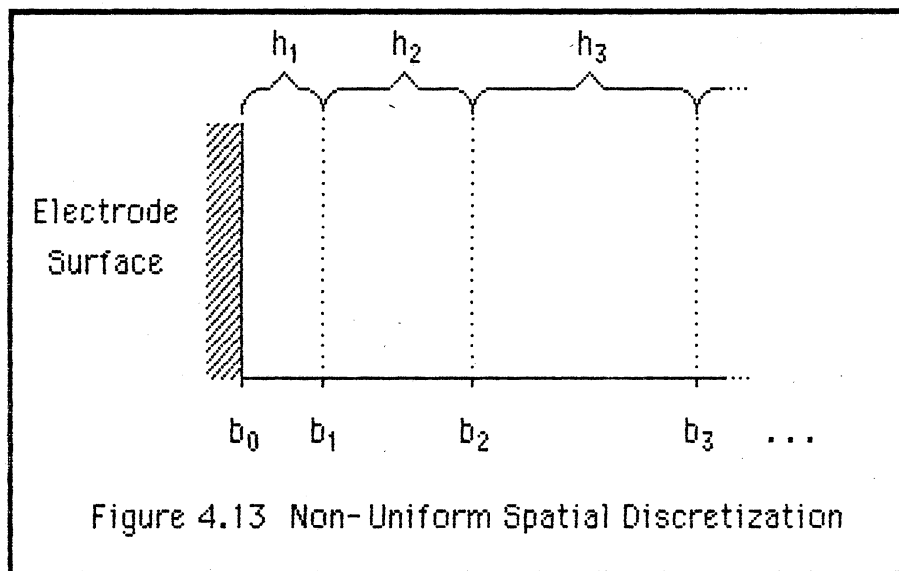
The Hermite cubic spline approximation of a function can be expressed as a linear combination of spline functions:

$$u(x) = \sum_{i=1}^{2n} \alpha_i \, \varphi_i(x) \qquad [4.42]$$

where $u(x)$ is the approximation function evaluated at x

α_i is the ith coefficient

$\varphi_i(x)$ is the ith spline function evaluated at x

n is the number of breakpoints (including the interval ends)

Because of the straightforward use of "value" and "slope" functions, it is quite easy to evaluate the coefficients:

$$\alpha_{2j-1} = f(b_j) \tag{4.43}$$

$$\alpha_{2j} = \frac{d\,f(x)}{d\,x}\bigg|_{x=b_j} \tag{4.44}$$

where f is the function to be approximated

j goes from 1 to n

b_j is the value of the jth breakpoint

The approximation of functions by Hermite cubic splines can be demonstrated by fitting the splines to a simple function. Figure 4.14 shows Hermite cubic value and slope functions approximating a hyperbola (dashed line) at a single breakpoint (shown as a crescent). While this example is not very practical, it illustrates the decomposition of the approximation into the value and slope of the function at a single point. A more practical example is shown in Figure 4.15; the hyperbola is approximated between two breakpoints.
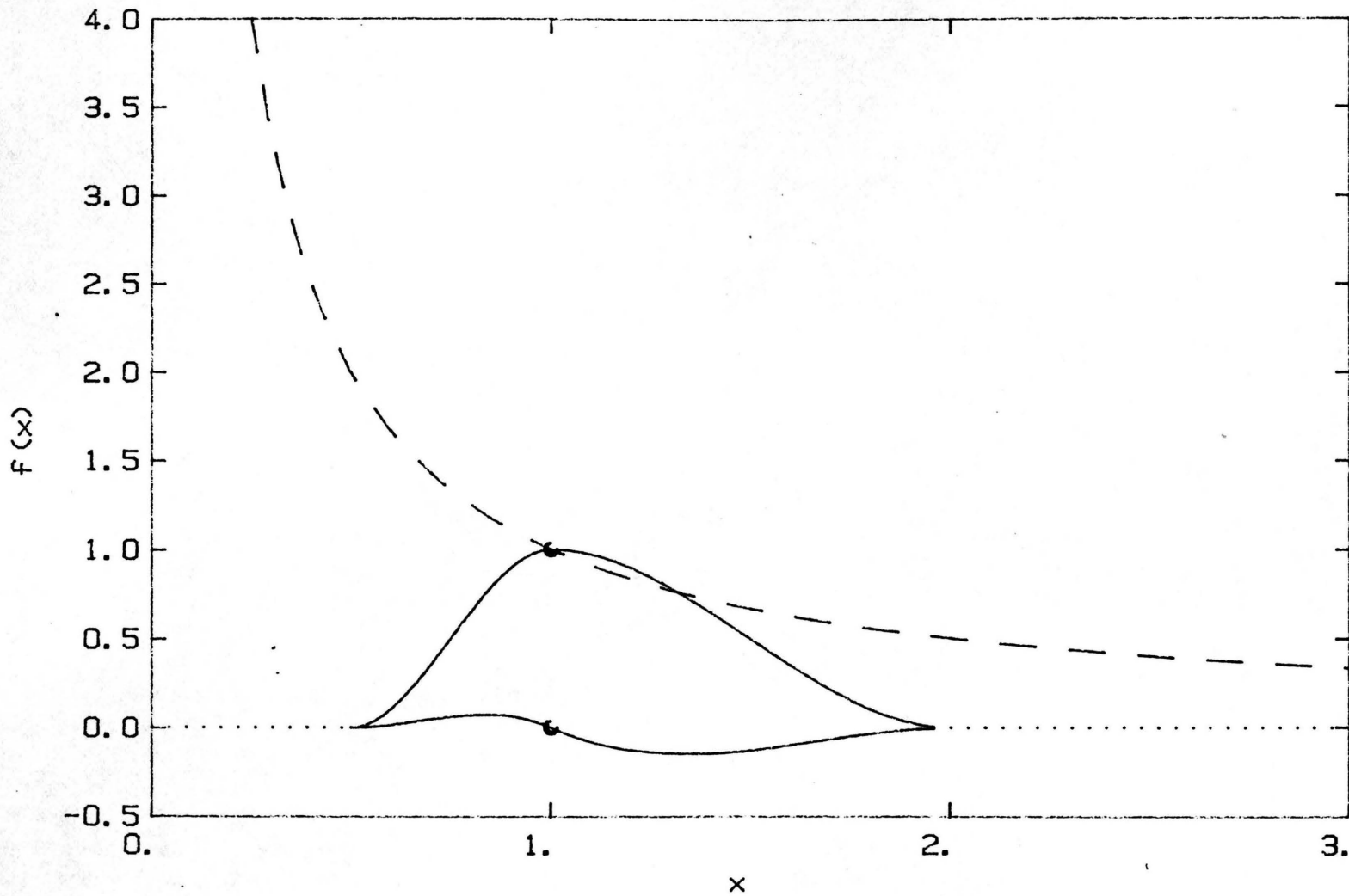
Figure 4.14   Components of a Hermite Cubic Spline
Approximation to a Hyperbola at a Single Point.
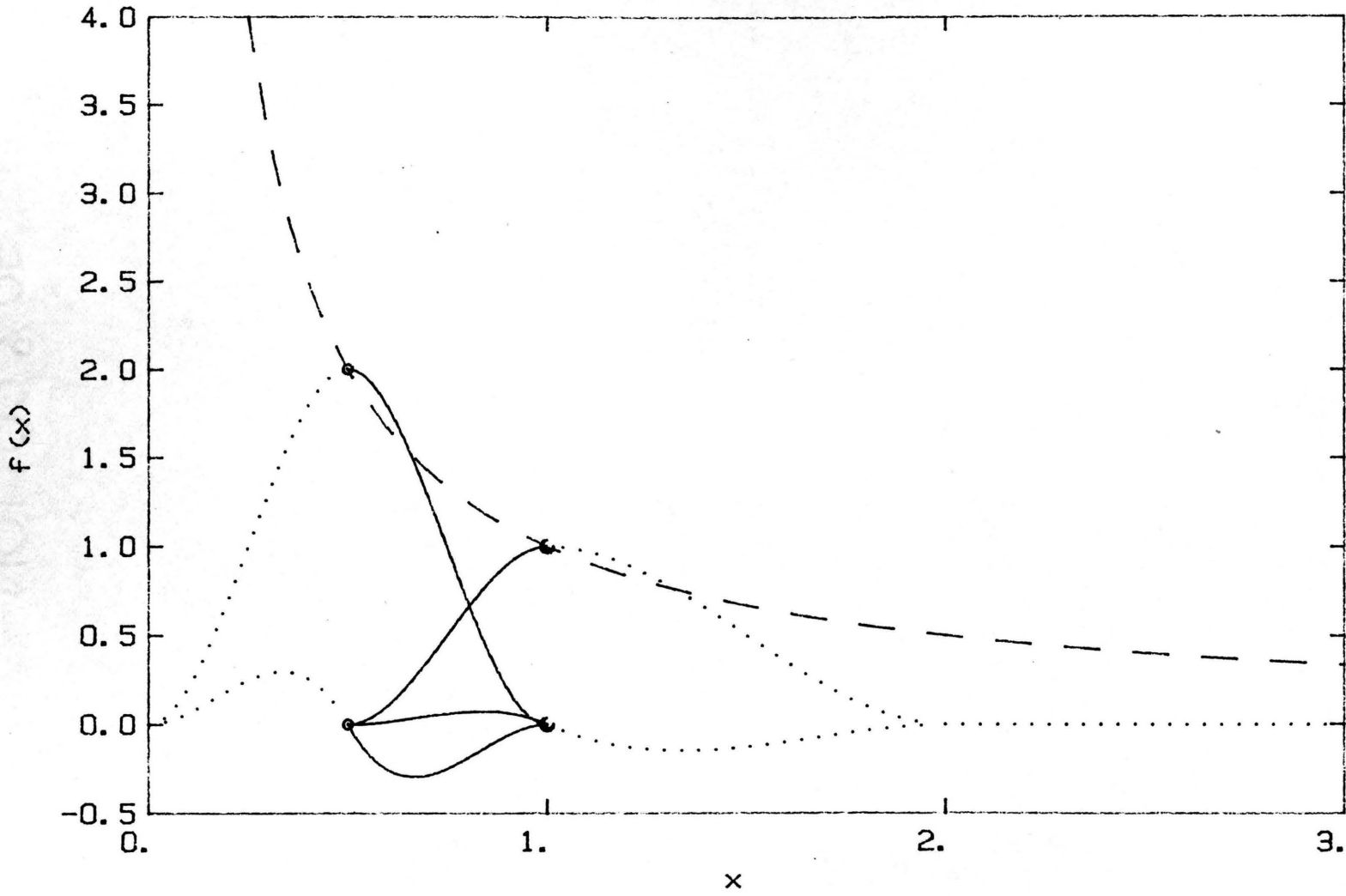
Figure 4.15 Components of a Hermite Cubic Spline Aproximation
for a Hyperbola Interpolated between Two Points.

Because the splines have finite values over two intervals between breakpoints, the approximated function at any non-breakpoint has contributions from four polynomial functions: two from the breakpoint to the left, and two from the breakpoint to the right. In Figure 4.15, the solid lines are the four spline functions that approximate the hyperbola between the two breakpoints. For these two breakpoints, four spline functions and four coefficients are needed for the approximation. Figure 4.16 shows in detail the four polynomial functions (solid lines) and the final interpolation function (dashed line) that is the sum of the four polynomials. In general, a set of n breakpoints requires 2n spline functions and 2n coefficients for a Hermite cubic approximation.

The difference between using splines and global polynomials for approximation functions can be illustrated by using three breakpoints for the approximation. The hyperbola has been chosen for the function to be approximated because it is usually a difficult function to fit well. Figure 4.17 shows the components of the Hermite cubic spline approximation to the hyperbola. The hyperbola is shown as a dashed line and the Hermite polynomials are shown as solid lines. The first, second, and third breakpoints are shown as a circle, a crescent, and a triangle. The total approximation is shown in Figure 4.18, where the solid line is the spline interpolation function. The dashed line is the original hyperbola and the quality of the fit can be seen by the difference between the two functions, original and approximation. The dotted line indicates the spline approximation outside the breakpoints, what might be considered the *extrapolatory* region. While splines are not defined for use as
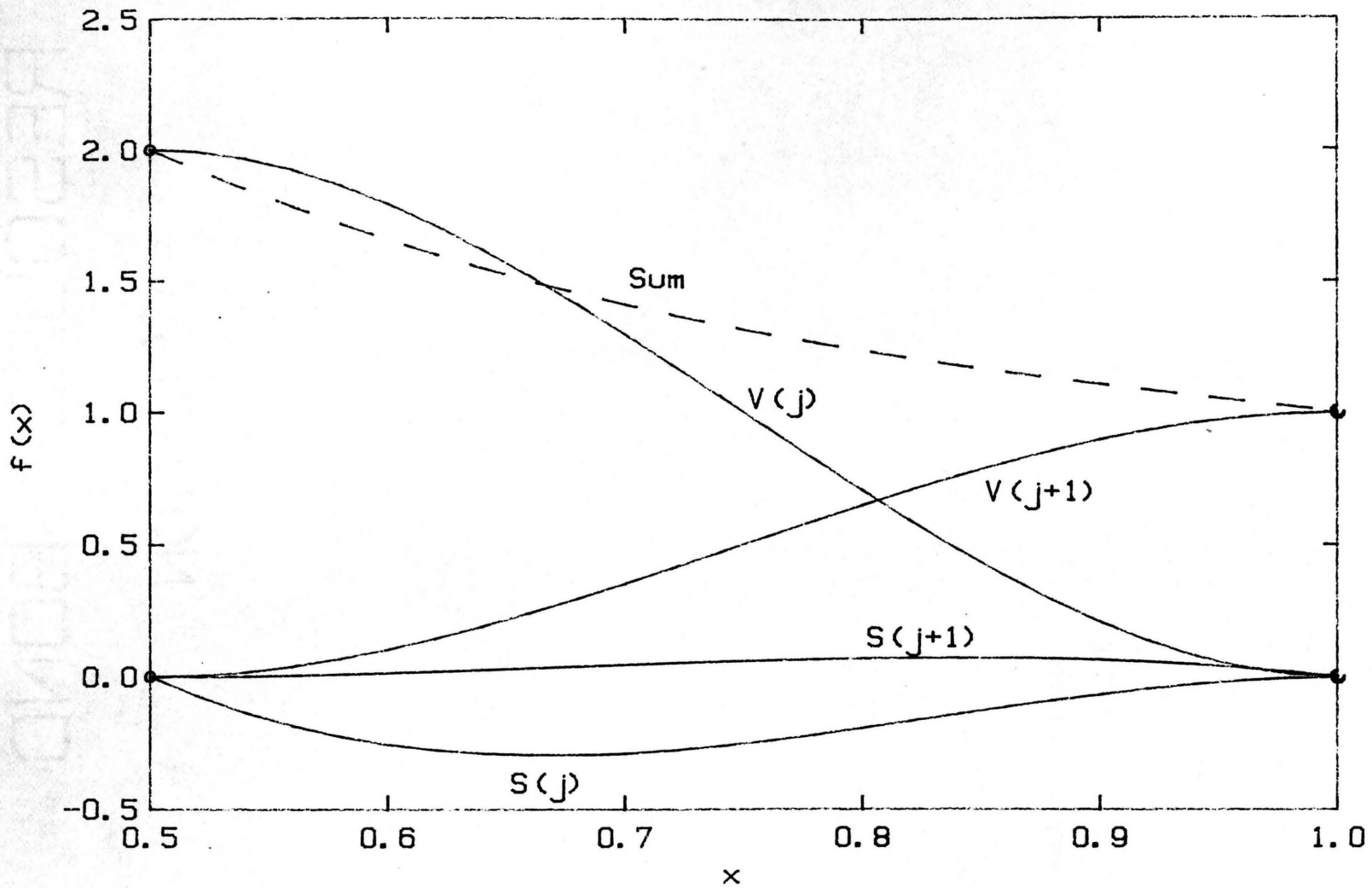
Figure 4.16   Components of a Hermite Cubic Spline Aproximation
Over a Single Interval between Breakpoints
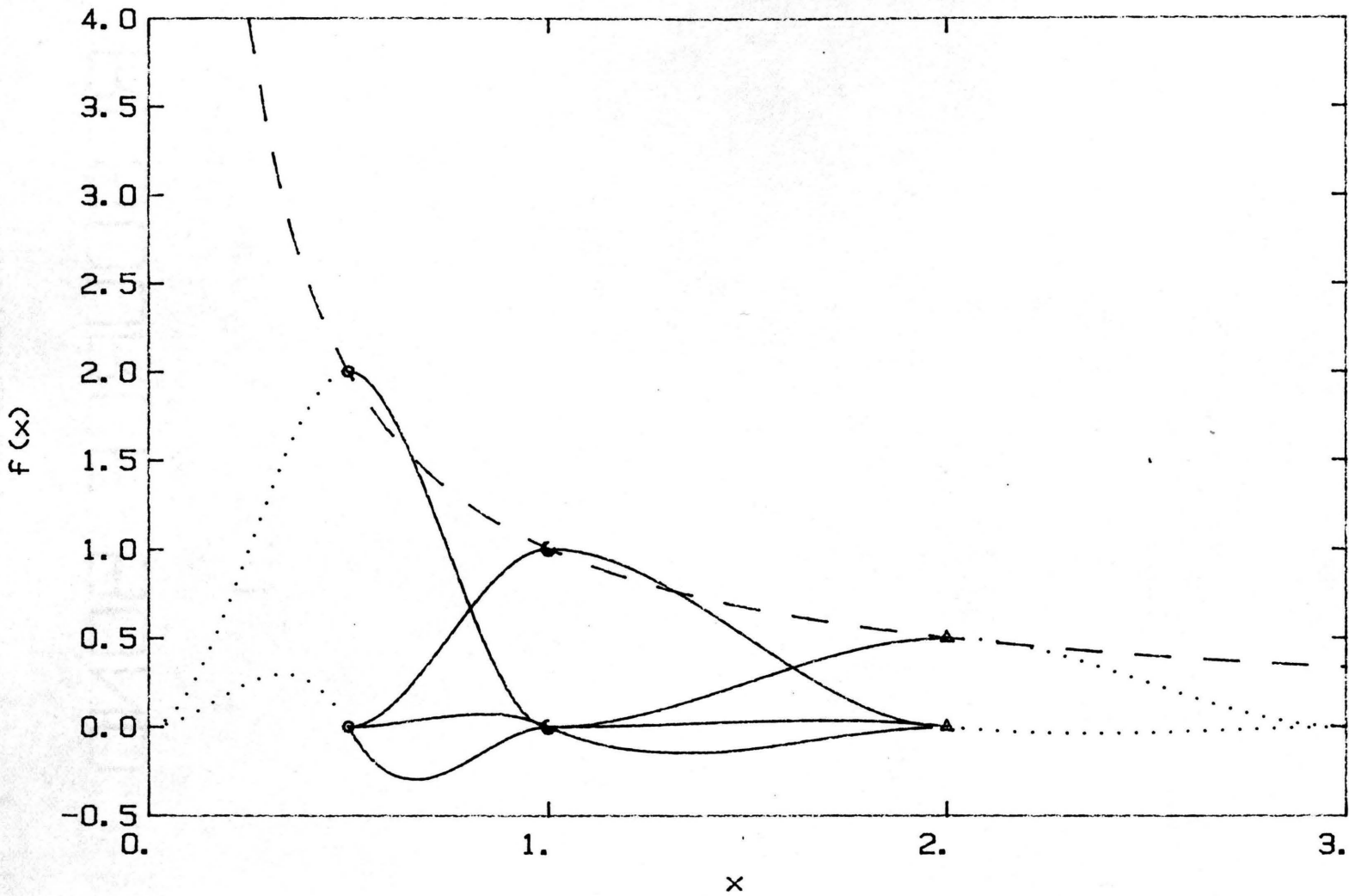
Figure 4.17   Components of a Hermite Cubic Aproximation to a Hyperbola

Figure 4.18   Hermite Cubic Spline Approximation to a Hyperbola

extrapolatory functions, it is instructive to note their well behaved nature in the smooth transition to a zero value outside the breakpoints. The same three breakpoints are used for a global polynomial calculated by using the *divided difference* method in de Boor [1979, page 10]. The result is a polynomial of order six, which is a degree five, or quintic, polynomial. The solid line in Figure 4.19 is this osculatory interpolant to the hyperbola (matching value and slope at each breakpoint). The dashed line is the hyperbola and the quality of the fit can be seen from the difference between the lines. The dotted line in Figure 4.19 shows the "extrapolatory" nature of the global polynomial; note particularly the divergent behavior for x values greater than 2. Global polynomials are also not extrapolatory functions. The relative errors of the two methods, splines and global polynomials, are illustrated in Figure 4.20. In this figure, the numeric difference between the approximation function and the original hyperbolic function is plotted. The errors at the breakpoints are forced to zero by the definitions of the approximation functions, and there is no effort to optimize the location of the breakpoints.

Given an approximation function, one next must develop a method for solving the differential equation using this function. A generalized technique, known as the "method of moments", can be used to develop many solution procedures. First, express the differential equation in an operator notation:

$$H(u) = g \qquad\qquad\qquad [4.45]$$
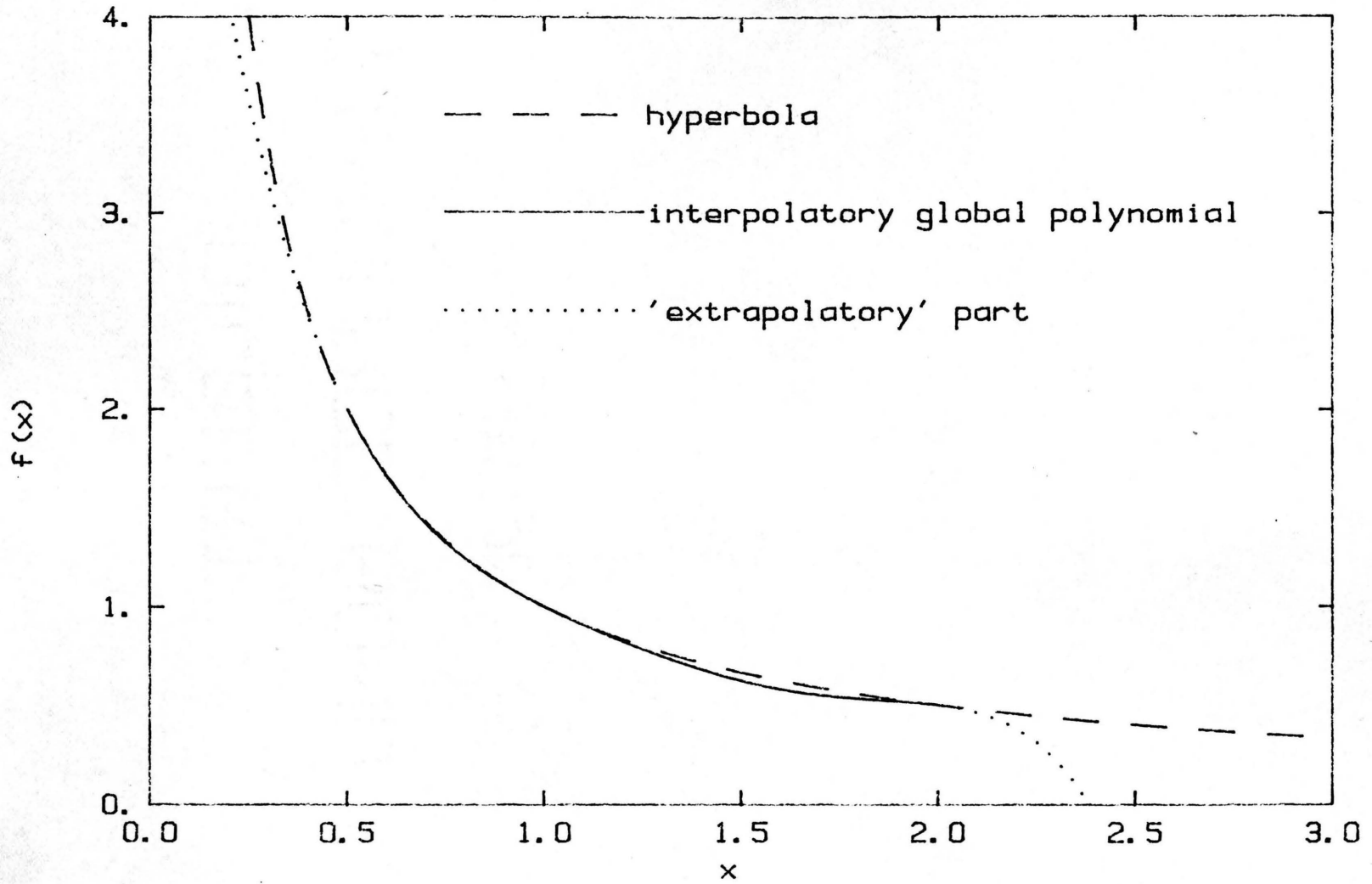
Figure 4.19 Global Polynomial Osculatory Interpolatant for a Hyperbola
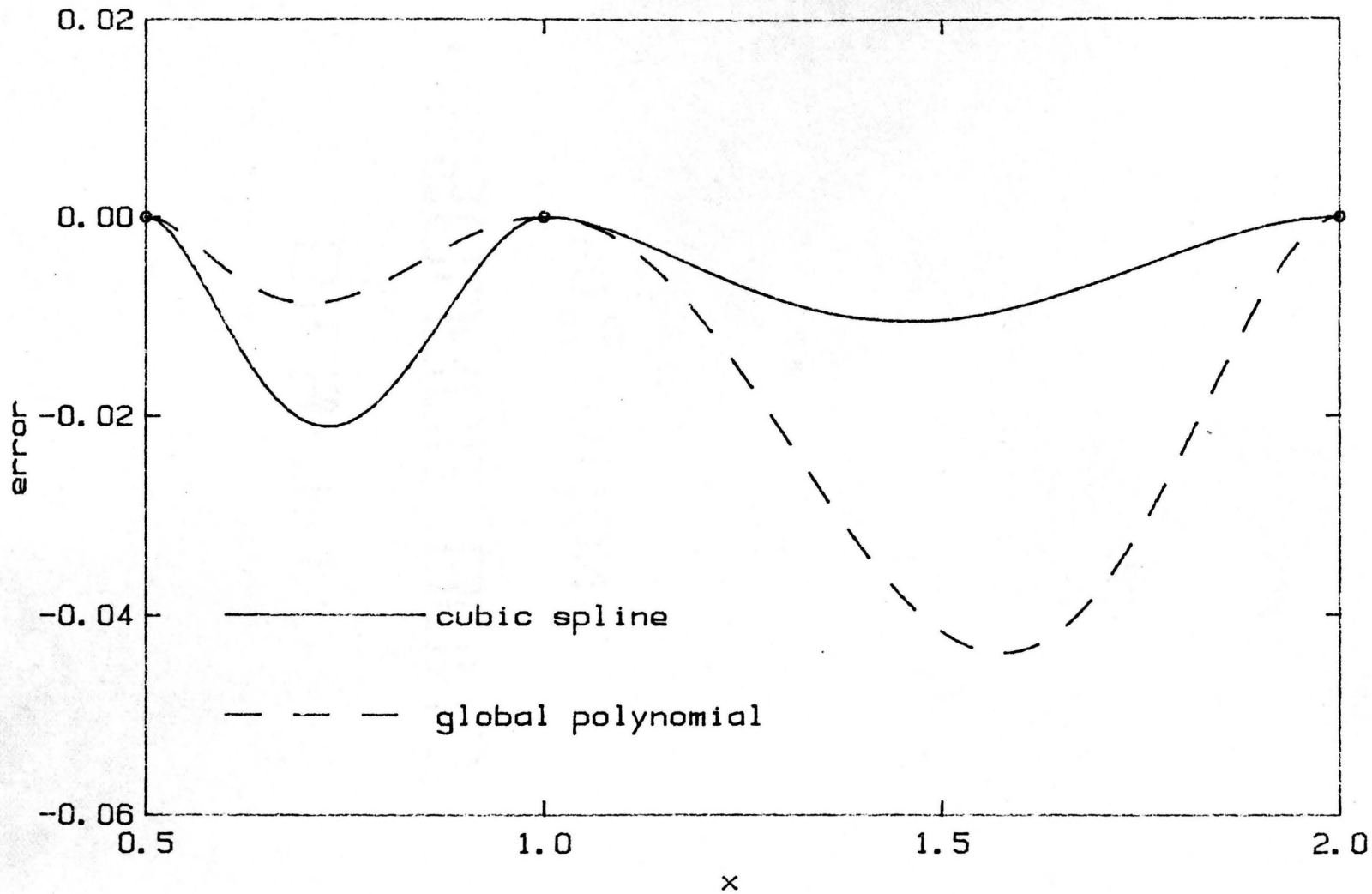
Figure 4.20   Comparison of the Approximation Error for
a Hermite Cubic Spline and a Global Interpolating Polynomial

124

where H is an operator

      u is the function approximating the solution

      g is a function determined by the problem

Then let an approximation to the solution be:

$$u(x) = \sum_{i=1}^{m} \alpha_i \, \varphi_i(x) \qquad [4.46]$$

Where the variables α are independent of x, and the functions φ are approximation functions (also known as basis functions). The method of moments states that the approximate solution can be found by solving the system of equations:

$$\sum_{i=1}^{m} \alpha_i \, \langle w_k, H[\varphi_i] \rangle = \langle w_k, f \rangle \qquad [4.47]$$

where w_k is a weighting function

  with k going from 1 to m

and the notation, $\langle w, f \rangle$ indicates an inner product:

$$\langle w, f \rangle = \int_{a}^{b} w(x) \, f(x) \, dx \qquad [4.48]$$

Two common solution methods can be derived in this manner. If the weighting functions are Dirac delta functions then the method is called "collocation". If the weighting functions are the approximating or basis functions themselves, then the method is called "Galerkin".

The collocation method was chosen for the present work for two reasons. First, the collocation method requires the least work in

evaluating the inner products. And second, if the problem is one dimensional in space or can be described in rectangular coordinates, a series of breakpoints will exactly span the region where the solution is desired. This is in contrast with a solution desired over an area or a volume for which there must be the equivalent of breakpoints and basis functions in two or three dimensions.

The delta function required to define collocation by the method of moments is:

$$w_m = \delta(x-g_m) \qquad\qquad [4.49]$$

Where the Dirac delta function ($\delta(z)$) is defined to meet the requirements:

1) if $z \neq 0$ then $\delta(z)=0$ $\qquad\qquad [4.50]$

2)
$$\int_{-\infty}^{+\infty} \delta(z)\, dz = 1 \qquad\qquad [4.51]$$

Substituting the delta function into the defining equation for the method of moments (equation 4.47):

$$\sum_{i=1}^{m} \alpha_i \; \langle\, \delta(x-g_k)\,,\, H[\varphi_i(x)]\, \rangle = \langle\, \delta(x-g_k)\,,\, f(x)\, \rangle \qquad\qquad [4.52]$$

From the properties of the delta function:

$$\sum_{i=1}^{m} \alpha_i \; H[\,\varphi_i(g_k)\,] = f(g_k) \qquad\qquad [4.53]$$

Which becomes the defining expression for the collocation method.

Returning to the mathematical model of the electrochemical system, express the approximate solution in the following form:

$$f(\tau,d) = \sum_{i=1}^{2n} \alpha_i(\tau)\, \varphi_i(d) \qquad\qquad [4.54]$$

Written this way, one can see an obvious relationship between this method and the method of separation of variables. It can be shown [de Boor 1978] that the best points for the Dirac delta functions of the collocation method are at the Gauss points. For the Hermite cubics, these are the zeros of the quadratic Legendre polynomial, which are:

$$g_{2i} = b_i + h_i (1 - 1/\sqrt{3}) / 2 \qquad\qquad [4.55]$$

$$g_{2i+1} = b_i + h_i (1 + 1/\sqrt{3}) / 2 \qquad\qquad [4.56]$$

where i goes from 1 to n-1

b_i is the ith breakpoint

$h_i = b_{i+1} - b_i$ is the interval size

Recalling that the differential equation is:

$$\frac{\partial f(\tau,d)}{\partial \tau} = \frac{\partial^2 f(\tau,d)}{\partial d^2} \qquad\qquad [4.57]$$

In this case, the operator, H, is:

$$H = \frac{\partial}{\partial \tau} - \frac{\partial^2}{\partial d^2} \qquad\qquad [4.58]$$

The function on the right-hand side of equation 4.45 is zero. Substituting the expression for the solution approximation into the expression for the above method of collocation:

$$\sum_{k=1}^{2n} \varphi(g_k) \frac{\partial \alpha_i(\tau)}{\partial \tau} - \sum_{k=1}^{2n} \alpha_i(\tau) \left. \frac{\partial^2 \varphi(d)}{\partial d^2} \right|_{d=g_k} = 0 \qquad [4.59]$$

This can be written in matrix/vector notation as:

$$A \frac{\partial \alpha(\tau)}{\partial \tau} - C \, \alpha(\tau) = 0 \qquad [4.60]$$

or

$$A \, \dot{\alpha}(\tau) = C \, \alpha(\tau) \qquad [4.61]$$

where

$$\dot{\alpha}(\tau) = \frac{\partial \alpha(\tau)}{\partial \tau} \qquad [4.62]$$

the elements of matrix **A** are: $\quad a_{ij} = \varphi_i(g_j) \qquad [4.63]$

the elements of matrix **C** are: $\quad c_{ij} = \left. \frac{\partial^2 \varphi_i(d)}{\partial d} \right|_{d=g_j} \qquad [4.64]$

The values, g_i, are the previously defined Gauss points.

For n breakpoints, there are (n-1) intervals, (2n) coefficients, and (2n -2) collocation (Gauss) points. From the above matrix, one can see that there are 2n-2 equations in 2n unknowns. The two additional

equations are provided by the two boundary conditions. The first condition
is that f=1 at the last breakpoint. To incorporate the time dependence of
the solution, the derivative with respect to time is taken, and the added
equation is:

$$\text{at} \quad d = d_{max} = b_n, \quad f=1 \qquad\qquad [4.65]$$

$$\text{thus} \quad \alpha_{2n-1}(\tau) = 1 \qquad\qquad [4.66]$$

giving,

$$\frac{\partial \alpha_{2n-1}(\tau)}{\partial \tau} = 0 \qquad\qquad [4.67]$$

The second condition relates to the rate of reaction at the surface.
At the first breakpoint, the dimensionless concentration gradient is equal
to the net dimensionless rate of reaction:

$$\text{at } d=0, \quad \frac{\partial f(\tau,d)}{\partial d} = f\, k_f(\tau) - (1-f)\, k_b(\tau) \qquad\qquad [4.68]$$

$$\text{or} \quad \alpha_2(\tau) = \alpha_1(\tau)\, [k_f(\tau) + k_b(\tau)] - k_b(\tau) \qquad\qquad [4.69]$$

The rate constants depend in a nonlinear fashion on the applied
potential, and that potential is often varied with time. That is why the
explicit dependence on time is shown. The usual way of incorporating this
dependence into the model is to take the derivative of the boundary
condition with respect to time. However, this was found to be inefficient
in the numerical algorithim. So the additional condition of satisfying the

differential equation at the surface of the electrode was used as the second boundary condition (this will be discussed further in chapter 5). This effectively makes the first breakpoint into a collocation point. The equations representing the boundary conditions as a function of time can be added to the equations representing the collocation formulation. This leads to the matrix system:

$$A \dot{\alpha} = C \alpha + D(\alpha, E) \qquad\qquad [4.70]$$

where E is the applied voltage

D is a nonlinear vector function

This describes the system fully and the approximation can be made as exact as one pleases at the expense of increased computation time and increased computer storage requirements.

This chapter closes with a comparison of the different approximation functions described at the start and their associated methods for solving the differential equation. The three approximation functions discussed were: a vector of points, global polynomials, piecewise polynomials, and natural basis functions derived by the method of Sørensen and Stewart. Table 4.1 on the following page compares the various options.

Table 4.1 A comparison of four approximation techniques used
in solving Parabolic Differential Equations

Method Number	Approximation Function	Points used(n)	error order	machine storage	execution time	ease of formulation	error redist.
1	vector of points	500	Δt^2+h^2	$5n=$ 2500	$6n^2=$ 1500000	easy	yes
2	global polynomials	≈20	*	$2n^2+2n=$ 840	$12n^3=$ 96000	moderately difficult	no
3	cubic splines	22	Δt^4+h^4	$12n=$ 264	$120n^2=$ 58000	moderately easy	yes
4	'natural' basis	≈5	na	$2n^2+2n=$ 21	$2n^3=$ 250	rather difficult	no

*Note: the order of the error using a global polynomial should be h to the power of the polynomial. However, an examination of the results of Whiting and Carr [1977] indicates that because of the inability to locate the collocation points in the areas of greatest error, the order must be higher to give the same error as a piecewise polynomial with variable interval size.

This table is by no means rigorous; it is intended merely to give a relative comparison. In preparing this table, an effort was made to represent the qualitative nature of the various solution methods, and not necessarily the precise measure of effectiveness. In the absence of reliable error estimates, it is difficult to compare the methods. The problem is exacerbated by the fact that the error estimates are true only in the limit of very small interval size, and very large storage requirements, and the thrust of the present work is to minimize the size of the problem. The error bounds at these larger interval sizes are not well characterized.

The starting point for table 4.1 was the number of points commonly used for an adequate finite difference solution (method 1) for typical electrochemical problems [Brown 1984, Magno 1983], about 500 points. The error order, storage size, and execution time for finite differences was for a Crank-Nicholson method [Davis 1984]. Storage size was measured as the number of floating point variables. Execution time was measured approximately as the total number of floating point operations in a simulation. The order of the time truncation error and the spatial truncation error were kept the same such that the number of time steps and the number of spatial steps were kept approximately equal. Both the global polynomial (orthogonal collocation, method 2) and the natural basis function methods (method 4) result in dense matrices and the storage space and execution times are from literature on matrix methods [Davis 1984, Finlayson 1980]. As noted under table 4.1, error estimates for global polynomial and natural basis functions are not available. The issue

is further complicated by the fact that the collocation points for methods 2 and 4 are fixed while the discrete points for method 1 and the breakpoints for methods 3 can be adjusted to concentrate the points in regions of greater approximation error. This is referred to as "error redistribution" in the table, and depends on locating more collocation points in areas with steep gradients in the solution function. For the purposes of the table, all discrete points and breakpoints were assumed to be evenly distributed. The problem of the distribution of these points considered further in the section on implementation of the model in the next chapter.

The number of collocation points for methods 2 and 4 had to be very rough estimates "guesstimated" from the literature on the methods: [Whiting 1977, Magno 1983] and [Sørenson 1982]. The number of breakpoints for method 3 was a direct scaling from the order of the truncation errors for methods 1 and 3. As will be discussed at length in the next chapter, method 3 uses banded matrices. Storage and execution times for this method were from Finlayson [1980].

In the end, Hermite cubic spline collocation was the method of choice because of: (1) its ability to distribute the breakpoints so as to minimize the error, (2) its relative simplicity of implementation, and (3) lower storage requirements and faster execution than methods 1 and 2. The use of the "natural" basis functions of Sørensen and Stewart may in time prove to be the best in terms of storage and execution time, but this is in the balance against the amount of effort to develop the functions. The natural basis functions would probably require the use of a computer

algebra system (programs such as MACSYMA or muMATH) to automatically generate the functions. On the other hand, automated numerical procedures exist for generation of both spline and global polynomials. Presently, there is no mathematical guarantee that the "natural basis function" method will converge to a solution, and there is no error estimation procedure for this method. For spline collocation (method 3), convergence can be shown and error estimates calculated. The optimum for execution time, storage requirements, and programming time is not well defined, but it is reasonable to expect that the spline approach is an efficient choice.

Chapter 5. Computer Implementation of a Mathematical Model for
an Electrochemical Oscillogenic Instrument

"A human being should be able to change a diaper, plan an invasion, butcher a hog, conn a ship, design a building, write a sonnet, balance accounts, build a wall, set a bone, comfort the dying, take orders, give orders, cooperate, act alone, solve equations, analyze a new problem, pitch manure, program a computer, cook a tasty meal, fight efficiently, die gallantly."

<div align="right">

Robert A. Heinlein
"The Notebooks of Lazarus Long"
Time Enough for Love

</div>

The electrochemical system for an oscillogenic implementation is described by the mathematical model consisting of the differential equation 4.18 with boundary conditions as in equations 4.19 and 4.20. As described in chapter 4, this system of equations will be solved using a cubic spline approximation (equation 4.56). A computer algorithm must be used to calculate the time and space dependence of the approximate solution. This chapter will give a *top-down* explanation of the implementation of this computer algorithm as a FORTRAN program on a digital computer. The expression, *top-down*, means that one starts with the overall problem and describes its solution by breaking it down into smaller problems, the smaller problems into still smaller problems, and so on.

The overall goal of the computer program is to determine the solution of a differential equation (4.18) by solving the following system of equations:

$$A \dot{\alpha} = C \alpha + D(\alpha, E) \qquad\qquad [4.72]$$

·To do this, the values for the elements of the matrices **A** and **C** (equations 4.64 and 4.65), and the elements of vector **D** must first be determined. These values are evaluated from the spline functions; in turn, the spline functions require a series of breakpoints for evaluation (equation 4.40 and 4.41). Given an initial condition for the vector of coefficients, $\alpha(\tau=0)$, the rate of change of each coefficient can be calculated, by equation 4.72 above. If the rate of change at every point in time is known, then these rates can be integrated to calculate the coefficients as a function of time. If the coefficients are known for every desired point in time, then the problem is solved.

Breaking these subproblems down further:

1. To fill **A**, **C**, and **D**, one must:
   a. Choose a vector of breakpoints.
   b. Fill the elements of the matrices.
      i. Calculate the values of the spline functions at the Gauss points.
      ii. Calculate the values of the functions at the boundaries of the electrolyte.
2. To find the solution, one must:
   a. calculate **C**α
   b. calculate **C**α + **D**
   c. calculate $d\alpha/dt$ = [**C**α + **D**]/**A**
   d. integrate $d\alpha/dt$
   e. start and stop simulation, report results

These tasks are parceled out to various FORTRAN subroutines:

    1.a REDIST, REDISTribution of the breakpoints, Listing 1

    1.b FILL, FILL matrix elements with values, Listing 4

    1.b.i BASWTS, BASis function WeighTS, Listing 5

    1.c.ii INTWTS, INTerior basis function WeighTS, Listing 6

    2a. BANMUL, BANded matrix MULtiply, Listing 9

    2b. QUASI, QUASI-reversible electrochemical kinetics, Listing 19

    2c. BANDET, BANded matrix DEcomposition, Listing 12

        and BANSOL, BANded matrix SOLver, Listing 13

    2d. RUKUFE, RUnge-KUtta-FEhlberg initial value problem solver,

        Listing 16

    2e. QUASELEC, QUASi-reversible ELECtrochemical simulation,

        Listing 20

The 'Listing' numbers refer to the computer listings in the appendix of
this work. These subproblems and the accompanying FORTRAN subroutines
will be discussed in order. Unusual capitalization of the names and titles
of the listings is used to illustrate mnemonic file names for the machine
readable copies of the programs. For example, "REDISTribute break points
subroutine (FORtran)" indicates a file called "REDIST.FOR" on a magnetic
disk. Magnetic disk copies of the listings are available for MS-DOS (IBM
PC compatible) computers and for the Apple Macintosh.

1a. Choose a vector of breakpoints.

    An initial guess can be made for the breakpoints based on some

intuitive idea of the resulting solution. Since the concentration will change most drastically near the surface of the electrode, the breakpoints should be spaced such that more breakpoints are placed closer to the surface than farther. Some guessing may result in a solution, but the question remains: What are the optimal locations for the breakpoints? Optimal in this case may mean either to minimize the solution error for a given number of breakpoints, or to minimize the number of breakpoints for a given error. Minimizing the size and execution time of the model is important, since the model is being run on a microcomputer. One additional consideration is that the program might be embedded within a larger one for parameter estimation [see beginning of chapter 4 and all of chapter 7]. For these reasons, the breakpoints were chosen to minimize the storage requirements and execution time within an acceptable error.

The location of the first breakpoint is the surface of the electrode. The last breakpoint must represent the bulk concentration of the active species in the solution. Two ways of choosing the value of this breakpoint are (1) transform the spatial coordinates such that infinity becomes some finite value that is used for the last breakpoint or (2) pick a value for the spatial coordinate that is large enough such that it is effectively infinity. The method used in this work is the latter; the value is picked experimentally such that during the desired solution time interval, the slope does not differ from zero by more than some small tolerance. At this point, the question is how to distribute the remaining breakpoints such that the error is minimized.

An example of the problems of breakpoint selection can be provided

from a simple illustration of the electrochemical situation. Let us take two breakpoints: one at the surface of the electrode and the other at some point in the bulk electrolyte solution. The values of the spline approximation must fit the boundary conditions at these points. Let the concentration at the surface be approaching zero, and the concentration in the bulk be unity. The slope of the approximating function at the breakpoint representing the bulk concentration must be zero. So there are only two spline functions over this interval: the slope function for the first breakpoint, and the value function for the second breakpoint. The value function, at the second breakpoint, has a fixed coefficient of unity but the slope function can be any value needed to correspond with the concentration gradient at the surface and thus the electrical current flowing to the electrode surface. Since one might expect this current to take on a large range of values, what range of values will give a reasonable approximation of the physical situation for a given breakpoint distribution? Figure 5.1 shows a series of approximating functions with different slopes at the electrode surface (x=0). The physical situation demands that the concentration never be greater than unity, because the species is being consumed at the surface. Two of the slopes shown give no problem. But the largest slope (slope=5) shows a numerical artifact: a concentration greater than physically possible. Thus the spline approximation is limited in how steep a gradient it can represent for a given step size (distance between breakpoints).

Figure 5.1 Spline Approximations of Different Slopes
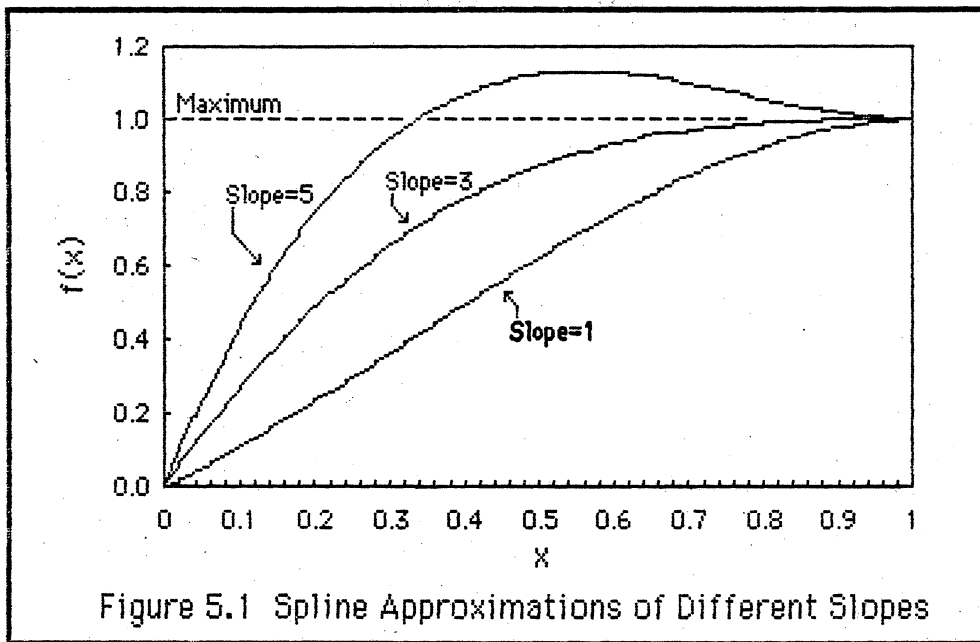
The maximum slope in this particular case can be represented as:

max slope = 3/h   [5.1]

    where h is the step size

As an approach to the problem of setting the locations of breakpoints to minimize the error, Ascher et al [1978] suggested that if the approximation error is equally distributed along the spatial dimension, then the overall error will be minimized. This reference suggested a technique for redistributing the breakpoints such that an estimate of the approximation error is evenly distributed across all breakpoint intervals. The suggested technique is used in the commercial software package, COLSYS, which solves boundary value problems rather than the parabolic partial differential equation discussed here. An implementation of this

algorithm was coded into a subroutine, REDIST (Listing 1). This routine
was used on a trial-and-error basis to adjust the breakpoints for
acceptable performance. This subroutine takes a vector of coefficients
and the breakpoint vector, and calculates a new vector of breakpoints with
more evenly distributed approximation error. A short example program
that demonstrates the use of this subroutine is in Listing 2, and the
output of this example program is in Listing 3.

Note that Ascher et al's algorithm, does not necessarily converge
toa fixed set of breakpoints. During the trial and error process it was
observed, that if the redistribution subroutine was repeatedly used to
adjust the breakpoints, the breakpoint vector often did not approach a
constant vector but rather jumped around considerably. Some judgment
must be used even with this method, but the subroutine REDIST can be
used as a guide to choose an appropriate vector of breakpoints.

Figure 5.2 shows an example of the use of this algorithim. A
portion of a hyperbola is approximated by Hermite cubic splines. On the
interval interpolated, four breakpoints are used. First, an even
distribution of breakpoints is used. Then the 'COLSYS' algorithm, as coded
in the subroutine REDIST, is used to calculate an improved series of
breakpoints that will tend to even out the error across the interval. Since
the function to be approximated is known exactly, the approximation error
is just the numerical difference between the spline approximation and the
hyperbola. Note that with the improved set of breakpoints, the maximum
error is much less, and the error seems to be spread out more evenly
across the interval.
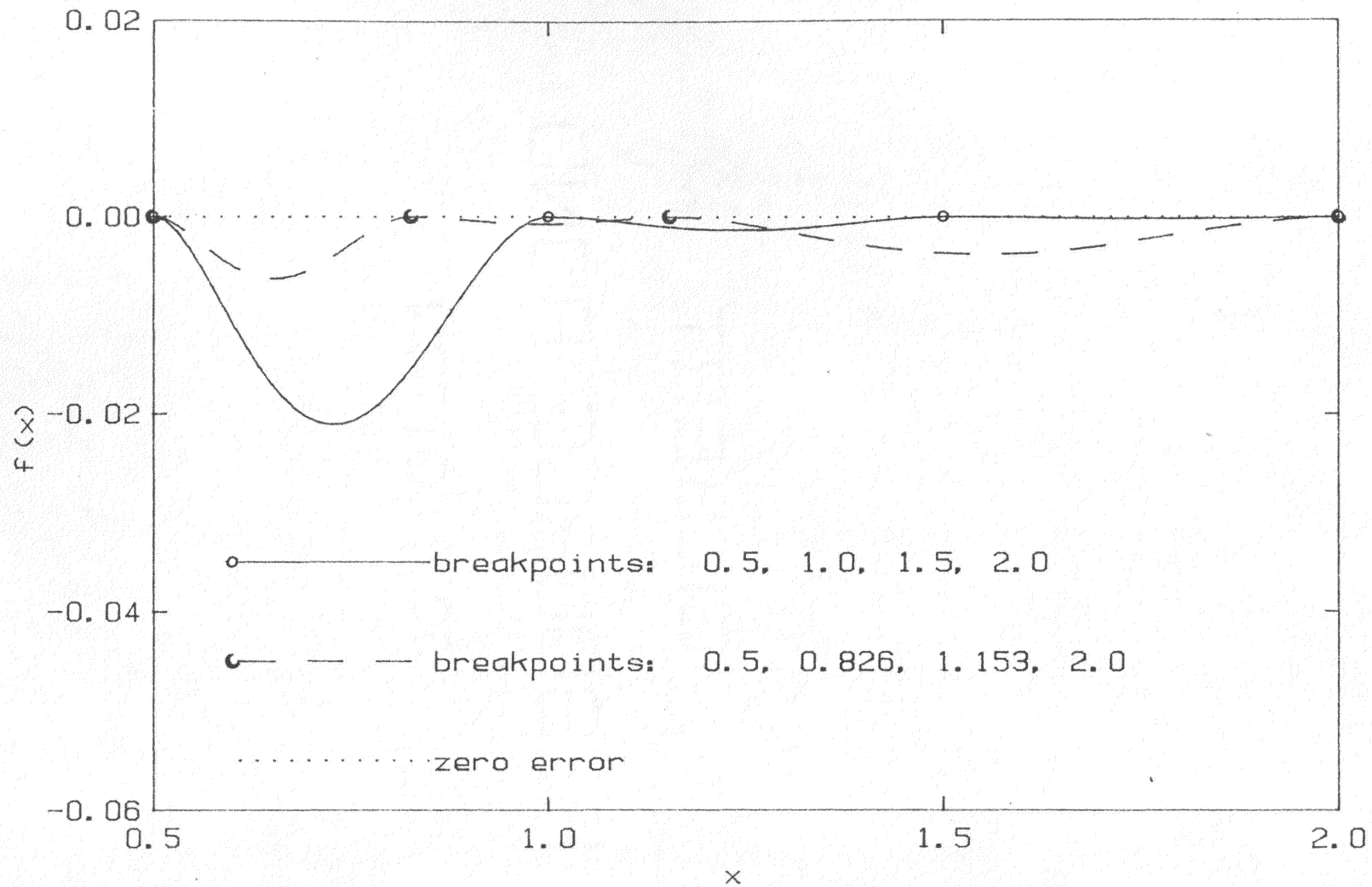
Figure 5.2 Approximation Error for Hermite Cubic Spline Approximation
to a Hyperbola for Two Different Breakpoint Distributions

1.b Fill the elements of the matrices

The subroutine called FILL, Listing 4, does the actual filling of the matrices with numerical values. This subroutine was not written as a general purpose routine; therefore no example programs are provided (to observe how the routine operates see the "useful for debugging" comments in the program QUASELEC described later). Note that because of the nature of the spline functions, the spline values will be non-zero only over the interval of three breakpoints at most. Thus the matrices have many zero values. The nonzero elements are along only five diagonals, and the matrices can be stored in a reduced structure known as a *banded* form.

$$
\begin{bmatrix}
X & X & O & O & O & O & O & O & O & O \\
X & X & X & X & O & O & O & O & O & O \\
X & X & X & X & O & O & O & O & O & O \\
O & O & X & X & X & X & O & O & O & O \\
O & O & X & X & X & X & O & O & O & O \\
O & O & O & O & X & X & X & X & O & O \\
O & O & O & O & X & X & X & X & O & O \\
O & O & O & O & O & O & X & X & X & X \\
O & O & O & O & O & O & X & X & X & X \\
O & O & O & O & O & O & O & O & X & X
\end{bmatrix}
$$

X ≡ finite values

O ≡ zero values

╲ ≡ line separating band structure

Figure 5.3 Structure of the banded matricies **A** and **C** for a 5 breakpoint model.
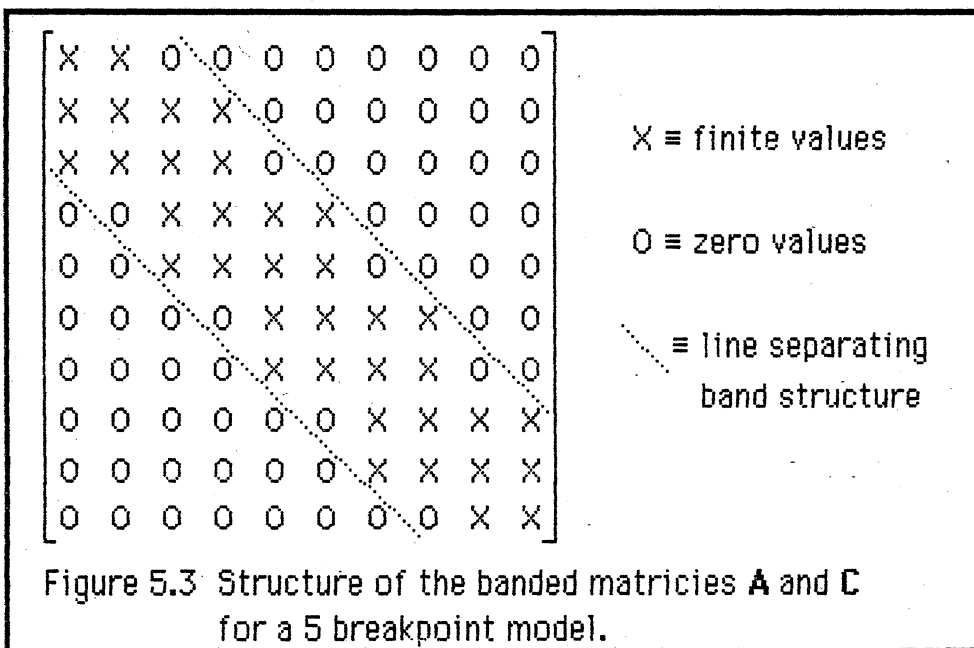
By storing only the matrix elements within the "band", the machine storage requirements are reduced. Algorithms for matrix manipulation

that are designed for the band structure can be made more efficient than those for a full or *dense* matrix. Figure 5.3 shows a schematic view of the banded matrix structure.

1.b.i Calculate the values of the spline functions at the Gauss points.

The breakpoints are used to define the spline functions and fill the **A** and **C** matrices with the values of the spline functions evaluated at the Gauss points. Since the Hermite cubic spline functions are used as the *basis* for a solution to the differential equation, the Hermite cubic splines are called the *basis functions*. The subroutine that evaluates the values of the basis functions at the Gauss points is called BASWTS (Listing 5), which is short for "BASis function WeighTS". This subroutine is taken from the routine of the same name by Dixon [1980].

An intuitive picture of the spline functions is useful for recognizing errors in the numerical results. Figure 5.4 shows the basis functions that are *supported*, that is, have non-zero values over an interval between two breakpoints. *Supported* means that these are the only non-zero basis functions on the interval between the two breakpoints. There are right and left 'value' functions, and right and left 'slope' functions. The Gauss points are also shown in this figure as dotted lines. These points are used to solve the differential equation by forcing the approximate solution to satisfy exactly the differential equation (4.18) at these points; these are also known as the *collocation* points. The values of the spline functions at the Gauss points are required in the calculations. First and second derivatives of the spline functions might also be required.
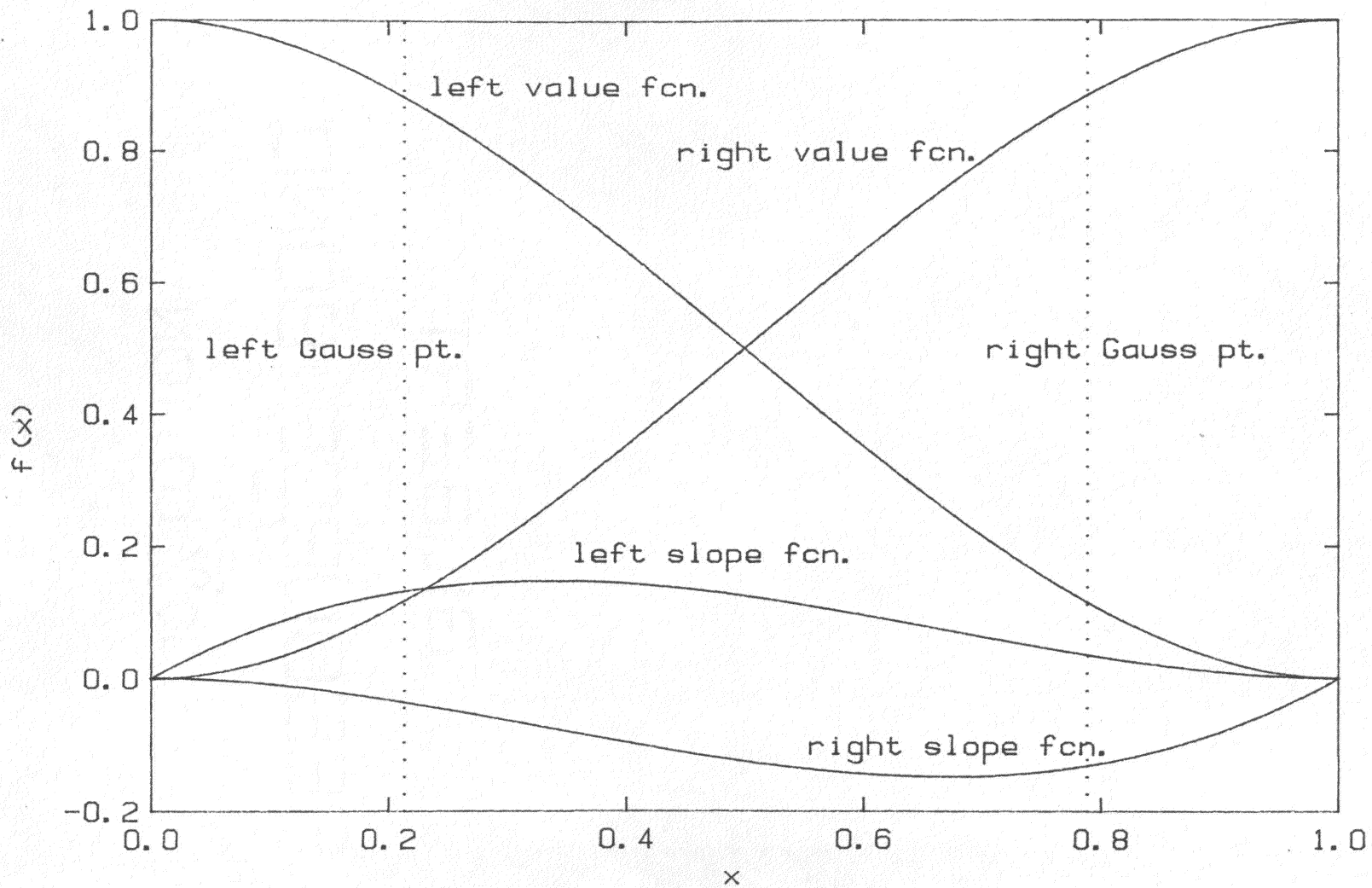
Figure 5.4 Basis functions supported over one interval
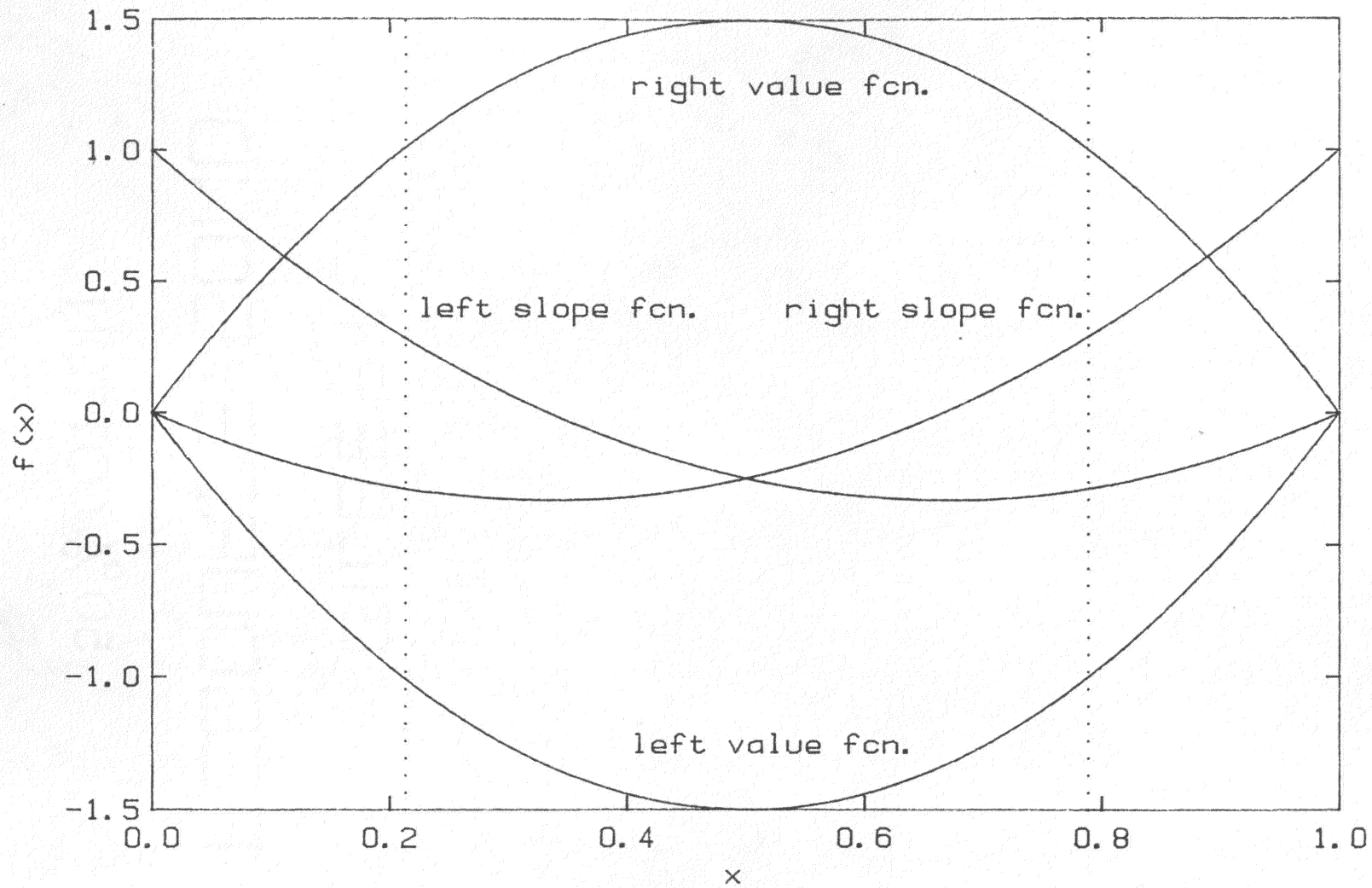
Figure 5.5 First derivatives of the basis functions
which are non-zero over one breakpoint interval
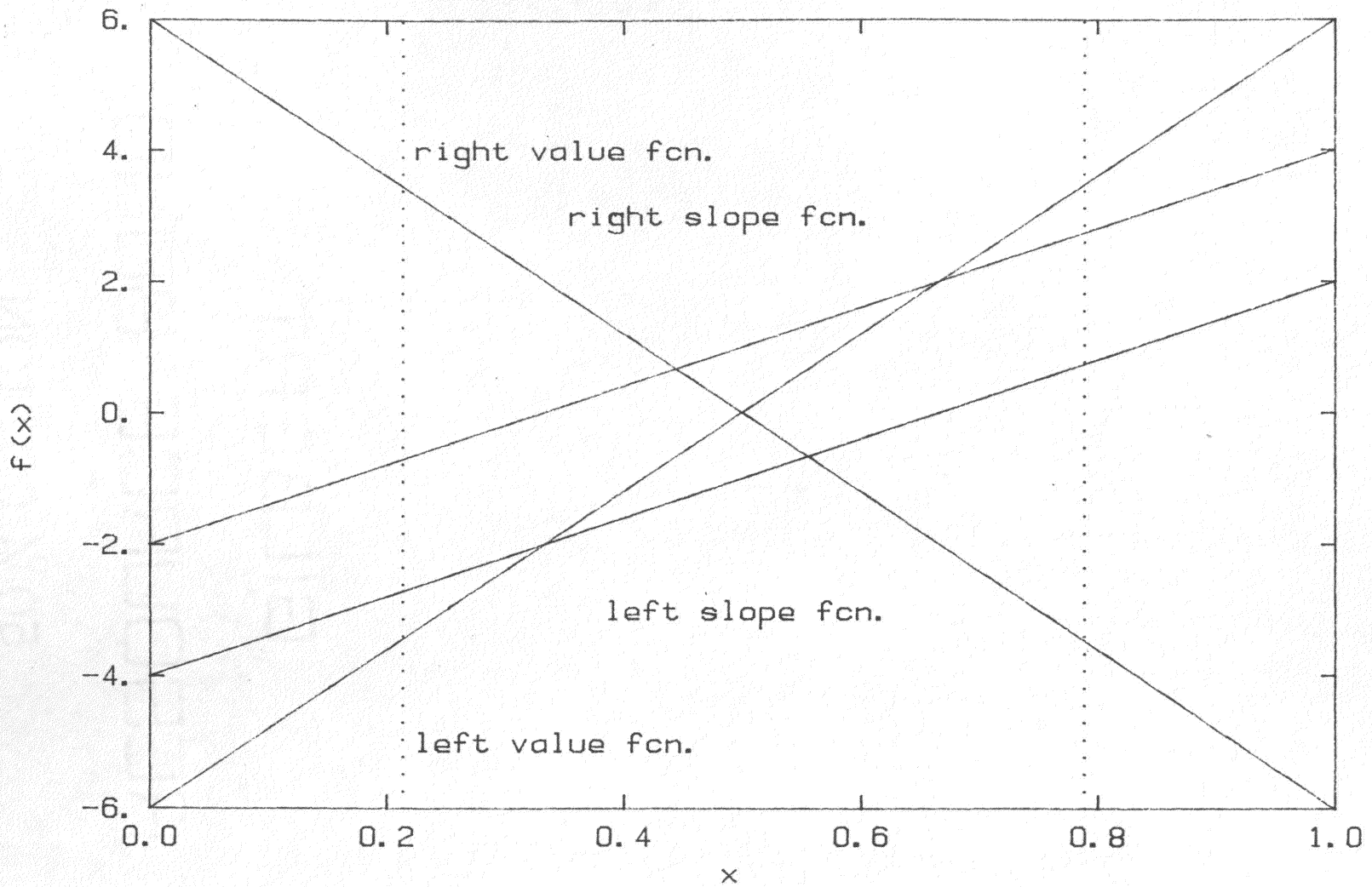
Figure 5.6 Second derivatives of the basis functions
supported over one breakpoint interval

(in the present electrochemical system, only the function values and second derivatives of the functions are needed). Figure 5.5 shows the first derivatives of the basis functions over an interval between two breakpoints, and Figure 5.6 shows the second derivatives. Note that any higher derivatives of the Hermite cubic splines are just constants.

Finally, since an uneven breakpoint distribution is allowed, what is the effect of changing the distance between breakpoints? Figures 5.7 to 5.9 provide the answers for two different breakpoint spacings: (1) the value function (Figure 5.7), (2) the first derivative of the value function (Figure 5.8), and (3) the second derivative of the value function (Figure 5.9). Note how the values at the Gauss points change. Figures 5.10 to 5.12 show the same graphs for the slope function.

1.b.i Calculate the value of the functions at the surface of the electrode

For the boundary condition at the electrode surface, the values of the spline functions at first breakpoint are required. The first rows of the matrices are concerned with satisfying the differential equation at this first breakpoint. The subroutine INTWTS (Listing 6), which is short for "INTerior basis function WeighTS", calculates the spline function values at any position between two breakpoints. This subroutine is taken from the routine of the same name by Dixon [1980]. The routine INTWTS is not as efficient or as precise as BASWTS, and so should not be used to replace BASWTS. The use of the subroutines BASWTS and INTWTS is illustrated by the program "SPLine TeST" of Listing 7; an example of the output of this program is shown in Listing 8.
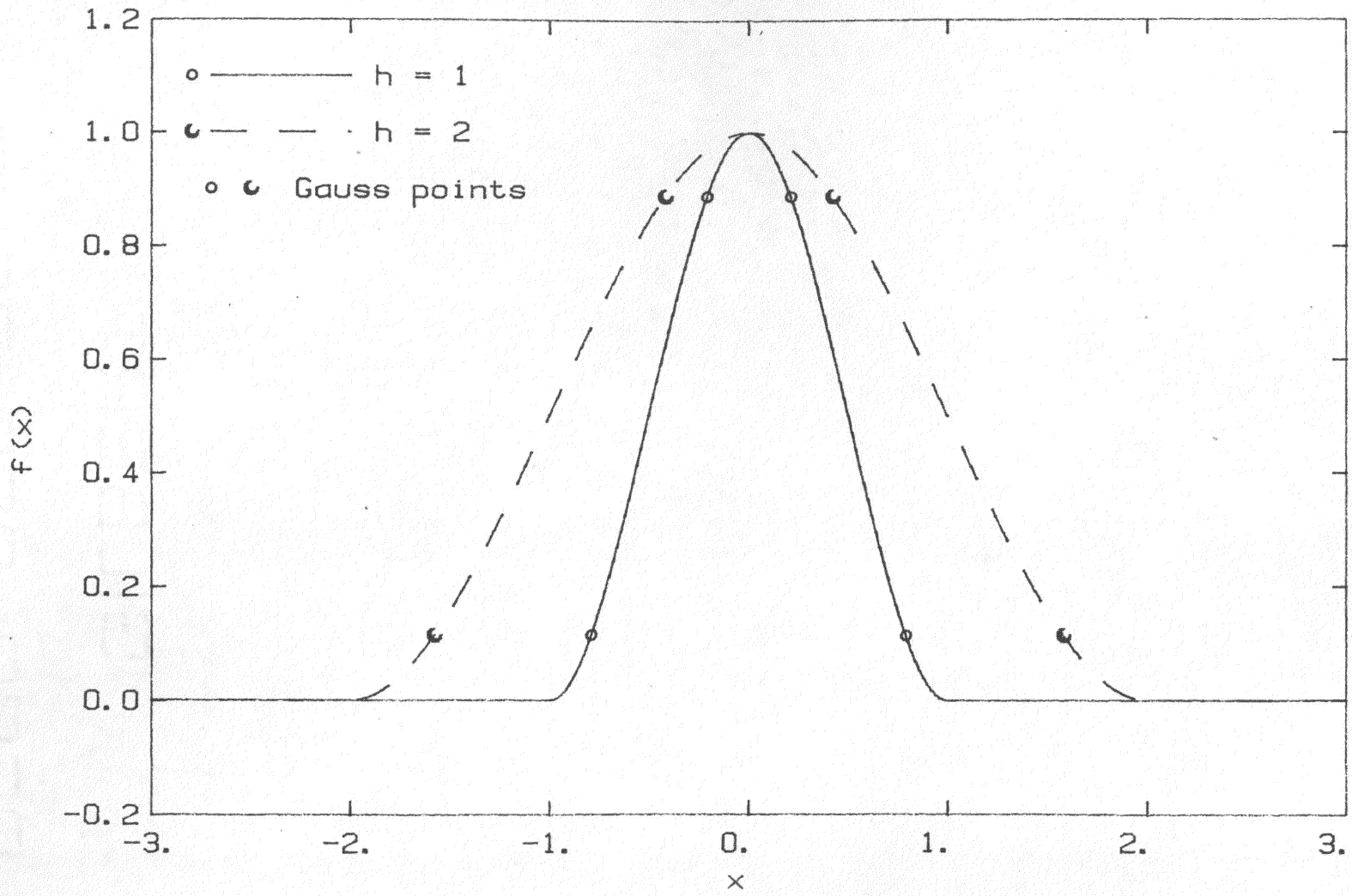
Figure 5.7 Hermite Cubic Spline, Value Function
(for two different mesh sizes)
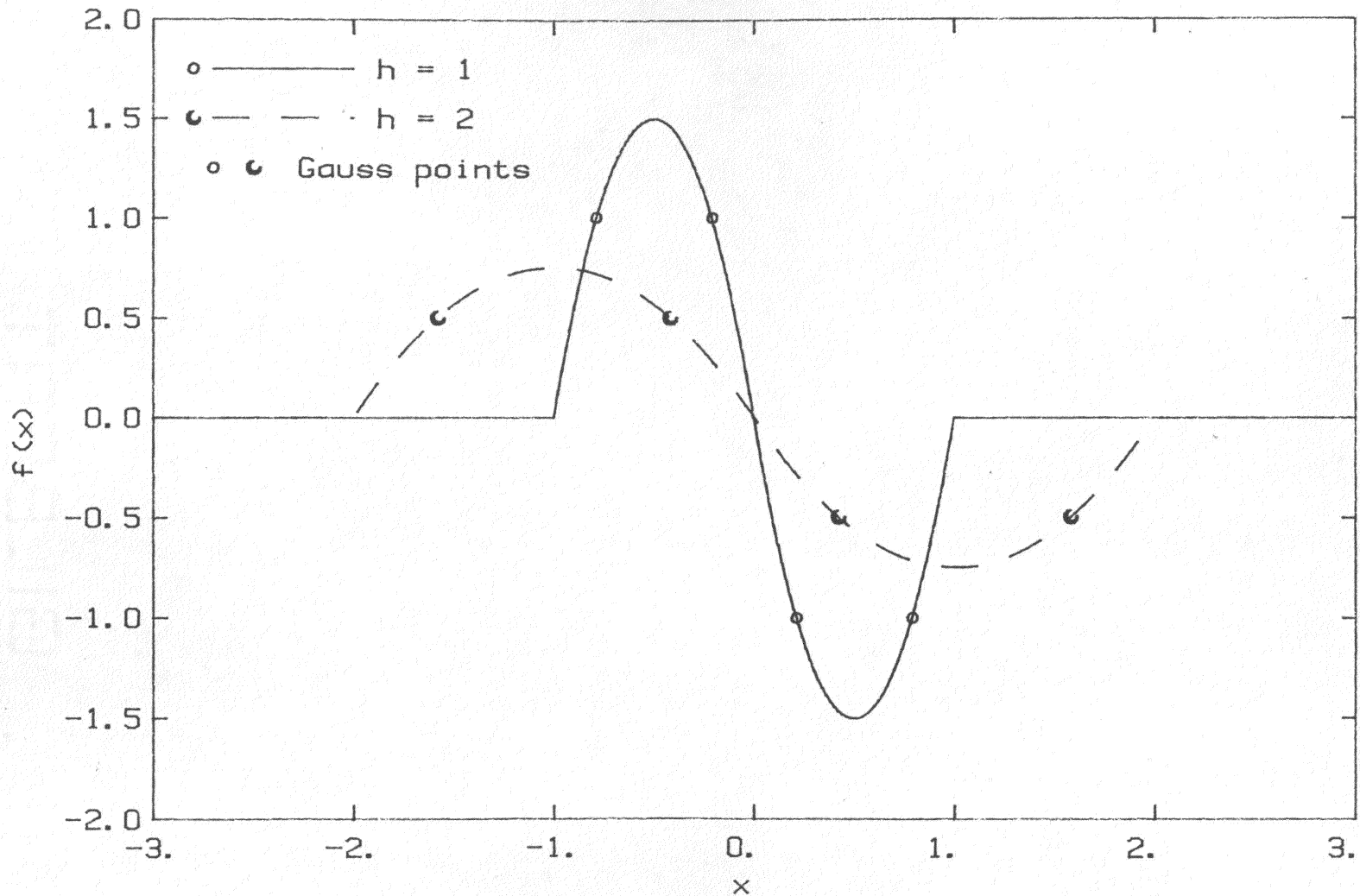
Figure 5.8 Hermite Cubic Splines, First Derivative of the Value Function
(for two different mesh sizes)
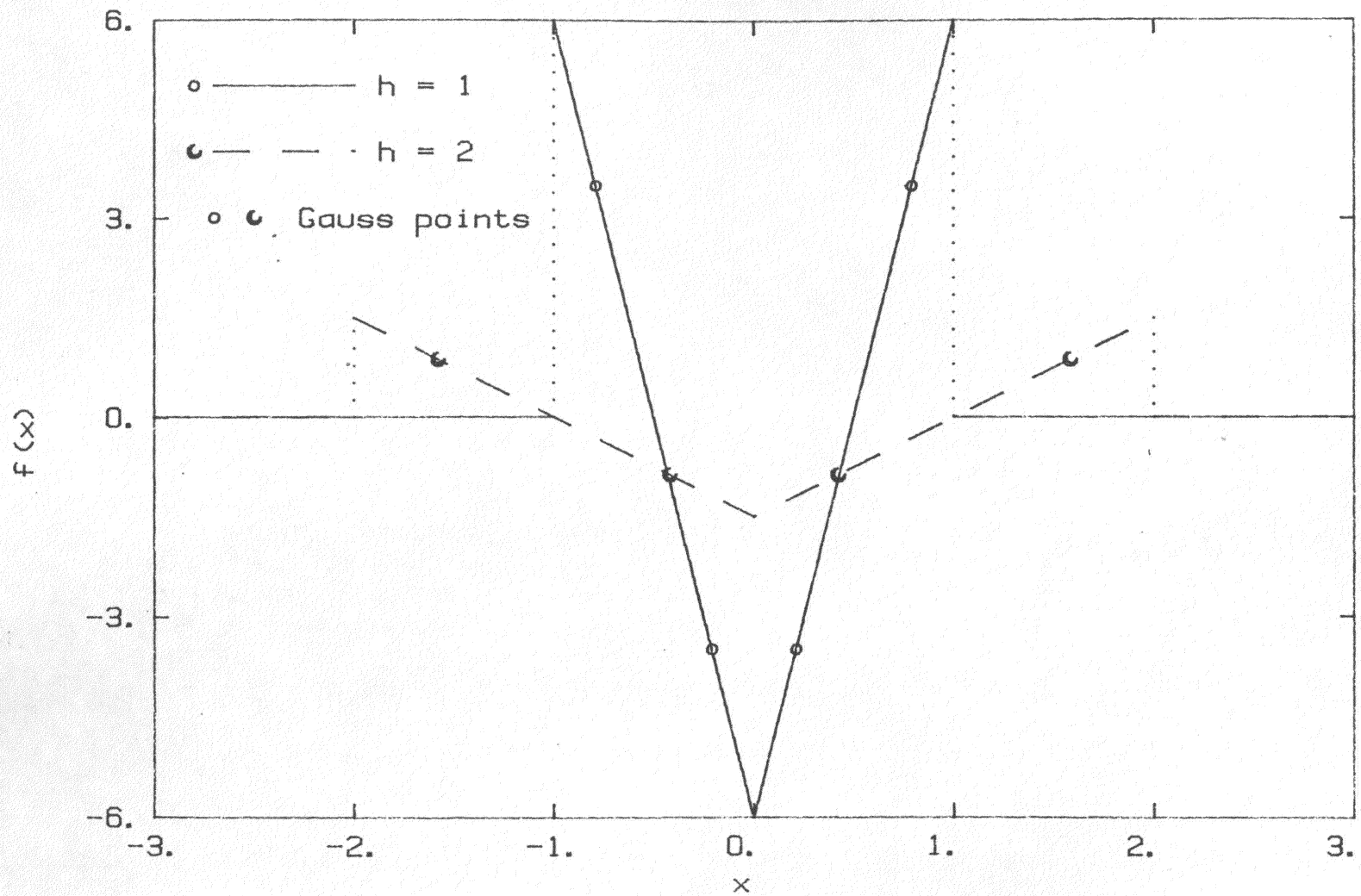
Figure 5.9 Hermite Cubic Splines, Second Derivative of the Value Function
(for two different mesh sizes)
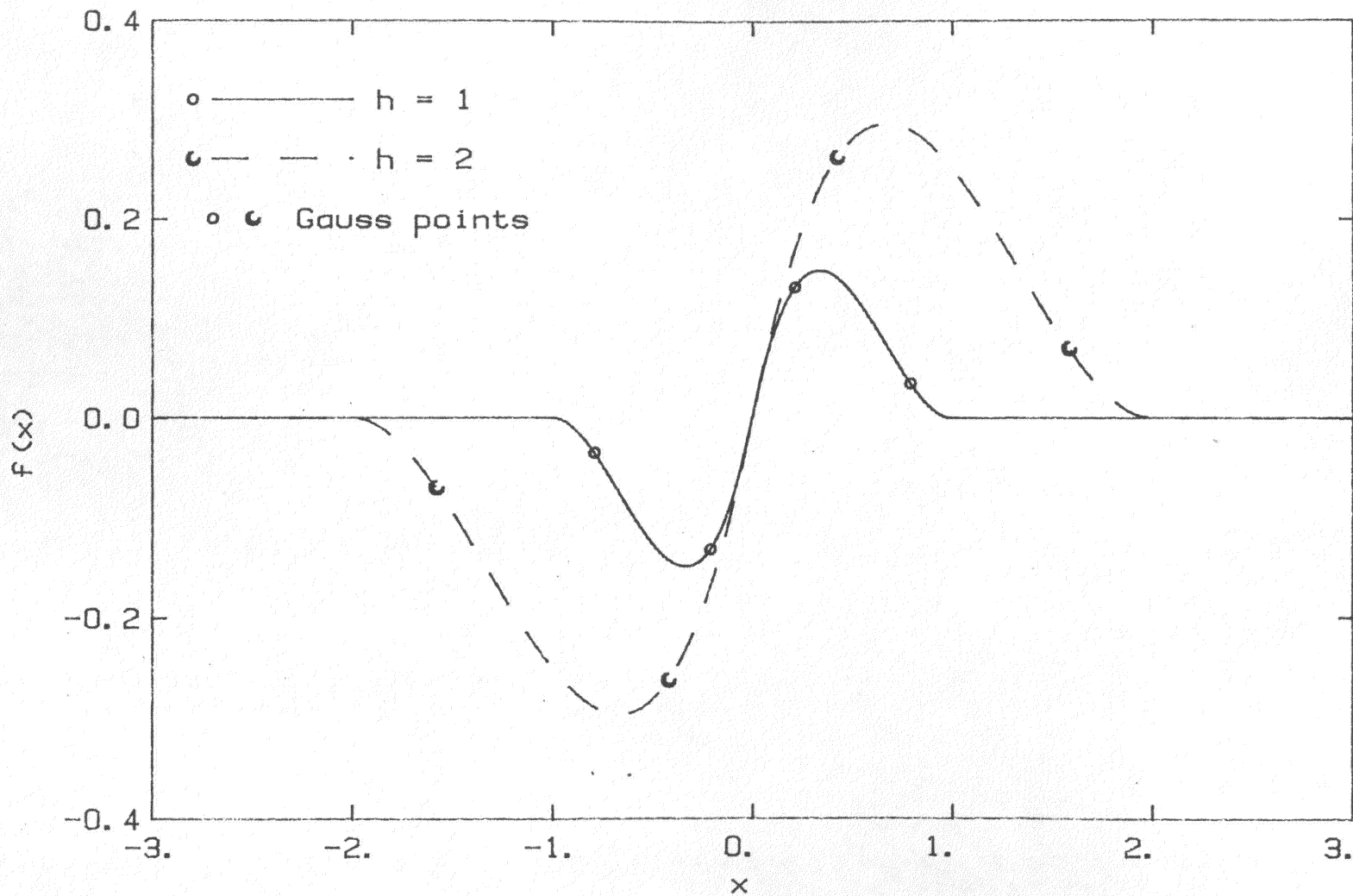
Figure 5.10 Hermite Cubic Splines, Slope Functions
(for two different mesh sizes)
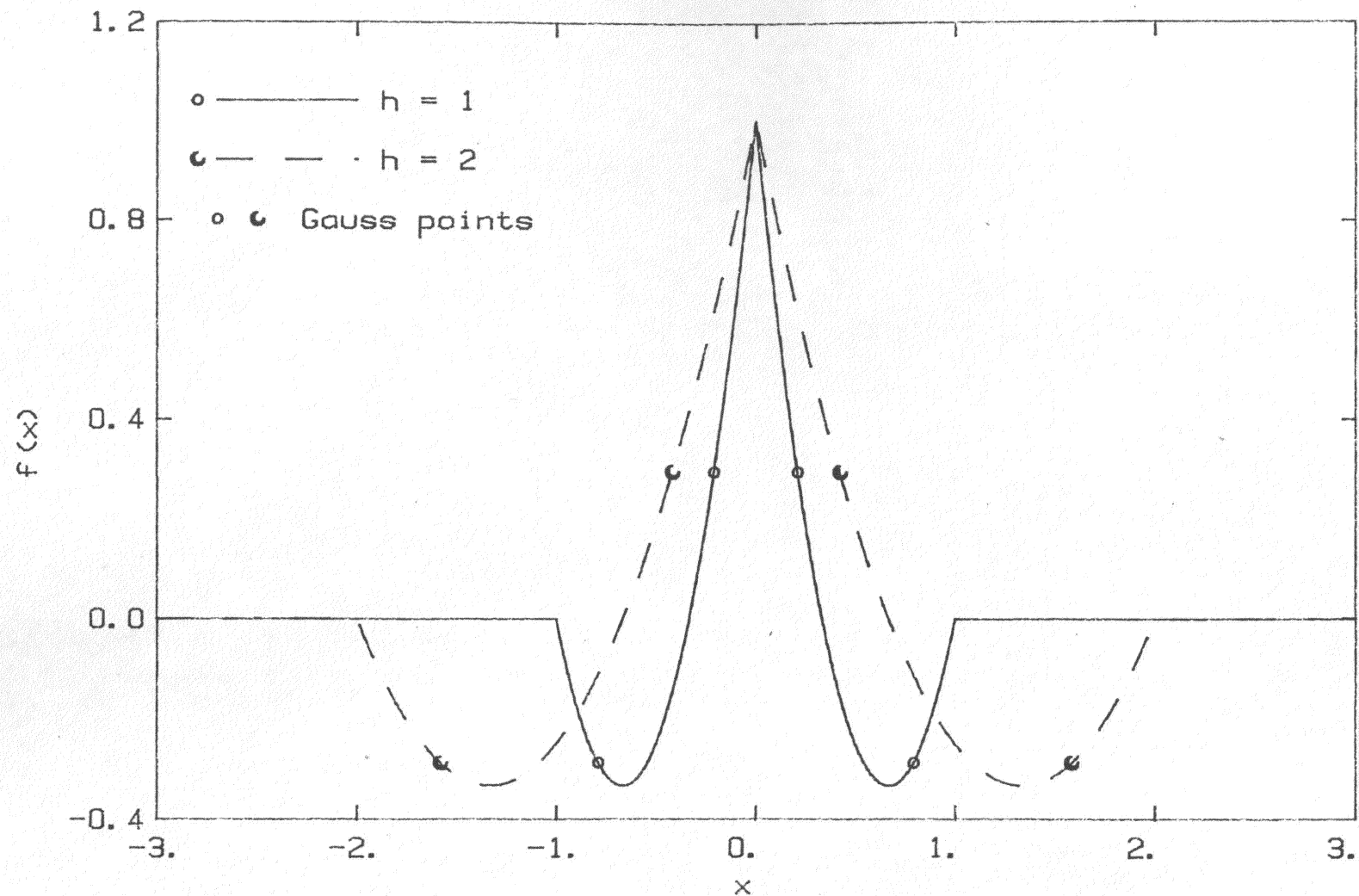
Figure 5.11   Hermite Cubic, Spline, First derivative of the Slope Function
(for two different mesh sizes)
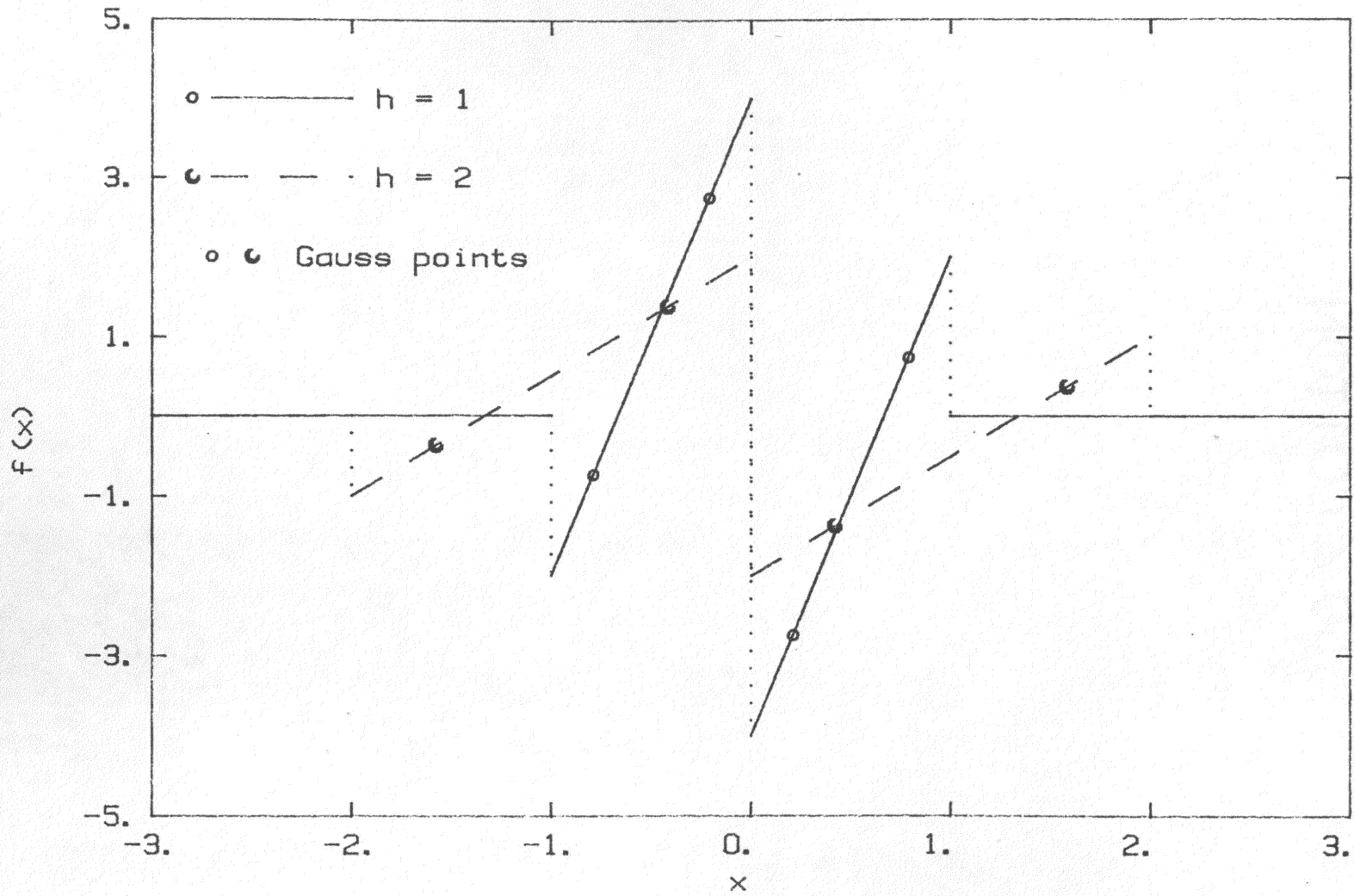
Figure 5.12 Hermite Cubic Spline, Second Derivative of the Slope Function
(for two different mesh sizes)

As mentioned in chapter 4, the usual way of incorporating a time dependent boundary condition is to differentiate the equation with respect to time. The boundary condition at the electrode surface was given in the previous chapter:

$$\alpha_2(\tau) = \alpha_1(\tau)\,[k_f(\tau) + k_b(\tau)] - k_b(\tau) \qquad\qquad [4.69]$$

The derivative with respect to time is:

$$\dot{\alpha}_2(\tau) = \dot{\alpha}_1(\tau)\,[k_f(\tau) + k_b(\tau)] + \alpha_1(\tau)\,[\dot{k}_f(\tau) + \dot{k}_b(\tau)] - \dot{k}_b(\tau) \quad [5.2]$$

Rearranging this into a form which follows that of equation 4.70:

$$\begin{bmatrix} -[k_f(\tau) + k_b(\tau)] & 1 \end{bmatrix} \begin{bmatrix} \dot{\alpha}_1(\tau) \\ \\ \dot{\alpha}_2(\tau) \end{bmatrix} = [\dot{k}_f(\tau) + \dot{k}_b(\tau)]\,\alpha_1(\tau) - \dot{k}_b(\tau)$$

$$[5.3]$$

Thus the matrix, **A**, would have some time dependent terms. Since equation 4.70 must be evaluated at every time step, if **A** changes with time, then it must be <u>decomposed</u> at every step. This a very time consuming operation. As an alternative, the boundary condition used at the electrode surface was to collocate at this point (the surface, d=0). That is, the differential equation is satisfied exactly at the boundary (at the first breakpoint). The elements of the matrix, **A**, are the spline function values at the collocation point; at d=0, the only non-zero spline function is the left value function, which is unity. The left hand side of equation 4.70 is then quite simple. The right hand side has the value of

the second derivatives and a generation term to take into account the electrochemical reaction:

$$\dot{\alpha}_1(\tau) = \left[\varphi_1(0)'' \quad \varphi_1(0)'' \quad \varphi_1(0)'' \quad \varphi_1(0)'' \right] \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \end{bmatrix} + \alpha_1 [k_f + k_b] - k_b \qquad [5.4]$$

The values for the second derivative of the spline functions are non-zero at the breakpoints. Four basis functions and their corresponding coefficients are involved in calculating the net second derivative at the first breakpoint. The values of these second derivatives are calculated by the routine INTWTS.

2.a Calculate $C\alpha$

After the matrices are filled with the appropriate values, the first step in evaluating the model is to multiply the banded matrix C by the coefficient vector α. While the banded structure of the A and C matrices in the model allow a more compact storage format, a special routine must be used for the multiplication. The routine BANMUL (Listing 9), which is short for "BANded matrix MULtiplication", multiplies a banded matrix by a vector. The author is responsible for this subroutine, and no claims for efficiency are made; however, the subroutine was written for relative ease of understanding. A program illustrating the use of this subroutine is given in Listing 10 and the output of the program in Listing 11.

2.b Calculate $C\alpha + D$

The elements of the vector, **D**, are time dependent and calculated in the subroutine QUASI (Listing 19), which is short for "QUASI-reversible electrochemical kinetics". A simple vector addition is also performed in this subroutine. This subroutine is by no means general purpose, and it must be read in the context of the program QUASELEC (Listing 20, to be described later).

2.c Calculate $d\alpha/dt = [C\alpha + D]/A$

Once the right hand side of the equation is evaluated, the left hand side, giving the rate of change of the coefficients, must be evaluated. The banded matrix, **A**, is decomposed into upper and lower triangular matrices by the subroutine BANDET (Listing 12), short for "BANded matrix DEcomposition - Triangular". This needs be done only when the values of the breakpoint change, as otherwise the elements of **A** are constant. The rate-of-change vector can then be evaluated by the subroutine BANSOL (Listing 13, short for "BANded matrix SOLver"), which solves the linear system, $A\alpha = b$, for α. Except for the correction of two small errors, the routines BANDET and BANSOL are based on the subroutines of the same name by Dixon [1980]. In turn, Dixon's routines are based on those in Martin and Wilkinson [1967]. A short program illustrating the use of BANDET and BANSOL is given in Listing 14, and the output of this test program is shown in Listing 15.

2.d Integrate $d\alpha/dt$

The vector of the rate of change of the coefficients can then be

integrated to find the final solution. This integration is done by the routine RUKUFE (Listing 16), which is short for "RUnge-KUtta-FEhlberg initial value solver". This routine is an implementation, by the author, of the fourth and fifth order Runge-Kutta-Fehlberg solver described by Davis [1984]. The chief features of this implementation are: estimation of the truncation error at each step and automatic adjustment of the step size to keep the truncation error within a given tolerance. The operation of the subroutine RUKUFE is demonstrated by the example program given in Listing 17, which solves example 5 of chapter 1 in the Davis [1984] text. The results of the example program are given in Listing 18.

2.e Start and stop simulation, report results

The "executive" routine is the main program QUASELEC (Listing 20), which is short for "QUASi-reversible ELECtrochemical model". This program was used to validate the mathematical method used. There are sections in this program for changing the type of simulation desired. The test used to validate the model was the simulation of a common electroanalytical technique, scanning voltammetry. In this technique, the voltage is scanned, with a constant rate of change, first in one direction and then in the reverse direction; Figure 5.13 shows a plot of the dimensionless potential as a function of time. The program called QUASELEC, Listing 10, was written to simulate scanning voltammetry. This model set up an initial guess for a vector of breakpoints, and used the subroutine FILL to calculate the values in the A and C matrices of the model.
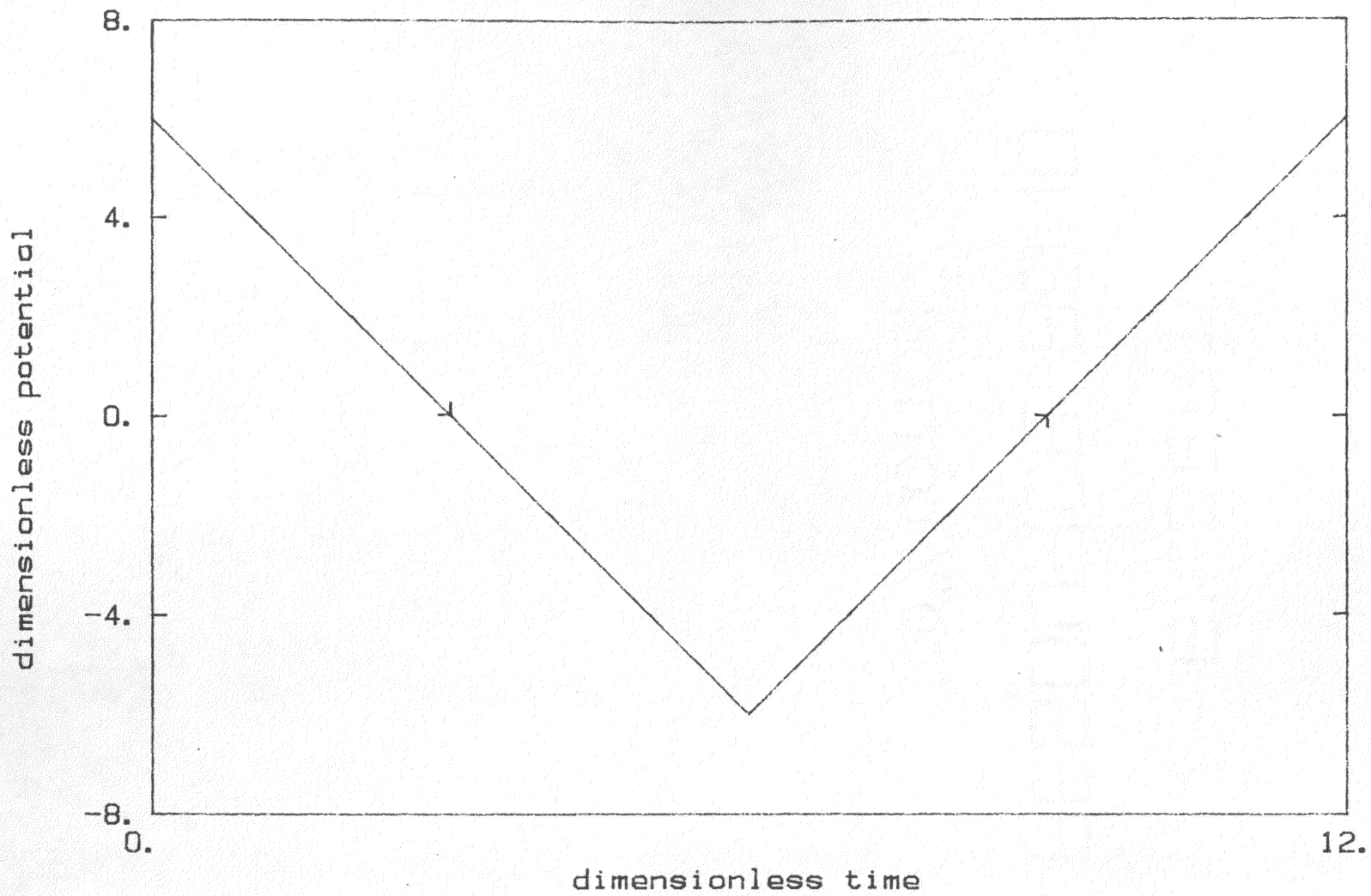
Figure 5.13 Applied potential versus time for scanning voltammetry

After setting up the initial conditions, the integrator subroutine RUKUFE was called to advance the solution by one time step. The dimensionless potential was a linear function of the dimensionless time. Two repeat loops (DO) were used, one for the forward potential scan and the other for the reverse potential scan. If needed, the subroutine REDIST was called to give an estimate of an improved vector of breakpoints; this was sometimes just noted for establishing an appropriate guess for the breakpoint vector, and sometimes the new estimate was used in the next calculation of the electrochemical response. The integrator subroutine used a subroutine, QUASI (Listing 19), which did the actual calculation of the rate of change of the coefficients by the spline collocation model explained above. The applied electrical potential (from equations 4.7 and 4.8) can be expressed in a dimensionless form:

$$P = \frac{nF}{RT} (E - E_0) \qquad [5.5]$$

where the constants (n, F, R, T, E_0) are defined after equations 4.6 and 4.8

The value of the slope (the concentration gradient) at the surface was taken as a measure of the dimensionless current flowing to the electrode,

$$\text{dimensionless current} = \frac{i}{n F A C_{A,bulk} k_0} = \frac{\partial f}{\partial d}\bigg|_{d=0} \qquad [5.6]$$

where the constants are as described after equations 4.6 and 4.8
    f is the dimensionless concentration
and d is dimensionless distance

The model used the previously described Butler-Volmer kinetics (equations 4.6 to 4.8).

Typical results from QUASELEC are shown in Figure 5.14. The reaction rate at the surface determines the flow of current to the electrode. The current depends on the rate constants and on the concentration of electroactive species at the surface of the electrode. There are two 'waves' on the current versus potential graph. One wave occurs during the scan from positive <u>to</u> negative dimensionless potentials; this wave is the *negative going* wave. This wave is also called the *cathodic* wave [Macdonald 1977] because the electrode acts as a cathode and species (species A in equation 4.1) are *reduced* (gain electrons and go from a higher oxidation state to a lower). The second wave occurs during the scan from negative <u>to</u> positive dimensionless potentials; this wave is the *positive going* wave. The positive going wave is also called the *anodic* wave because the electrode acts as an anode and species (species B in equation 4.1) are *oxidised* (lose electrons and go from a lower oxidation state to a higher). Several key points along the curve are marked in Figure 5.14; subsequent figures show the concentration profiles in the electrolyte corresponding to the marked points. The initial condition is shown in Figure 5.15; the breakpoints are marked as dotted vertical lines. The rate constants depend on the applied potential; as the potential becomes more negative, the rate of the forward reaction (A going to B) increases. This results in a consumption of species A at the surface as Figure 5.16 illustrates. As the potential is swept, the forward reaction rate constant (k_f in equation 4.7) increases while the
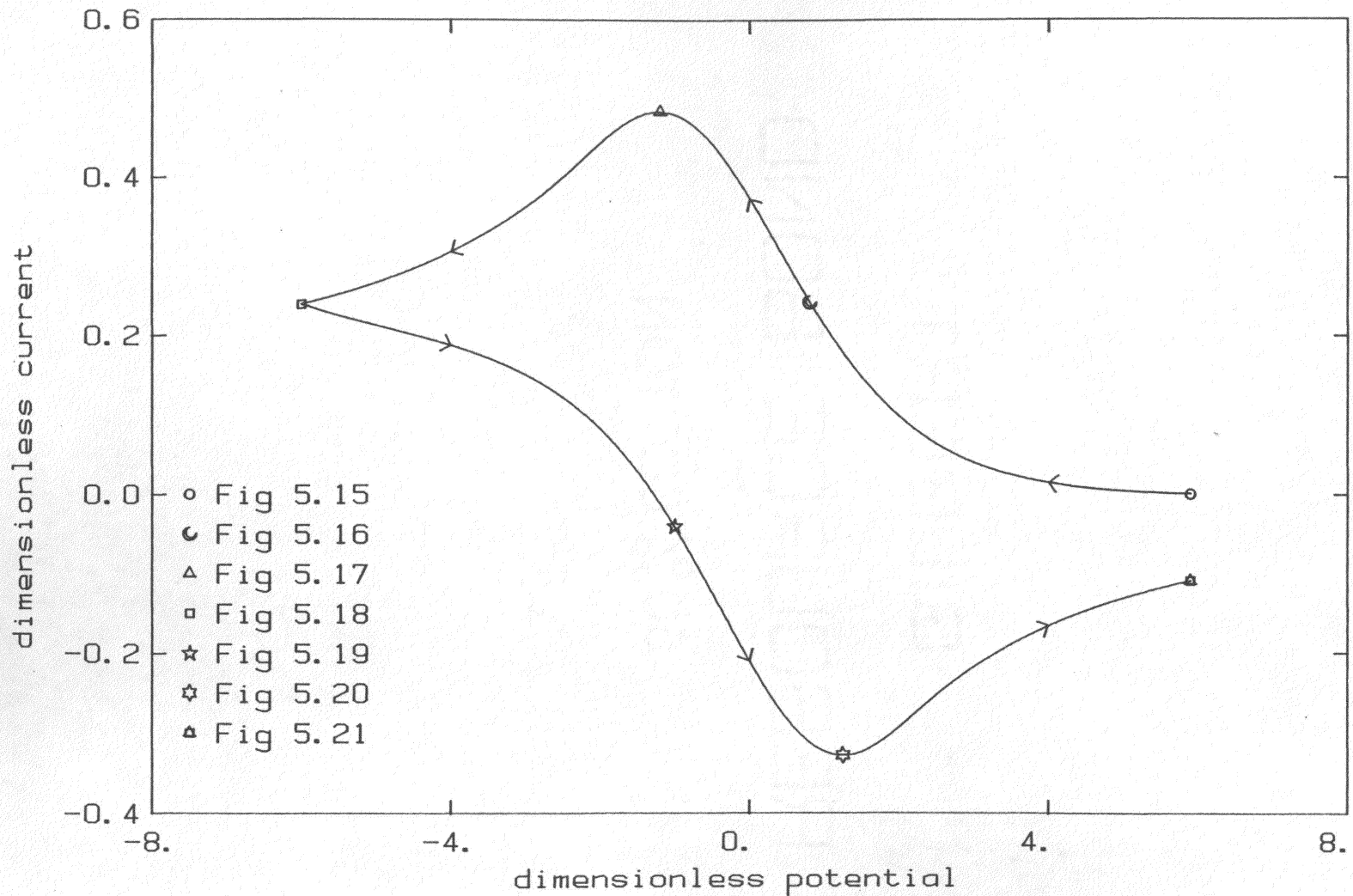
Figure 5.14   Current versus applied potential for scanning voltammetry
(points correspond to concentration profiles in following figures)
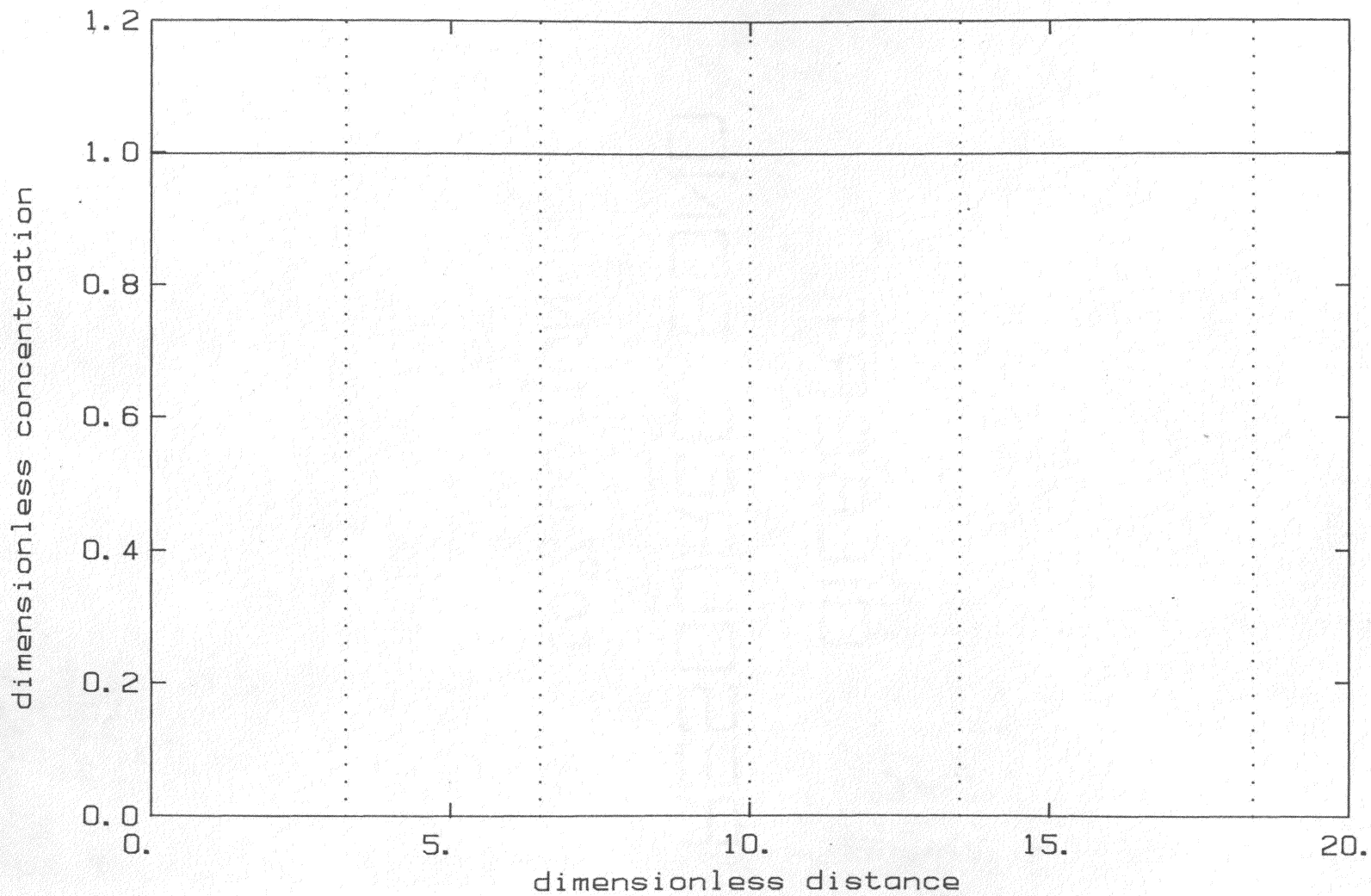
Figure 5.15 Initial Concentration Profile for Scanning Voltammetry.
(dotted lines are at the breakpoints) [Fig. 5.14: P=6.0, i=0.0]
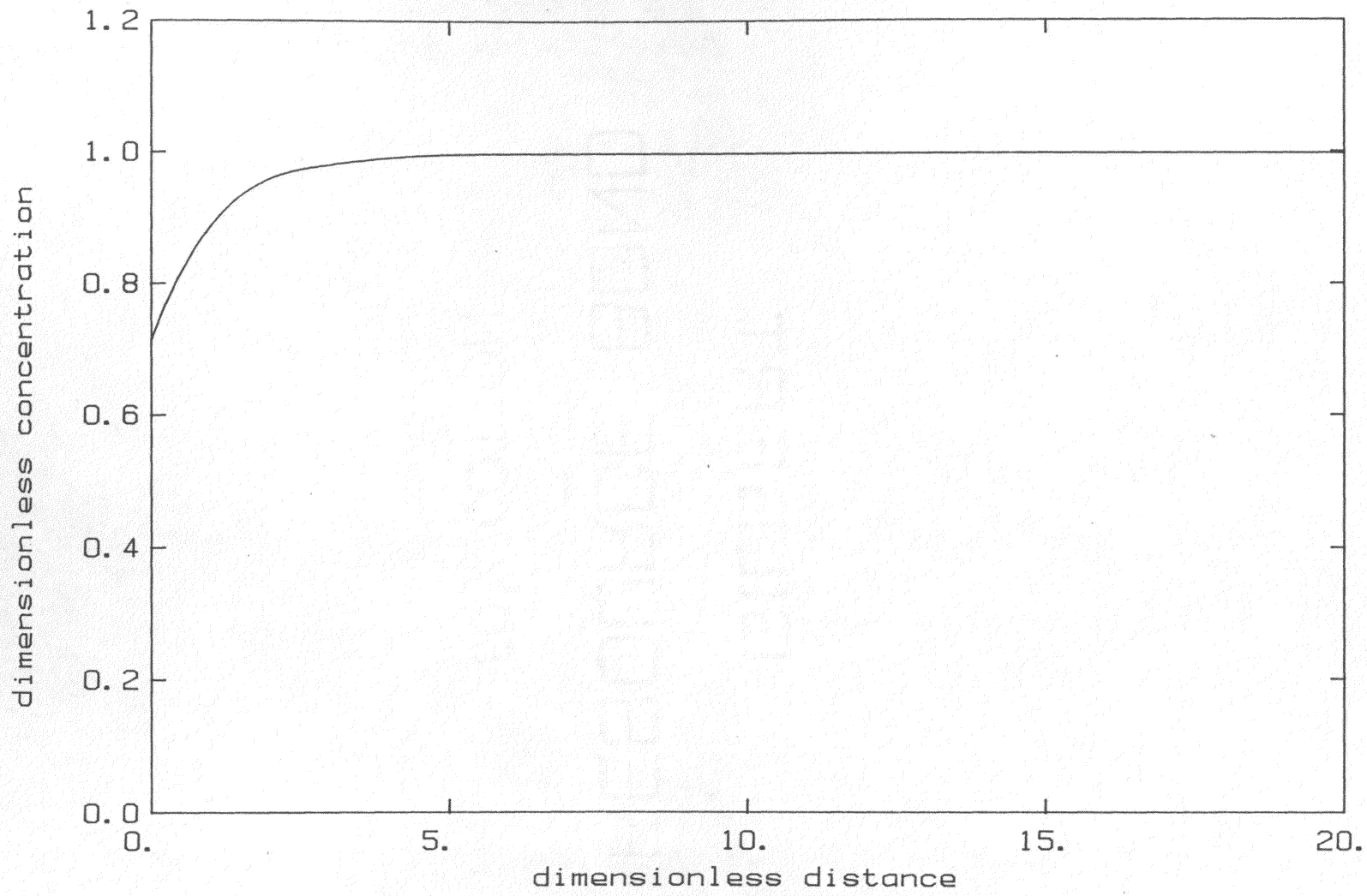
Figure 5.16 Concentration Profile for Scanning Voltammetry
(rising current) [Fig 5.14: P=0.8, i=0.24]

concentration of species A decreases. The net rate of reaction, which is a product of these two terms, passes through a maximum; thus the current also reaches a maximum value. Figure 5.17 shows the concentration profile at the maximum forward reaction rate. This maximum current is known by electrochemists as the diffusion limited current and results in a peak on a current (i) versus potential (E) graph. As the potential continues to decrease, the concentration at the surface of the electrode also decreases to a small value as seen in Figure 5.18.

After the direction of potential scan is reversed, the reverse reaction rate constant begins to rise, the concentration of species A at the surface of the electrode increases, and the current becomes negative (showing conversion of B into A). This point is shown in Figure 5.19. On the reverse sweep, as the concentration of B decreases (C_A+C_B=constant) and the reverse reaction rate constant increases, there is a minimum in the reaction rate (measured by the current). Figure 5.20 shows the concentration profile at this minimum. As the potential returns to the starting potential, the surface concentration reaches the starting concentration, but the final concentration profile (Figure 5.21) is different from the initial concentration profile, showing a net consumption of species A.

The above qualitative behavior (figures 5.14 to 5.21) of the model is consistent with other works in electrochemistry [Evans 1983, Macdonald 1977, Maloy 1983]. To verify the quantitative behavior of the model, its behavior was investigated at two extreme conditions. At the first extreme, the reaction rate is fast with respect to the potential sweep
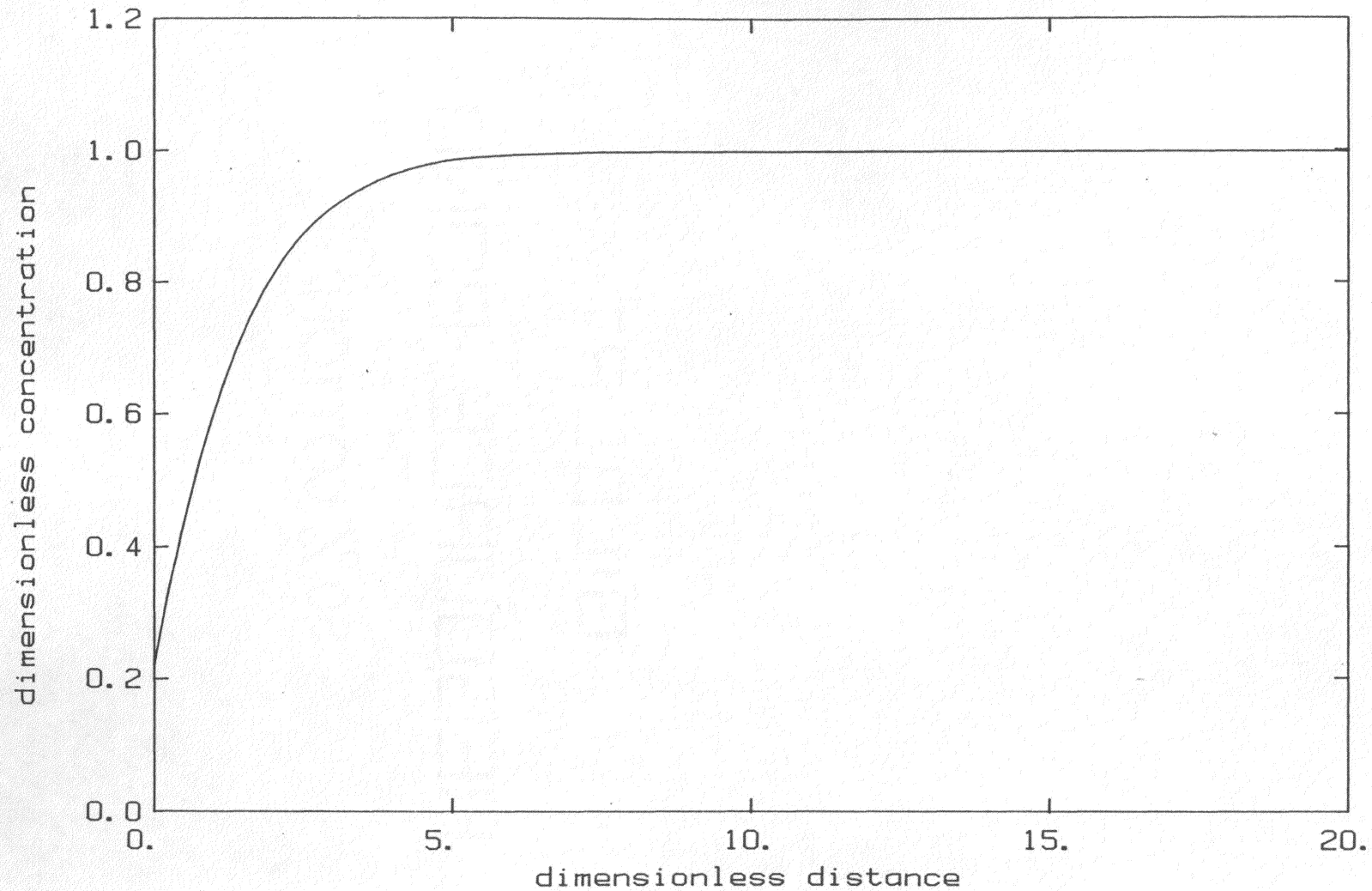
Figure 5.17 Concentration Profile for Scanning Voltammetry
(peak positive current) [Fig 5.14:   P=-1.2, i=0.48]
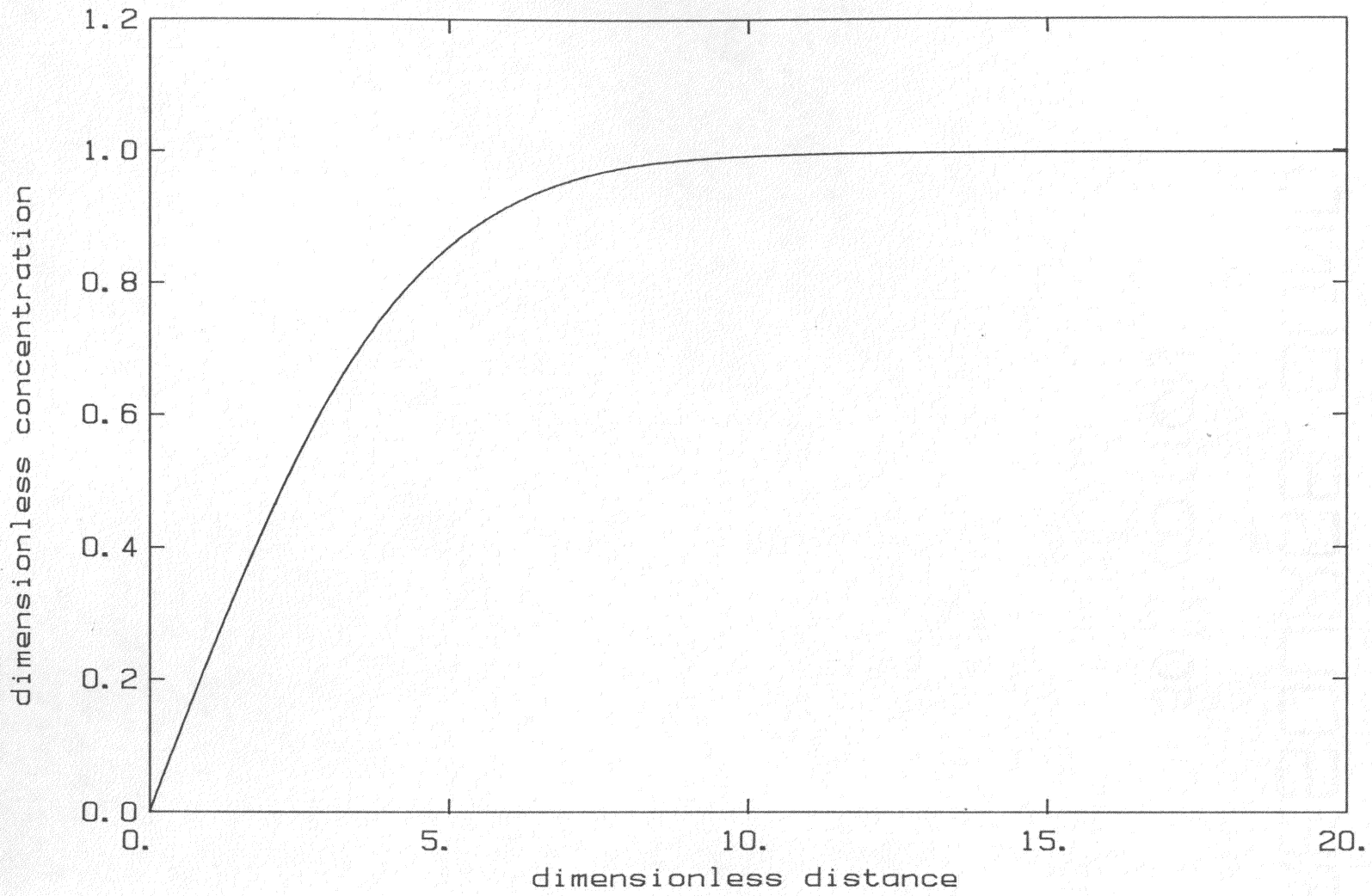
Figure 5.18   Concentration Profile for Scanning Voltammetry
(scan direction switch) [Fig5.14:   P=-6.0,   i=0.24]
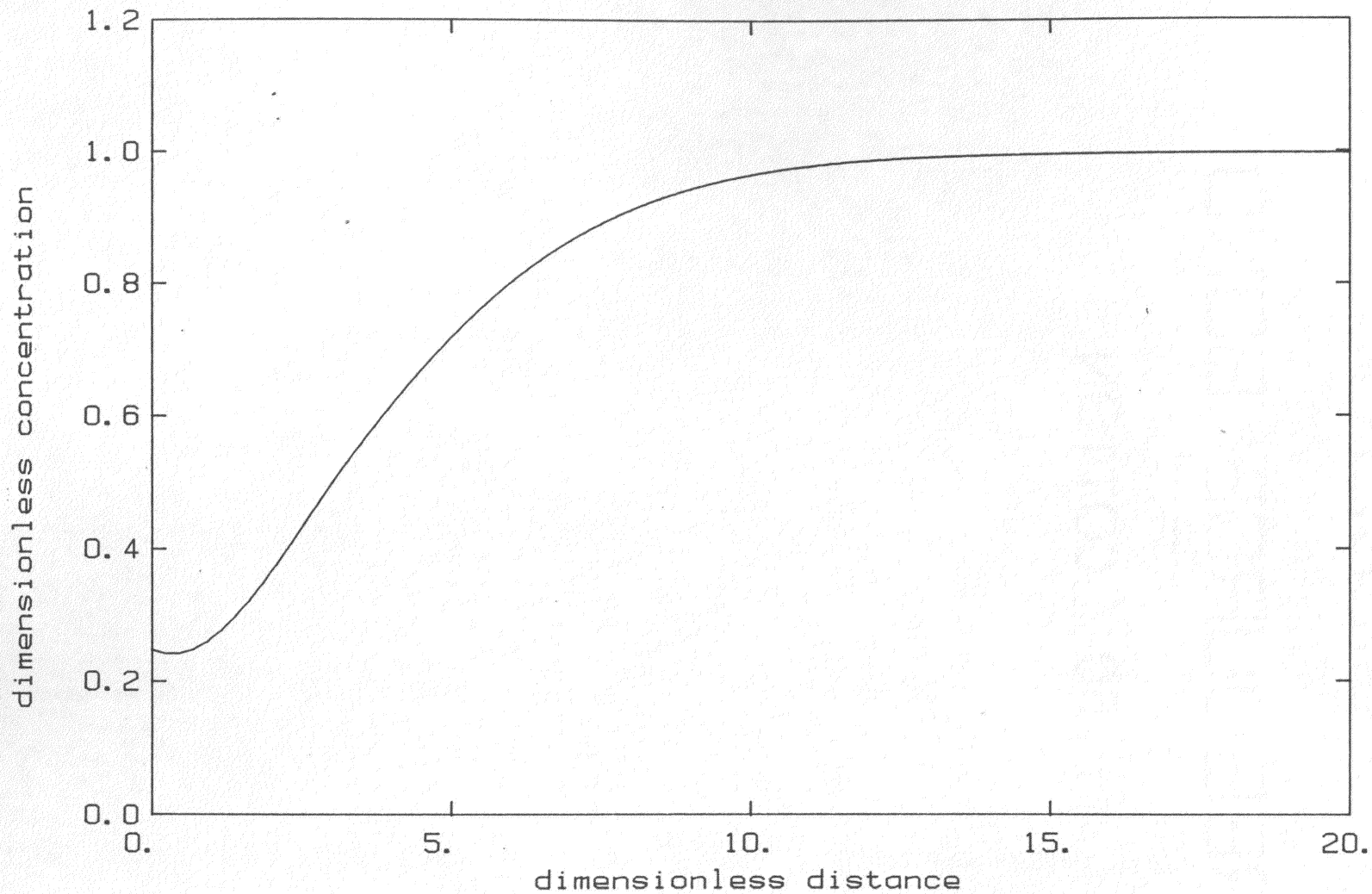
Figure 5.19 Concentration Profile for Scanning Voltammetry
(falling current) [Fig 5.14: P=-1.0, i=-0.04]
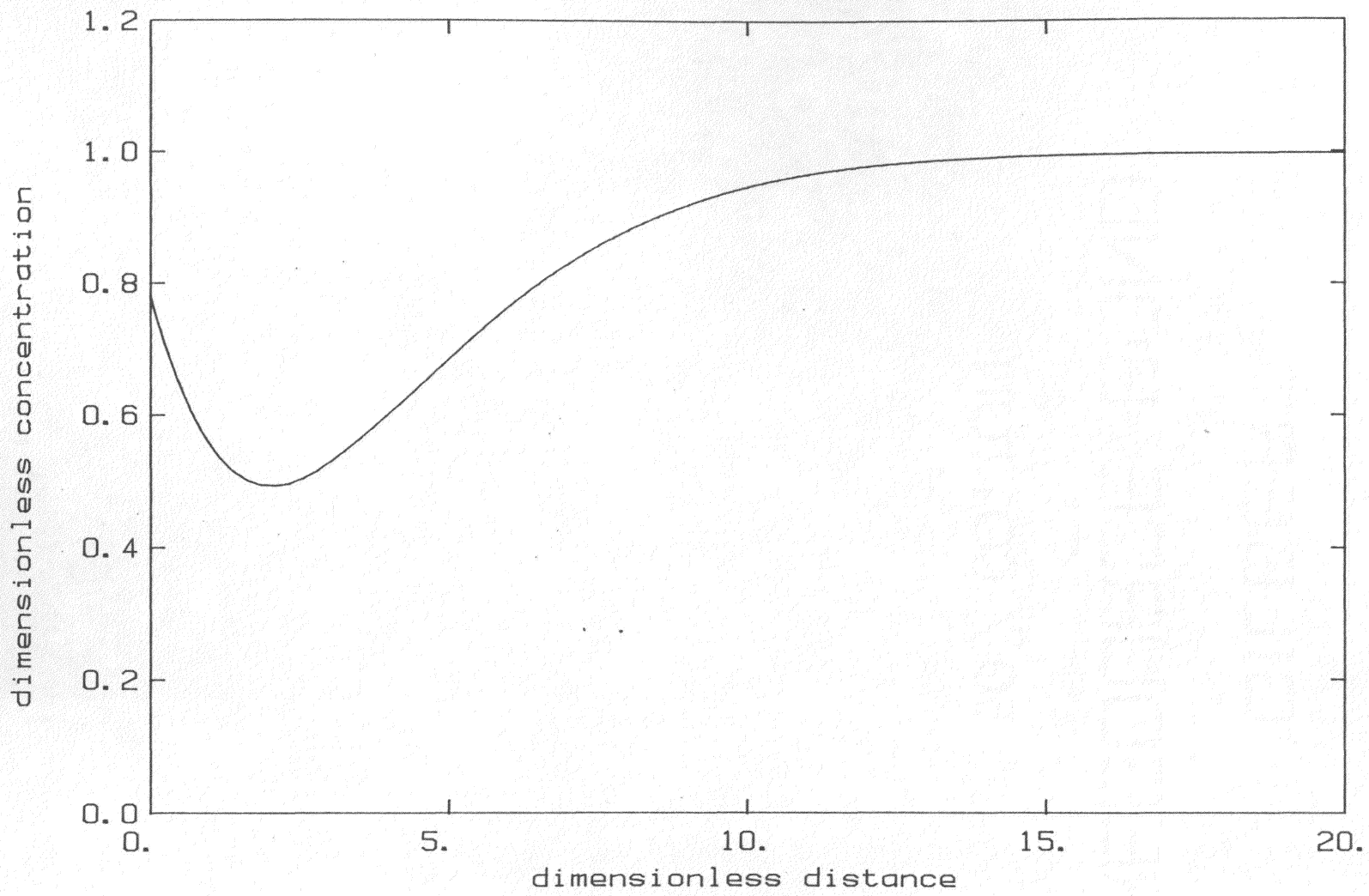
Figure 5.20   Concentration Profile for Scanning Voltammetry
(peak negative current) [Fig 5.14:   P=1.25,   i=-0.33]

169
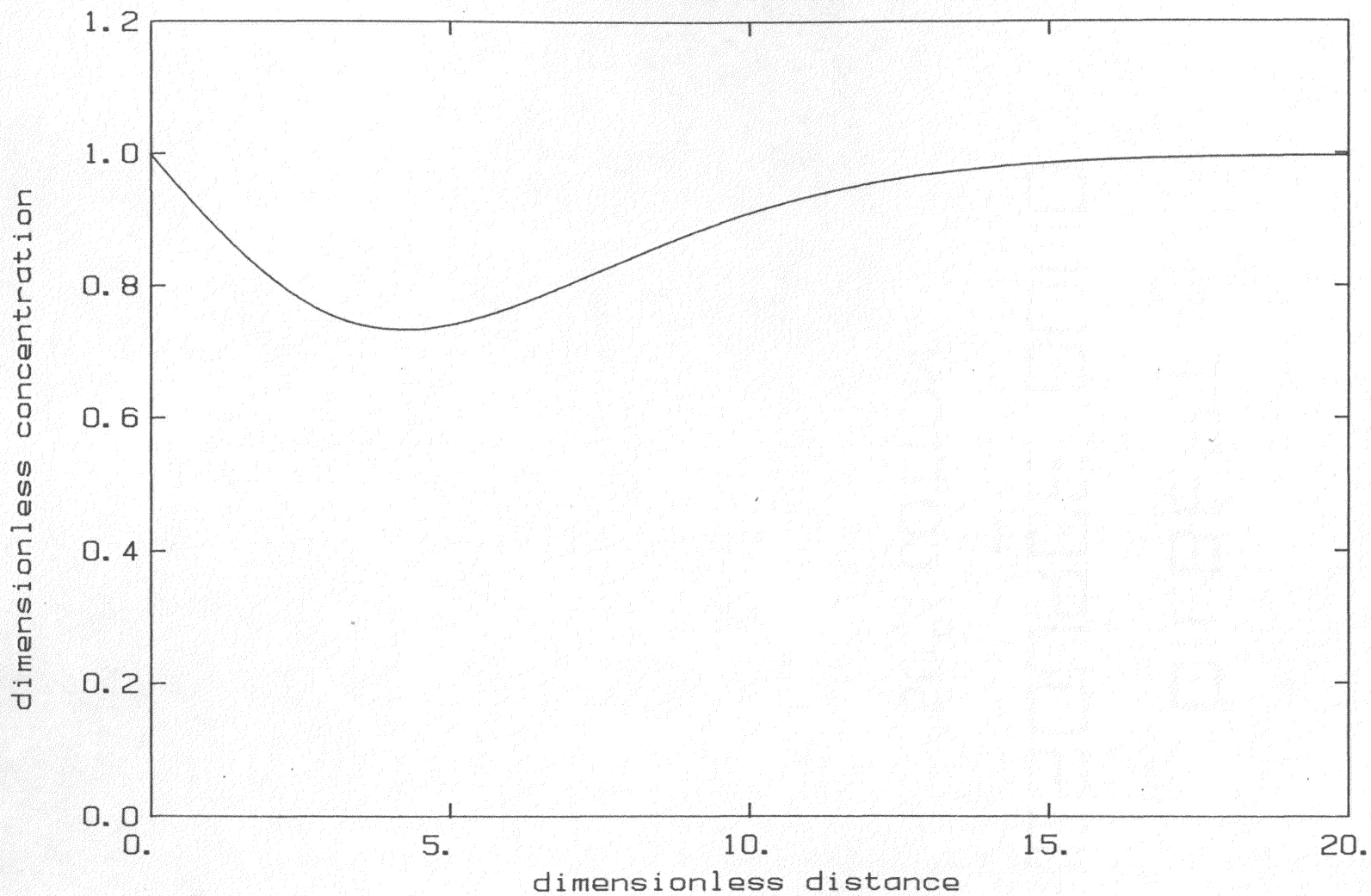
Figure 5.21   Concentration Profile for Scanning Voltammetry
(end of scan) [Fig 5.14:   P=6.0, i=-0.11]

rate. This condition embodies the assumption of equilibrium conditions at the electrode surface at all times. Electrochemical systems that meet this condition are called "reversible". At the second extreme, the reaction rate is slow compared to the potential sweep rate, and the assumption is made that the reverse reaction (B goes to A) never occurs at all. This condition is called "irreversible" by the electrochemists. Conditions in between these two extremes are called "quasi-reversible" by electrochemists.

Macdonald [1977] states: "In practice, the "reversibility" or "irreversibility" of a reaction is dependent upon the [potential] sweep rate, and a reaction that exhibits reversible behavior at low sweep rates will exhibit irreversible characteristics at high sweep rates." The mathematical model of chapter 4 is a model of a quasi-reversible electrochemical reaction, which can approach either extreme by varying the dimensionless potential scan rate:

$$\frac{dP}{d\tau} = \frac{nF\,\mathcal{D}}{RT\,k_0^2}\,\frac{dE}{dt} \qquad [5.7]$$

Where the constants are those defined after equations 4.2 and 4.8.

Values of $dP/d\tau$ approaching zero give reversible behavior, while values of $dP/d\tau$ approaching infinity give irreversible behavior (more quantitative statements will be made later). Three general modes of behavior can be seen: (1) reversible, (2) quasi-reversible, and (3) irreversible. These three modes are illustrated in figures 5.22, 5.23, and 5.24 respectively. Note that the peak heights and peak positions change with the scan rate, and the shape of the curves changes somewhat as well.
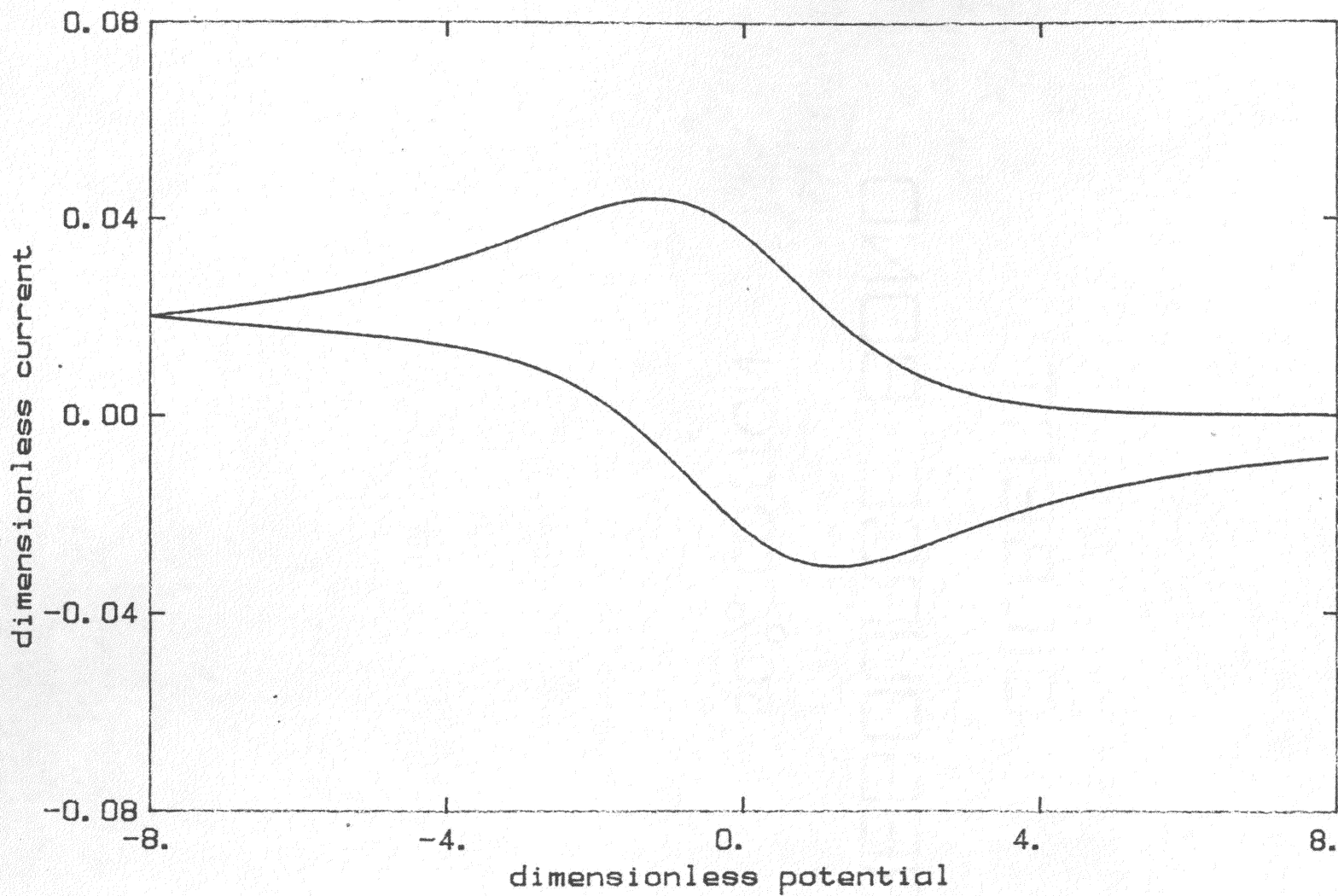
Figure 5.22 Scanning Voltammetry, "Reversible"
(dimensionless scan rate = 0.01)
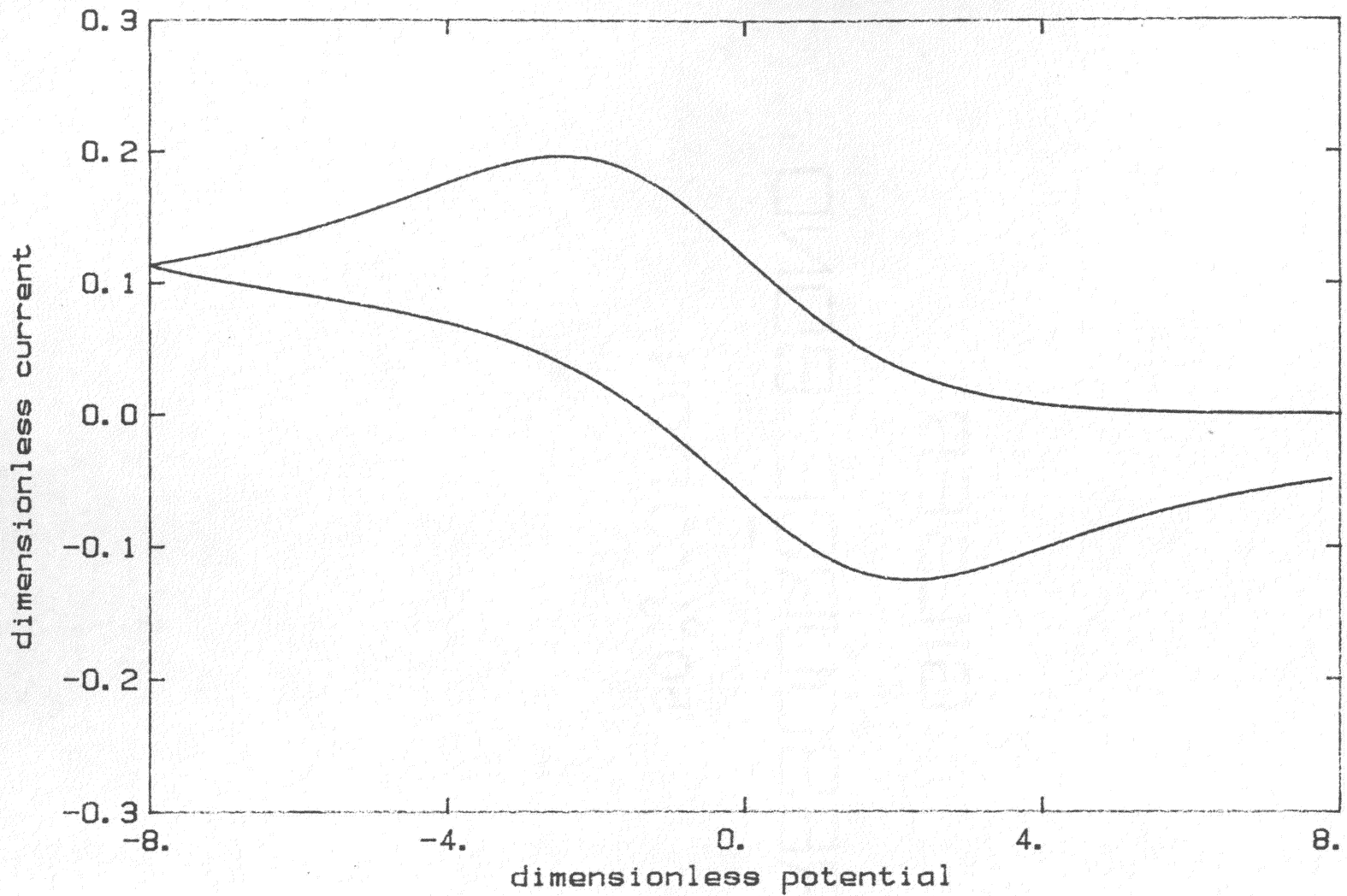
Figure 5.23 Scanning Voltammetry, "Quasi-reversible"
(dimensionless scan rate = 0.25)
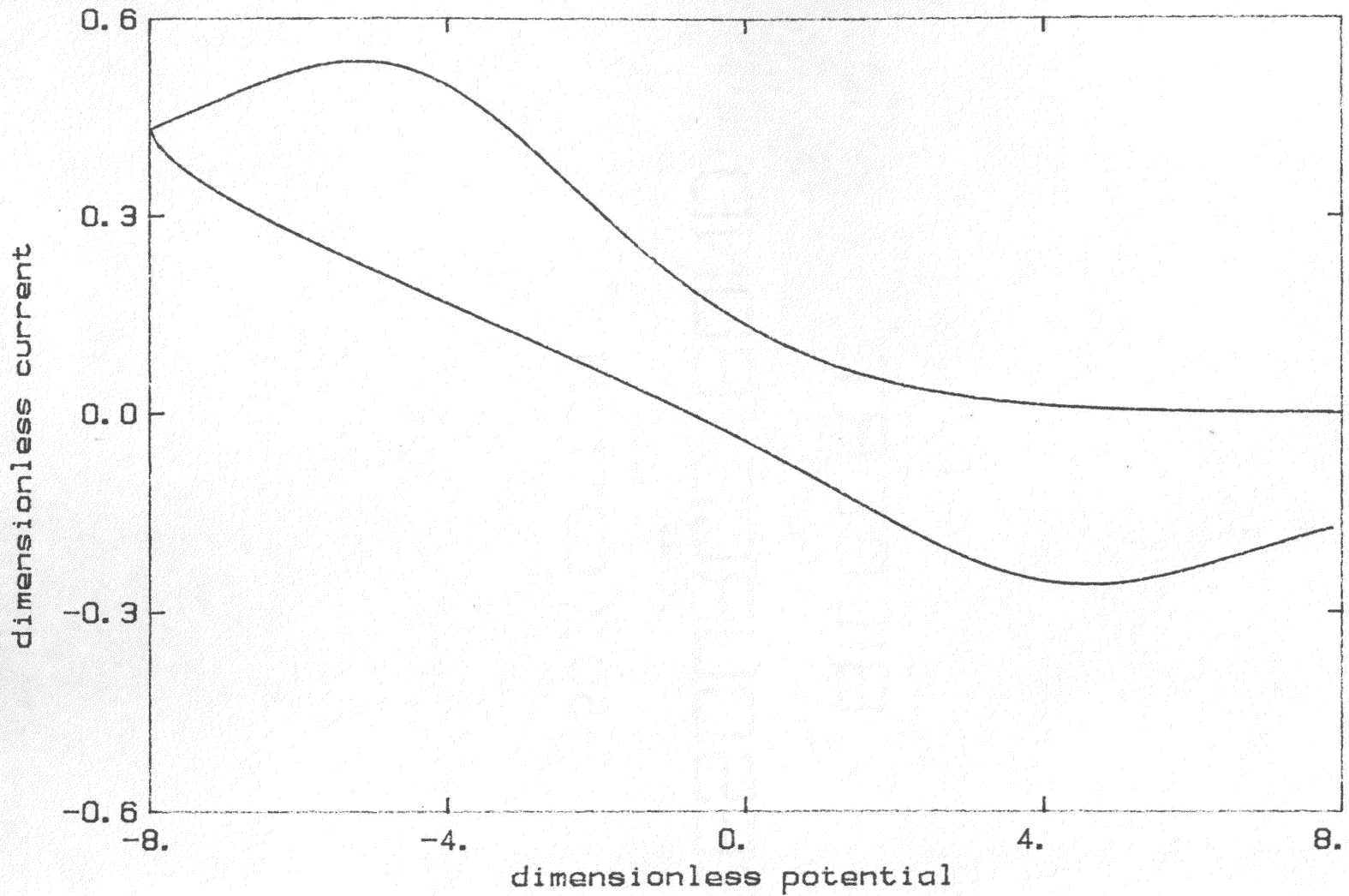
Figure 5.24   Scanning Voltammetry, "Irreversible"
(dimensionless scan rate = 2.25)

The first quantitative test is in the reversible kinetics regime; the dimensionless potential scan rate is set very low so that the reaction rate is much larger than the scan rate. Macdonald [1977] restates the results of Nicholson and Shain [1964], who showed several characteristics that can be used to determine the quality of the simulation. Figure 5.25 shows the separation of the peaks from dimensionless potential of zero. The figure shows good agreement with the results of Nicholson [Tables I and II, 1964]. For the negative going (cathodic) wave, the peak current is at a dimensionless potential of 1.109 (Nicholson) compared to 1.110 for the present work. And for the positive going (anodic) wave, the minimum current is at a dimensionless potential of 1.19 (Nicholson) compared to 1.18 for the present work. Under reversible reaction conditions, the solution requires quite a small time step-size, partly as a result of the rate of reaction varying by more than a factor of 400 in going from the maximum to the minimum potential (this difficulty can be reduced by using what are called "stiff" solvers such as Gear's method; see Davis [1984]). A second test of the simulation in the reversible regime is shown in Figure 5.26, which shows the symmetry of the peak heights. The cathodic peak height is measured from the initial current of zero. The anodic peak height is measured from an extension of the cathodic wave such that the total elapsed time from the starting point is the same for both the cathodic reference curve and the anodic minimum current. The values of 0.0631 (Nicholson) and 0.0630 (present work) compare well. Thus the mathematical model solved via spline collocation is reasonably precise for reversible conditions.
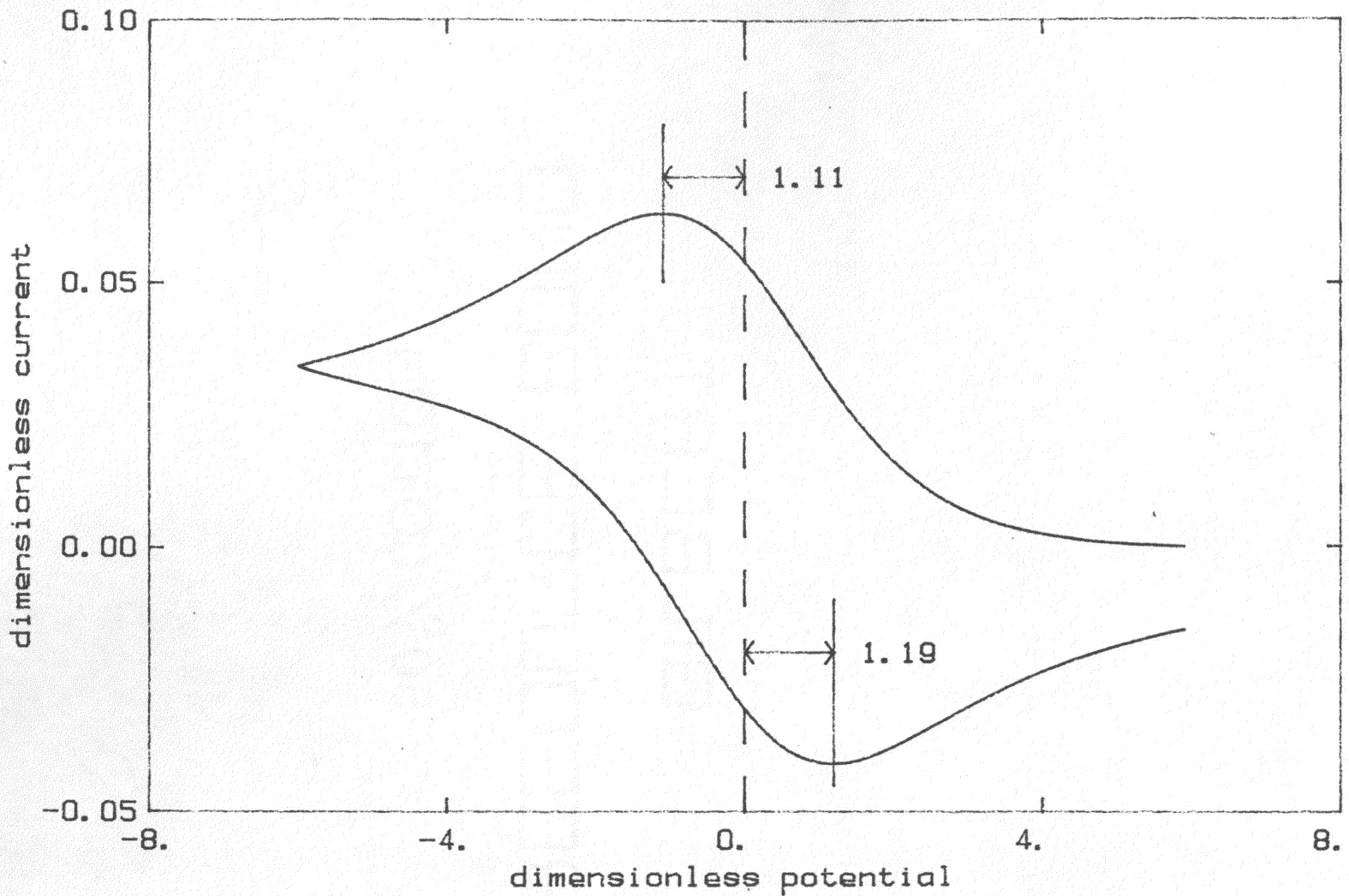
Figure 5.25 Current versus potential under "reversible" conditions showing peak position (dimensionless scan rate = 0.02)
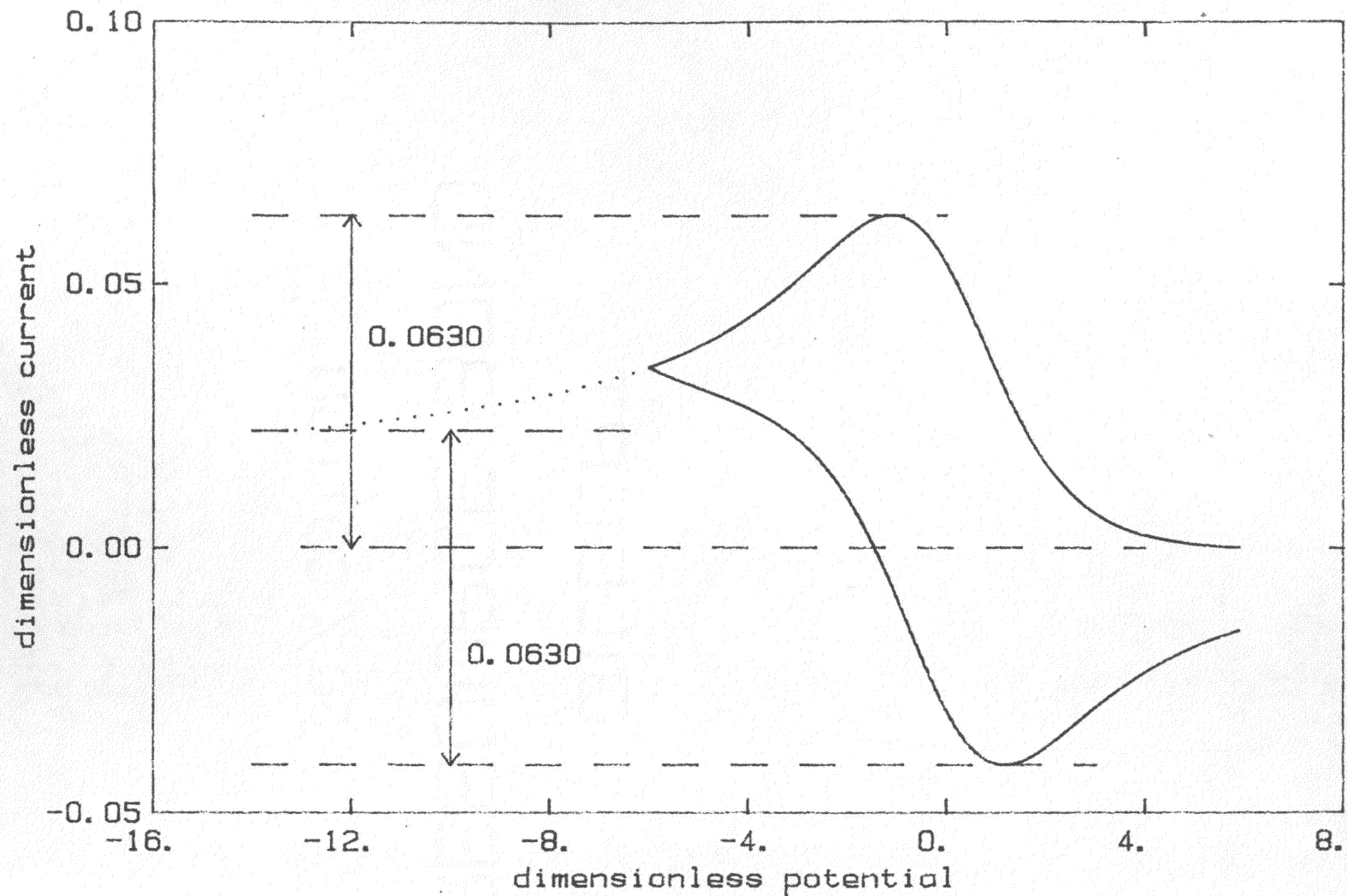
Figure 5.26 Current versus potential under "reversible" conditions
showing peak heights (dimensionless scan rate = 0.02)

Macdonald [1977] also showed that, for the extremes of reversible and irreversible conditions, there are asymptotic functions for the peak height as a function of the square root of the scan rate. Equations 6.32 and 6.44 of that work, which are derived from equations 25 and 47 of Nicholson [1964], can be converted to the present work's notation. The peak dimensionless currents under reversible and irreversible conditions are:

$$i_{peak, rev} = 0.4463 \left[\frac{dP}{d\tau} \right]^{1/2} \qquad [5.8]$$

$$i_{peak, irr} = 0.4958 \, \beta^{1/2} \left[\frac{dP}{d\tau} \right]^{1/2} \qquad [5.9]$$

where β is the 'charge transfer coefficient' from equations 4.7 and 4.8

As the dimensionless potential scan rate increases, there should be a transition from the reversible regime to the regime of irreversible kinetics. The peak currents are linear with respect to the square root of the potential scan rate, and the difference between the two kinetic regimes is only a constant of proportionality. Macdonald [1977] illustrated the transition from reversible to irreversible behavior in his Figure 6.4 which shows peak current as a function of the square root of the potential scan rate. Figure 5.27 shows the results of the simulation the first time the peak heights were calculated as a function of scan rate. The dashed lines indicate the two asymptotic extremes of the quasi-reversible model, as calculated by Nicholson and Shain [Nicholson 1964, Macdonald 1977]. The model did not follow the desired

characteristics at all. The boundary condition used for the concentration at the surface of the electrode was suspected.

Since derivatives of the basis functions may be discontinuous at the breakpoints, the values used in this first simulation are those of the basis functions to the right of the breakpoint, which is the standard convention [equation 3.16 Davis 1984]. At the central breakpoint, the second derivative of the "value" basis function is continuous (see Figure 5.9), while the second derivative of the "slope" basis function is discontinuous and changes sign (see Figure 5.12). This suggested a re-evaluation of the values used for the basis functions at the surface. Intuitively, it seemed that the "slope" function should go through an inflection point at the central breakpoint; that is the second derivative should be equal to zero (see Figure 5.10). Similarly, any other "slope" function appears flat at the outlying breakpoints, which would mean a second derivative of zero.
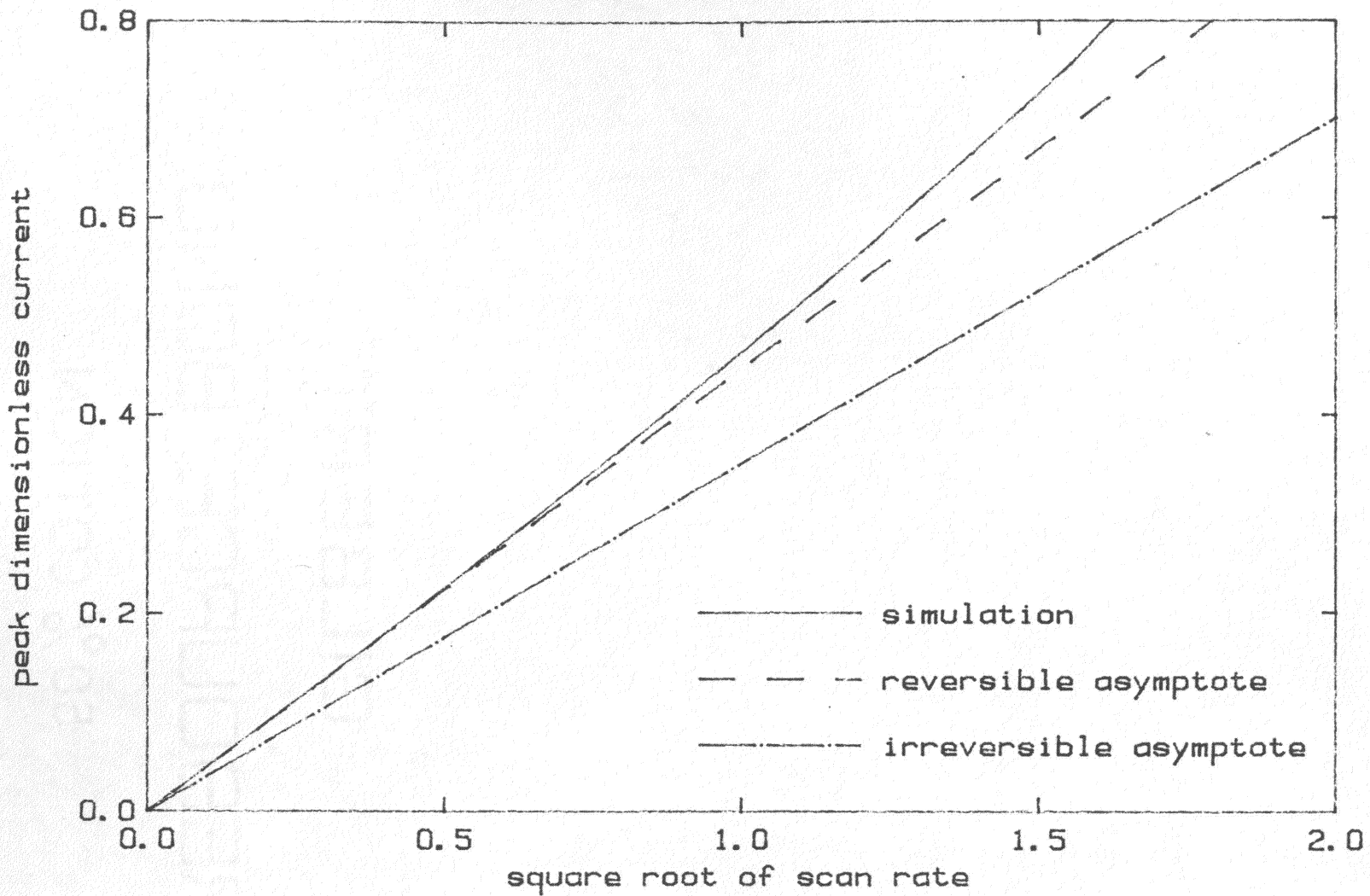
Figure 5.27 Transition from 'reversible' to 'irreversible' kinetics
(values for the splines at the breakpoint by normal convention)
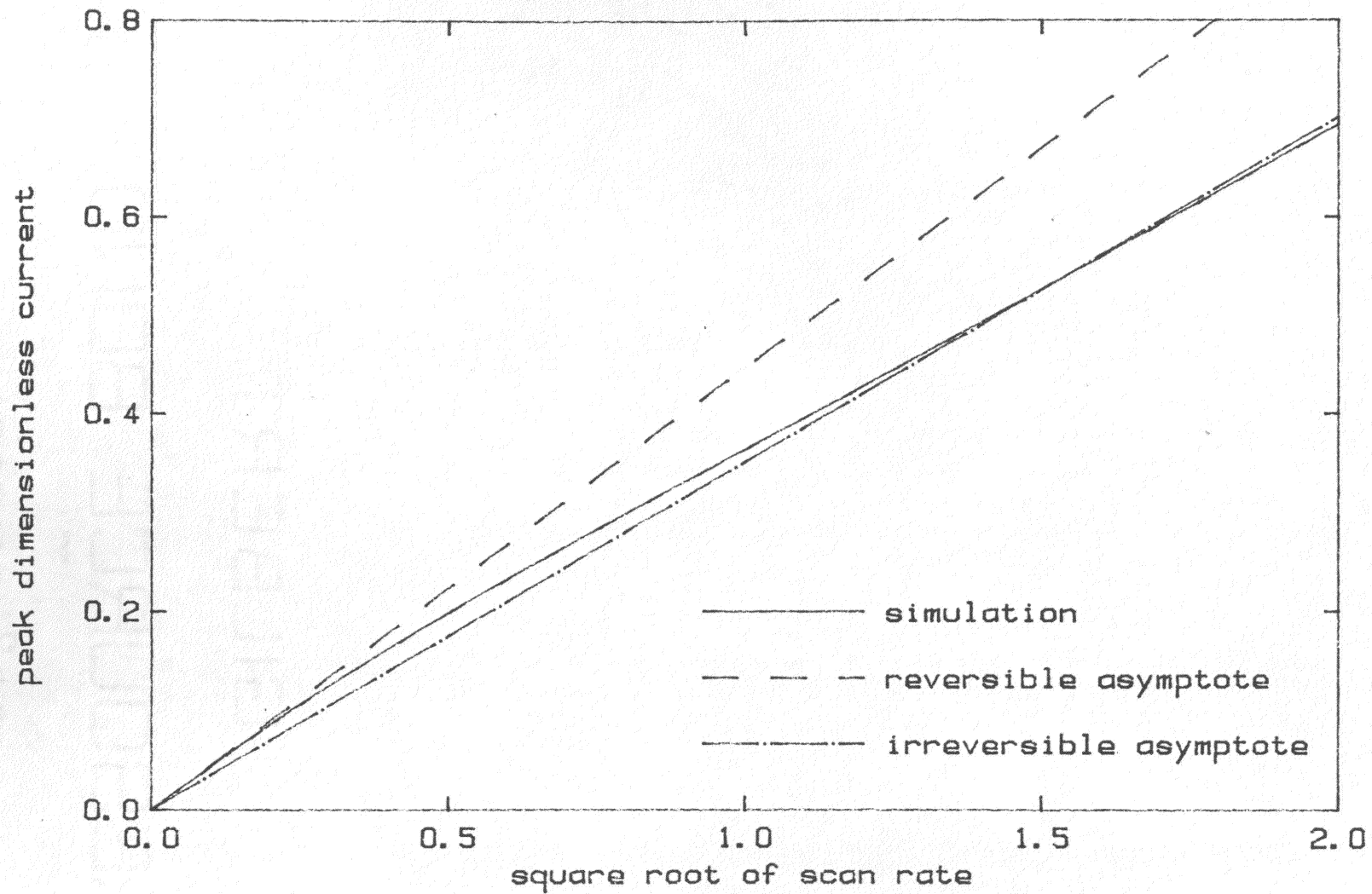
Figure 5.28 Transition from "reversible" to "irreversible" kinetics
(second derivative of slope functions set to zero at surface)

So, the values of the second derivatives of the slope functions were set to
zero at the first breakpoint (the electrode surface); while this helped
(see Figure 5.28), there was still a discrepancy at the high scan rates.
Recall an item from Fourier transform theory:

> ... the Fourier series for $f(x)$...converges to the value $f(x)$ at each
> point of continuity of $f(x)$; at each point of discontinuity of $f(x)$ the
> Fourier series converges to the arithmetic mean of the values
> approached by $f(x)$ from the right and the left.

[Theorem 23, pg 431, Rainville 1964]

The difference between the standard convention for spline approximation
[Davis 1984, de Boor 1978], and the proposed convention established by the
above theorem for Fourier approximations is illustrated by Figure 5.29
below.

Figure 5.29 Continuity conditions for spline functions
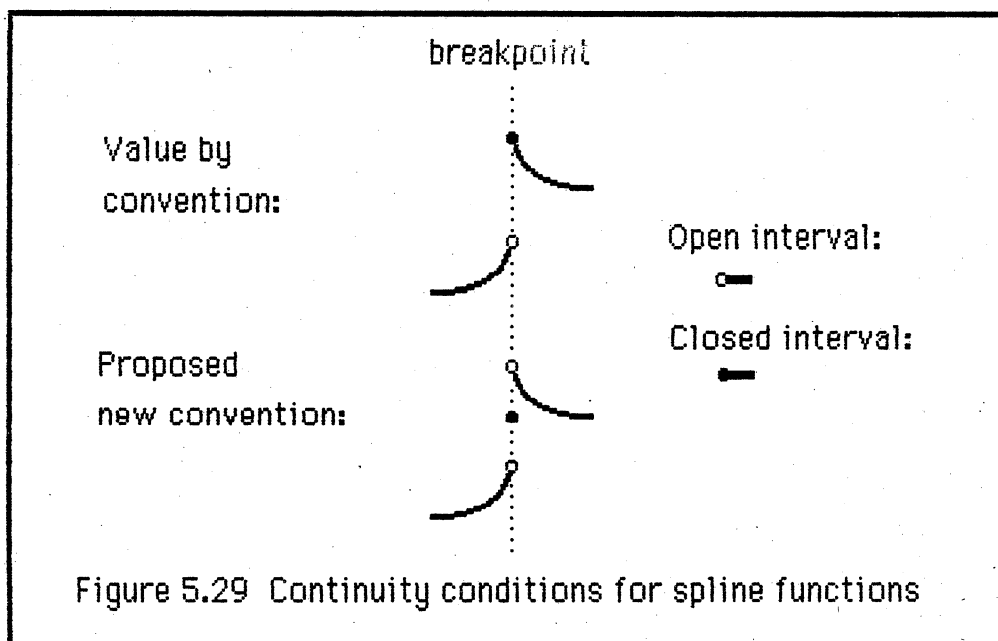
The spline "value" functions have finite second derivatives at the first

breakpoint, but for distances to the left of the first breakpoint, the second derivative is zero (constant value). The average between the two values for the second derivative was used. The actual values used for the second derivative of the spline basis functions at the first breakpoint were:

$$\varphi''_{value,1}(d=0) = 0.5\ \varphi''_{value,1}(d=0+dx) \qquad [5.10]$$

$$\varphi''_{slope,1}(d=0) = 0 \qquad [5.11]$$

$$\varphi''_{value,2}(d=0) = 0.5\ \varphi''_{value,2}(d=0+dx) \qquad [5.12]$$

$$\varphi''_{slope,2}(d=0) = 0 \qquad [5.13]$$

where dx is an infinitesimally small value

This gave the correct behavior as can be seen in Figure 5.30. From this figure, one can see that the behavior of the system at a dimensionless scan rate of less than about 0.01 is "reversible", while the behavior of the system at a dimensionless scan rate greater than about 2.25 is in the "irreversible" regime.

At low scan rates, the peak currents approach the limiting case of reversible kinetics, and at high scan rates the peak currents approach the limiting case of irreversible kinetics. Thus the model is validated in both limiting cases.

The successful model was then applied to the simulation of an electrochemical oscillogenic technique. This combined model is the FORTRAN program OSCGEN (OSCilloGENic instrument) in Listing 22.
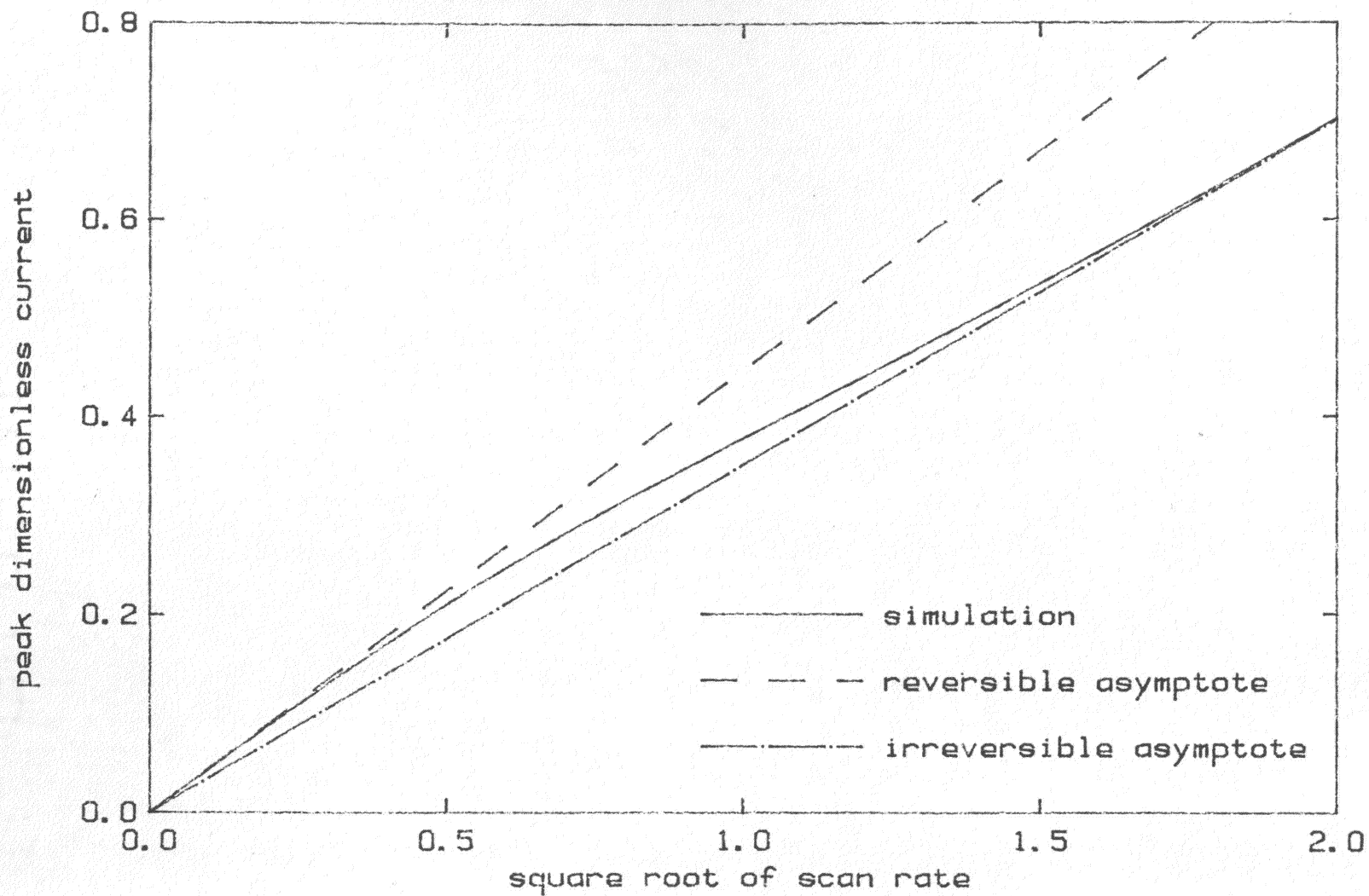
Figure 5.30 Transition from 'reversible' to 'irreversible' kinetics
("Fourier" convention for the splines at the breakpoints)

The value of the current flowing to the electrode was used as an input to a nonlinear feedback element. The output of this element, one of two potentials, was applied to the electrode as illustrated in Figure 5.31.

Figure 5.31 Block diagram of electrochemical oscillogenic instrument.
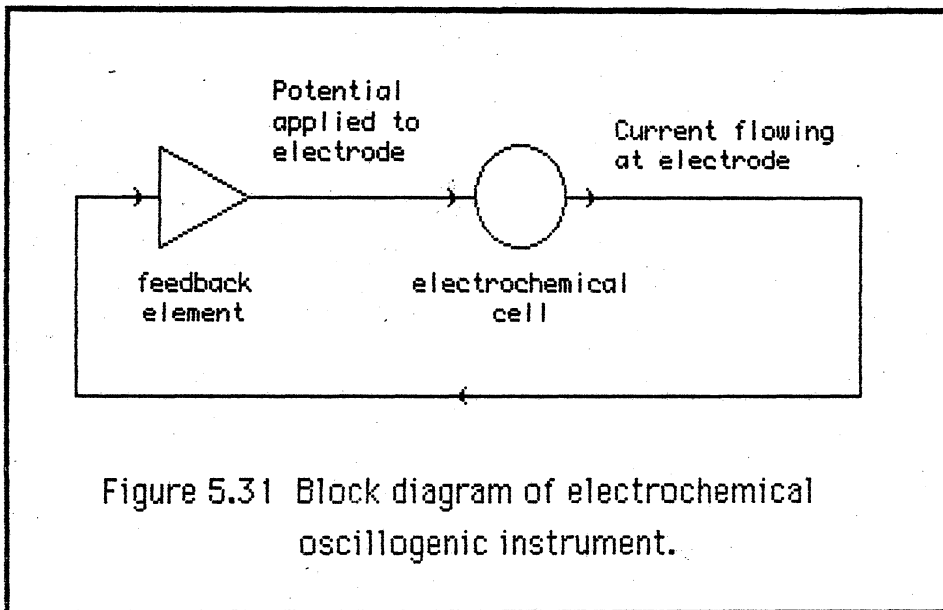
The rate of change of the system was calculated by the subroutine BINPOT (BINary applied POTential) in Listing 21. The action of this routine depended on the state (either high or low) of the potential applied to the simulated electrochemical system by the simulated feedback system. To assist in the design of a suitable nonlinear feedback switching unit, a simple square wave potential input was applied in a simulation of the electrochemical cell (Figure 5.32). When the potential changes, the current flowing at the electrode jumps to an extreme value and then decays somewhat. The code in Listing 22 used to generate Figure 5.32 is identified by the comment: "forcing function".

dimensionless time
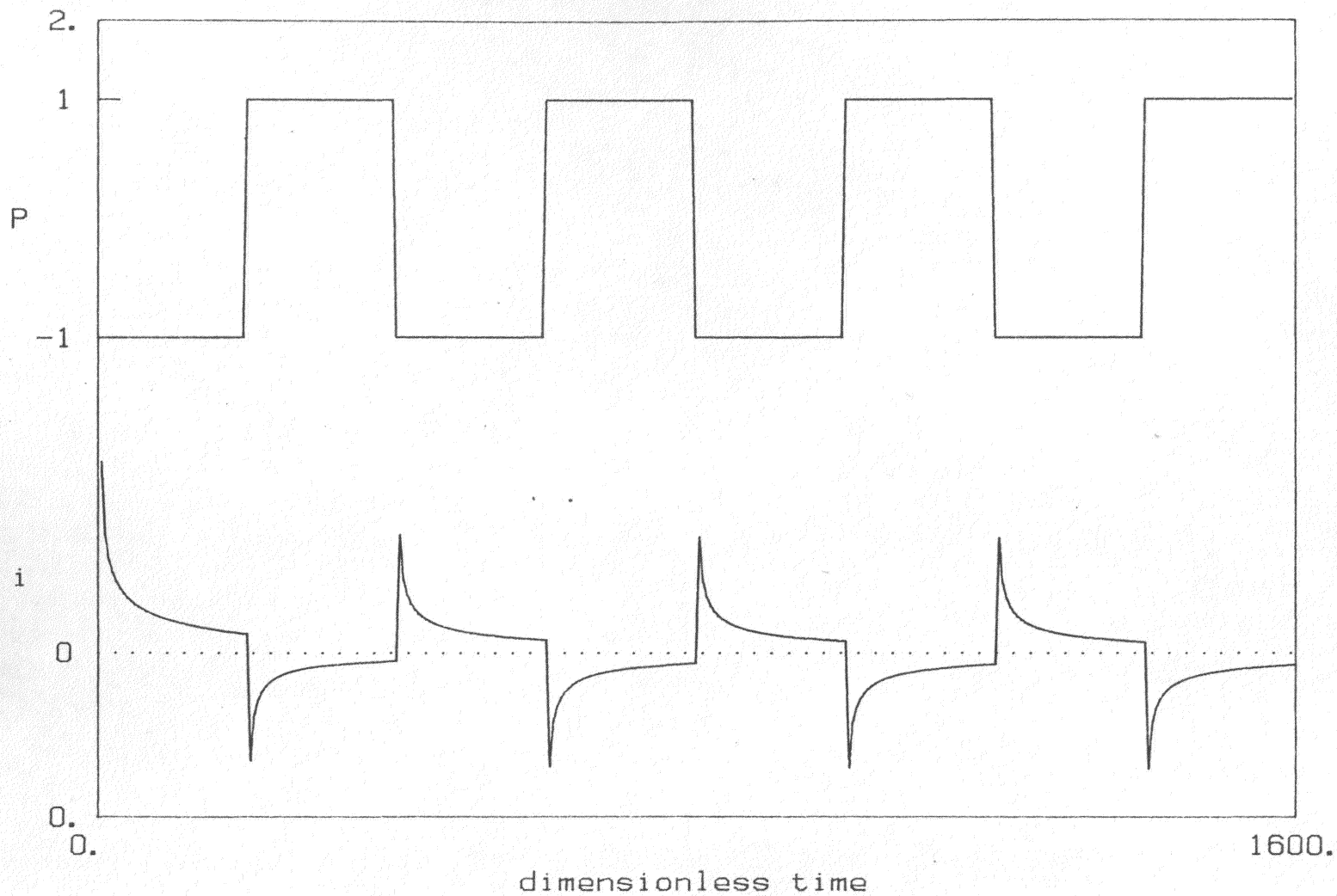
Figure 5.32 Response of an electrochemical system
to an applied square wave potential.

As described in chapter 2, one possible oscillator configuration applies two potential values alternately to the electrode with the switching rate controlled by the reaction kinetics of the electrochemical system. The function which generates this switching between potentials is a nonlinear feedback signal external to the electrochemical cell. One such nonlinear function (suggested by Figure 5.32) is a function which changes the output voltage if the current gets "close" to zero. The code in Listing 22 which acts in this fashion is identified by the comment (in **bold** face): "switching based on current". An electronic circuit that might generate the same function as the FORTRAN code is proposed in Figure 5.33; the circuit elements used in this figure are described by Rony [1979] and Lancaster [1974, 1977].

Figure 5.33 Possible feedback element for
switching based on "current"
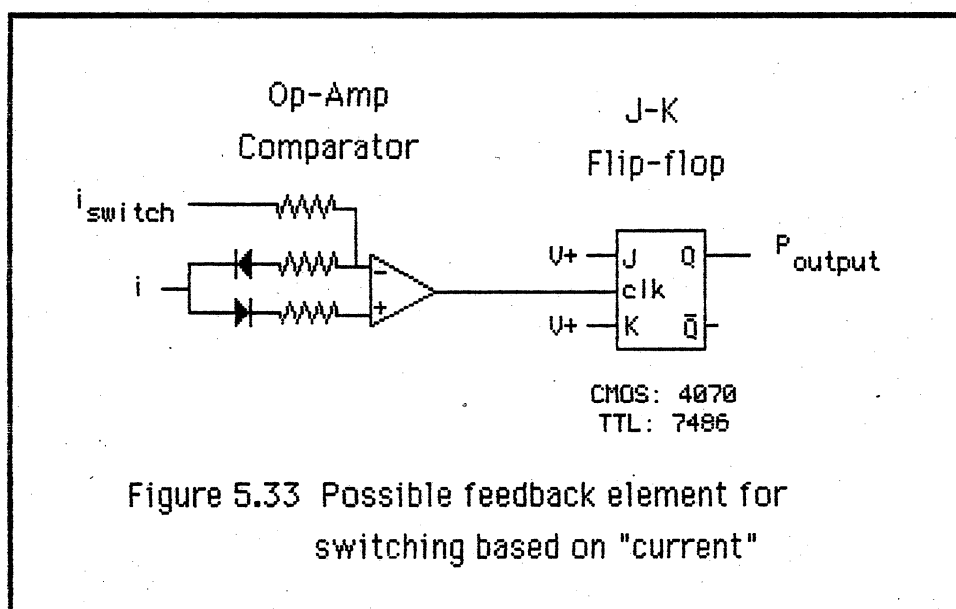
The results of using this type of nonlinear feedback element are shown in Figure 5.34.
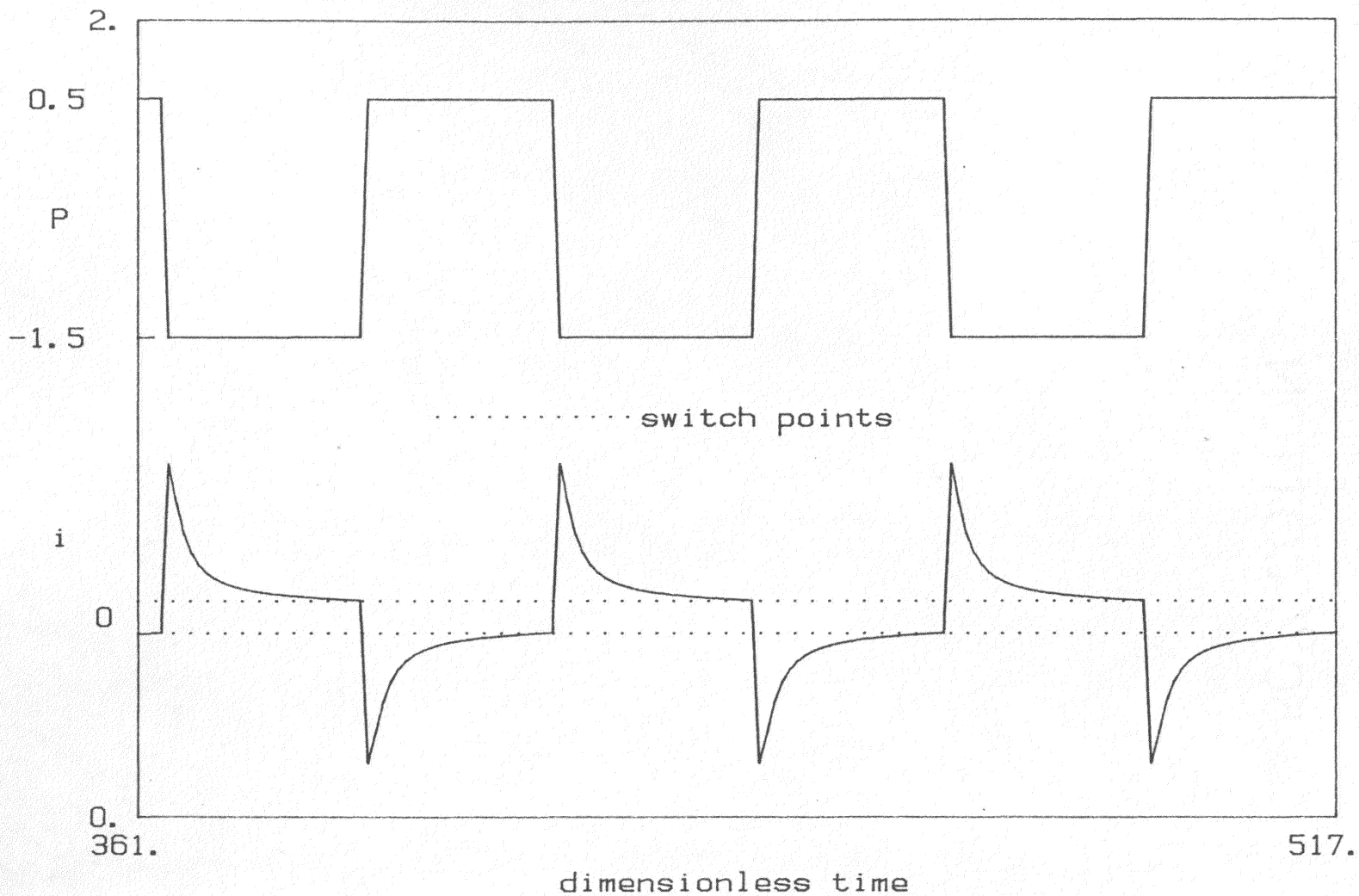
Figure 5.34 Electrochemical oscillogenic instrument simulation.
Nonlinear feedback switching function based on current.
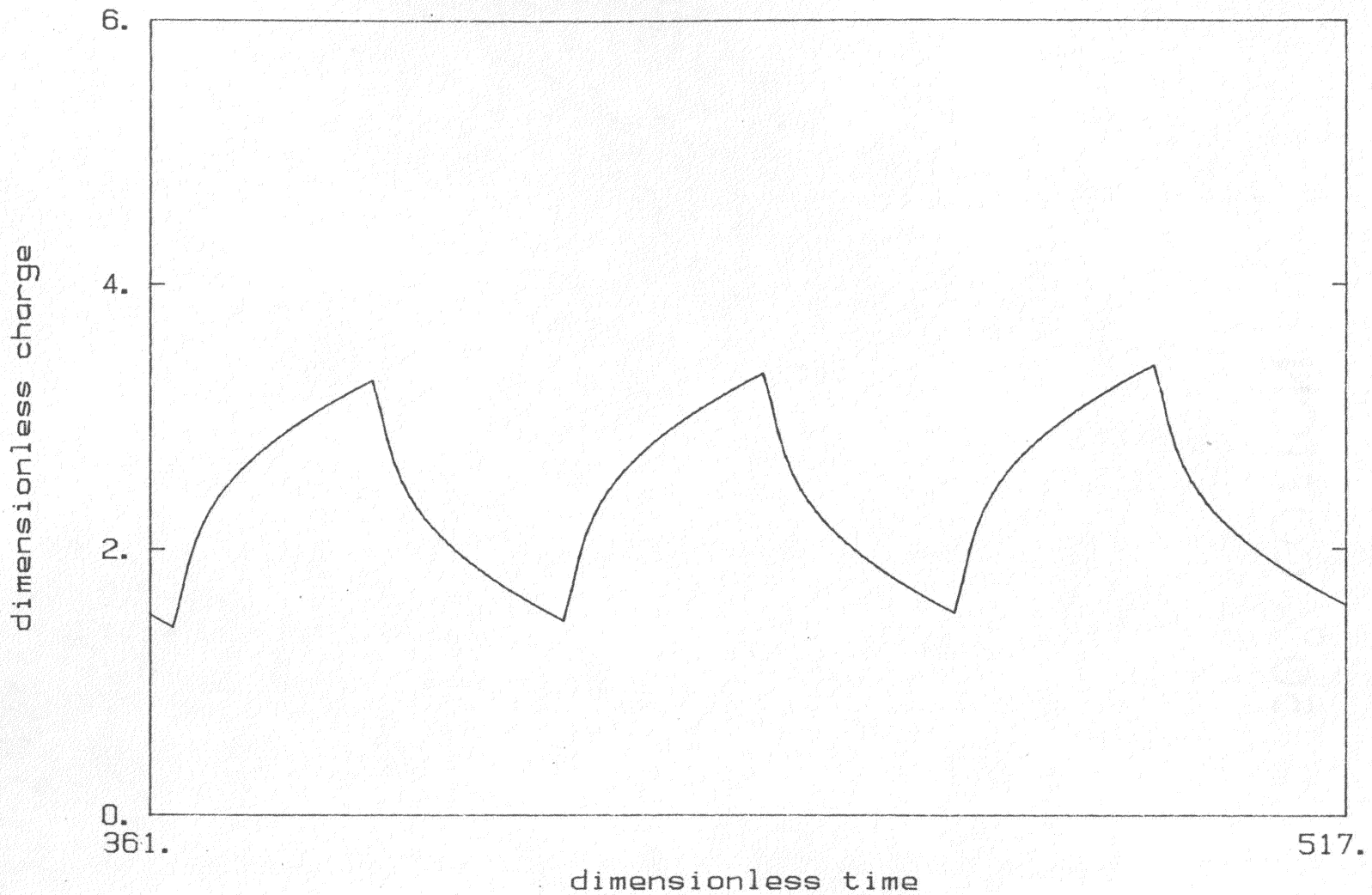
Figure 5.35 Charge transferred as a function of time.
(integral of the current in figure 5.34)
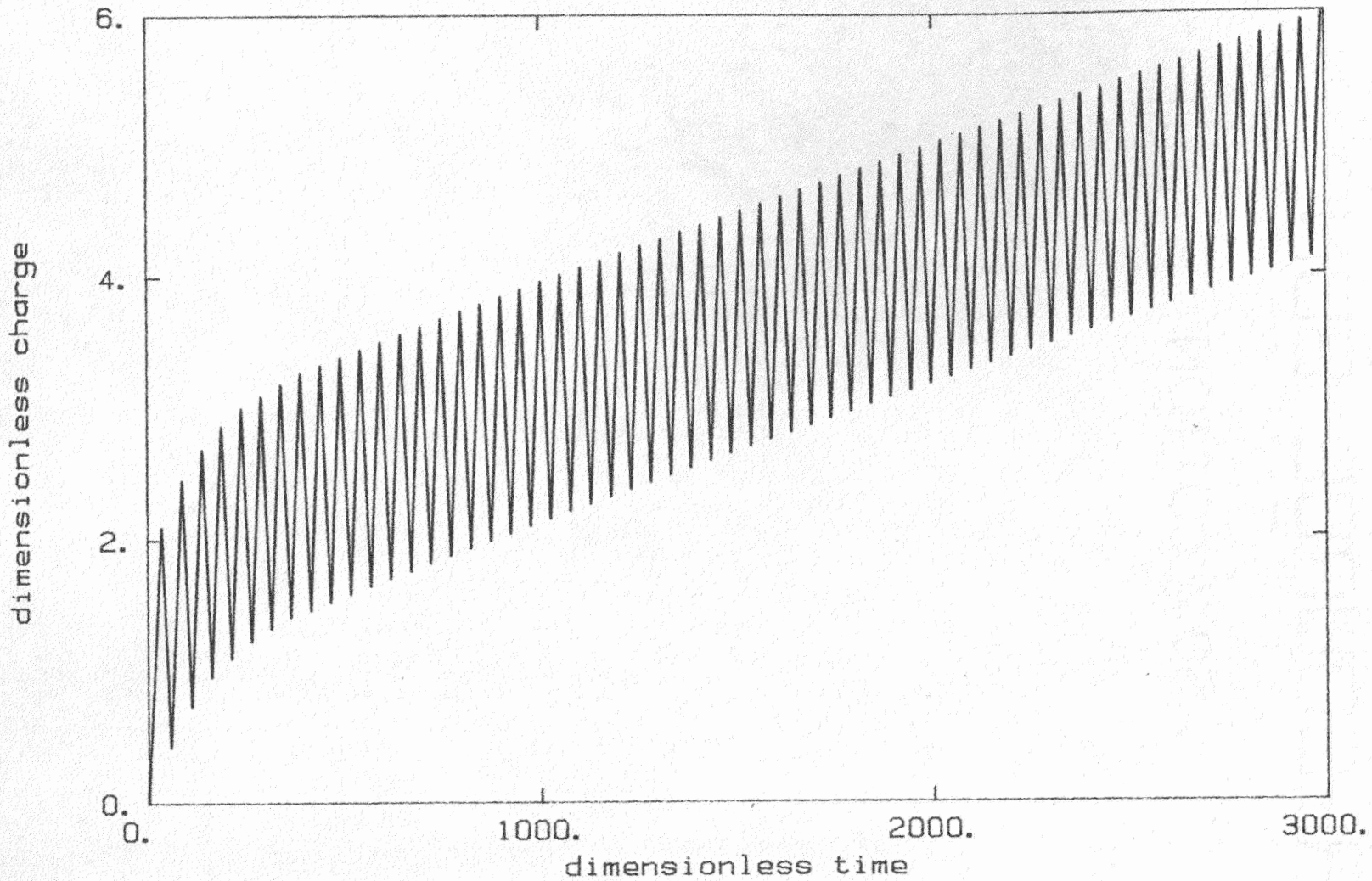
Figure 5.36   Charge transferred as a function of time.
(long term trend, switching based on current)

Switching the potential as a function of the current produced a waveform that was not really periodic, because there was a net flux of electroactive ions towards the surface of the electrode. Figure 5.35 shows the integral of the current (the net charge transferred). The long term drift of the net charge transferred can be seen in Figure 5.36; there is an increase in the net charge transferred. This indicates a non-steady state condition and an unstable oscillator. This also means that if the product of the electrochemical reaction is not soluble in the bulk electrolyte solution, then there will be a net deposit on the surface of the electrode. This may change the characteristics of the electrode over time, and introduce another source of drift.

As an alternative (already alluded to in chapter 2) to switching the potential based on the current, the switching element was modified to switch on the charge transferred.

Figure 5.37 Possible feedback element for switching based on "charge"
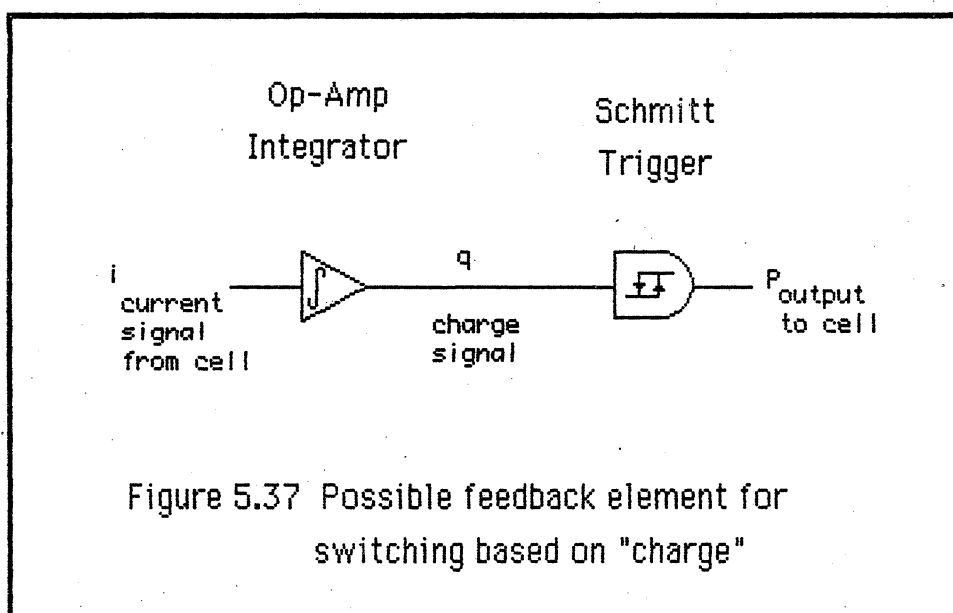
It was hoped that this would lead to a more stable oscillator. The section of FORTRAN Listing 22 commented as: "switching based on charge" implemented just such a switching method. An electronic circuit designed to be functionally similar to the FORTRAN code is shown in Figure 5.37.

The results of the oscillogenic instrument simulation using switching based on charge are shown in Figure 5.38. For comparison, the current as a function of time is shown in Figure 5.39, and the long term trend in the charge transferred is shown in Figure 5.40. The current flowing to the electrode is a measure of a flux in the system. A more stable oscillator results from following the guidelines of chapter 2, and from integrating this flux and switching the voltage input by using a function of the charge transferred. The charge is a cumulative quantity while the current is a resilient quantity.

The model of the oscillogenic instrument can be used in a sensitivity analysis of the instrument. The equations defining the dimensionless quantities are:

$$\text{dimensionless time} = \tau = t\, k_0^2 / \mathcal{D}_{Am} \tag{4.16}$$

$$\text{dimensionless current} = \frac{i}{n\, F\, A\, C_{A,bulk}\, k_0} = \left.\frac{\partial f}{\partial d}\right|_{d=0} \tag{5.6}$$

From these equations, one can see the direct effect of each quantity listed. The switching function (the Schmitt trigger) depends on the charge transferred. If the rate of charge transfer doubles, then the time between switch points halves and the frequency of the oscillator doubles. Since the switching is proportional to the integral of the current, the frequency
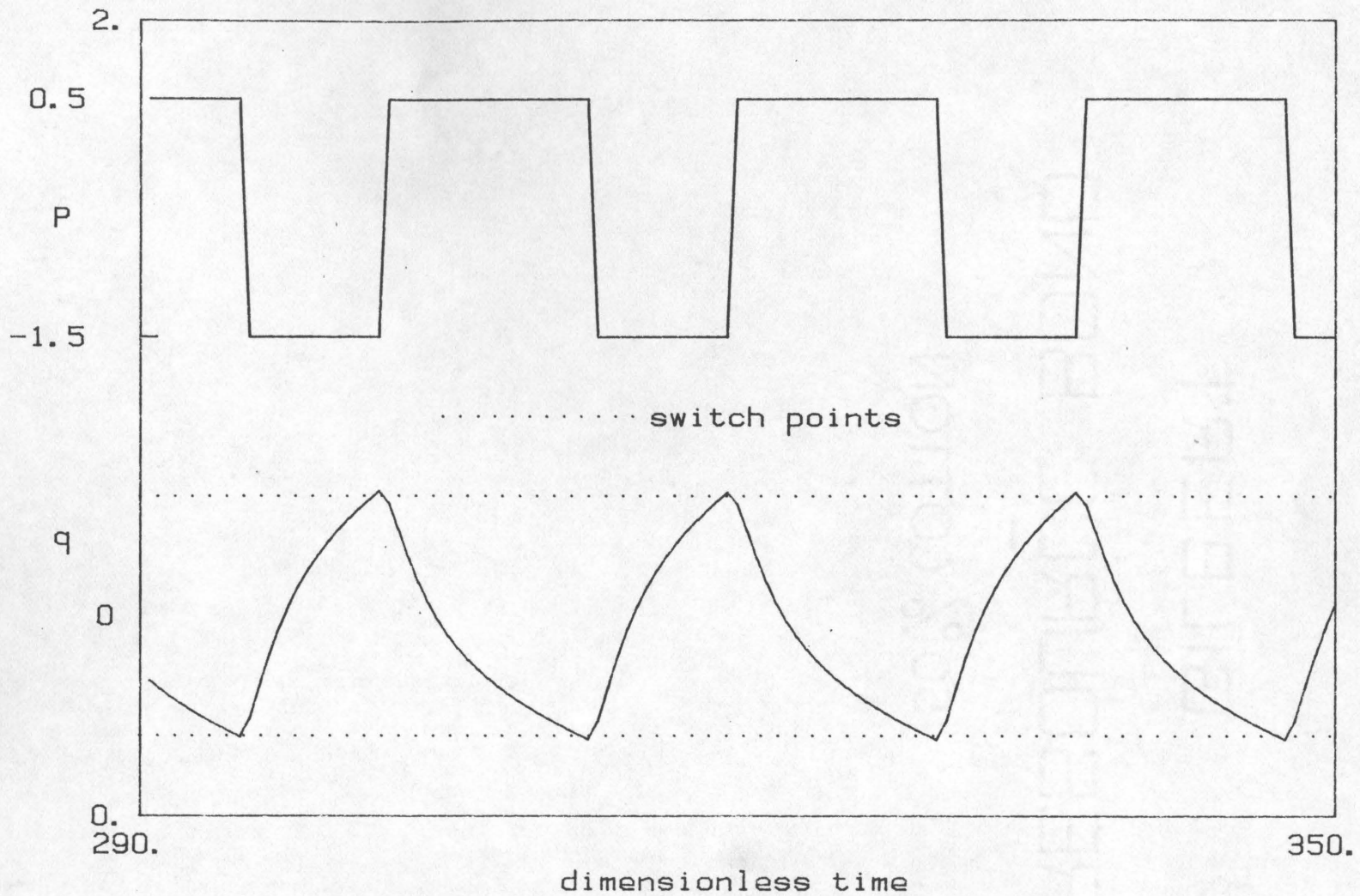
Figure 5.38 Electrochemical oscillogenic instrument simulation.
Nonlinear feedback switching function based on charge.
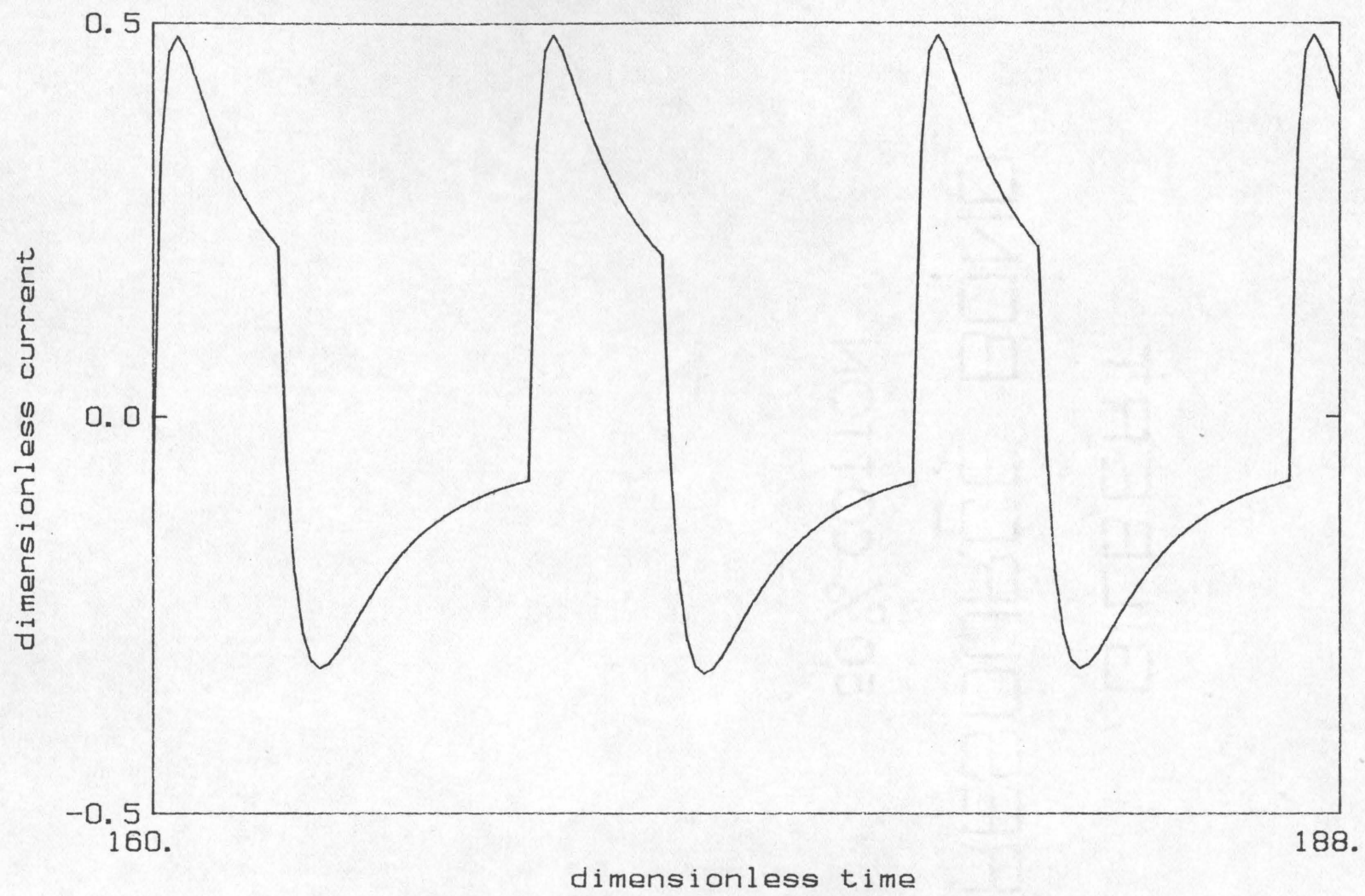
Figure 5.39 Electrochemical oscillogenic instrument simulation.
Nonlinear feedback switching function based on charge.
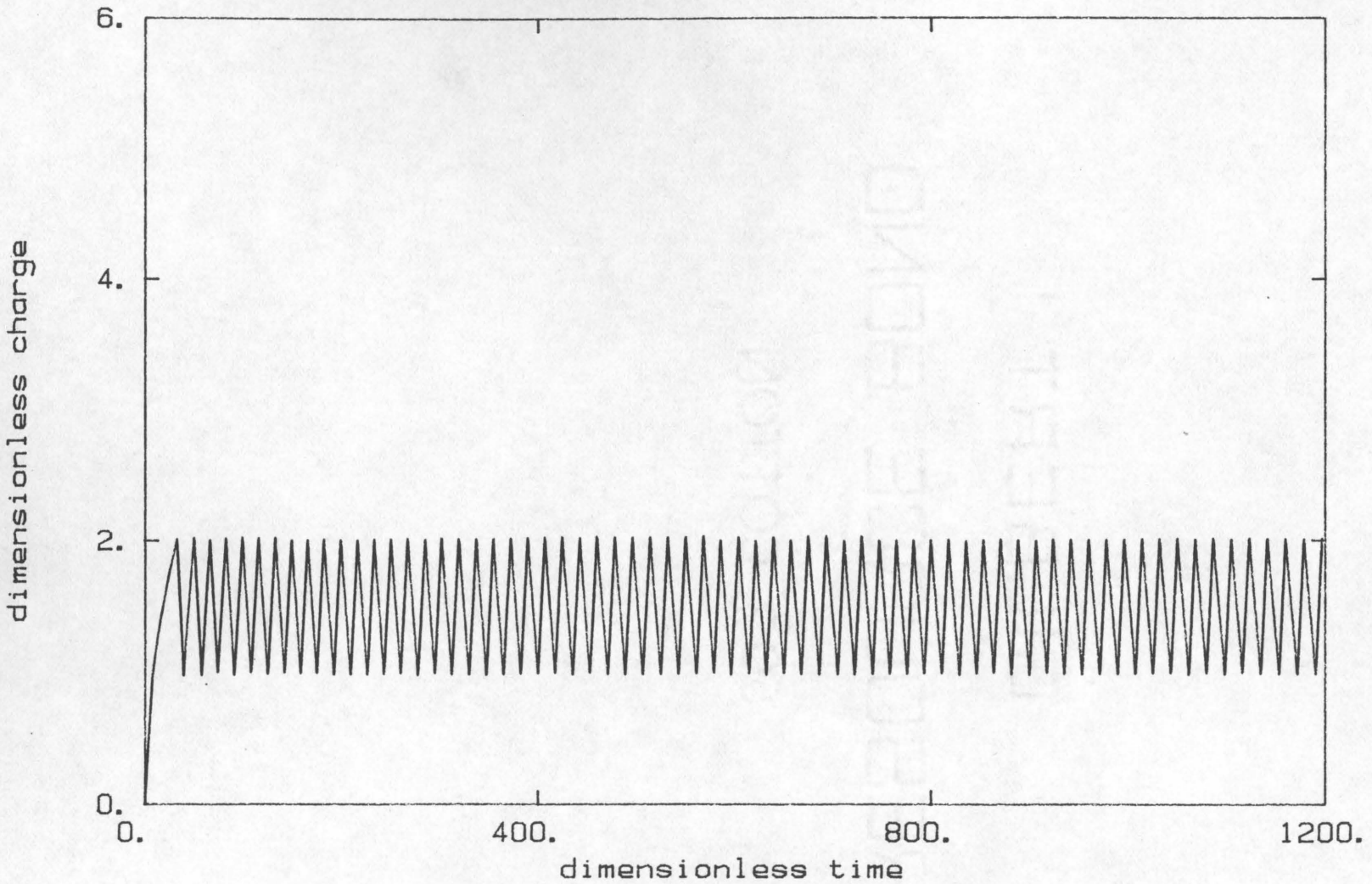
Figure 5. 40 Charge transferred as a function of time.
(long term trend, switching based on charge)

of the oscillogenic instrument will be directly proportional to the current. Equation 5.6 shows the variables which affect the conversion from dimensionless current to dimensioned (physically observed) current: the surface area of the electrode (A), the number of electrons reacting for each ion reacting (n), and the bulk concentration of the electroactive species ($C_{A,bulk}$):

$$\text{frequency} \propto 1 / \text{charge (q)} \qquad [5.14]$$

$$\text{frequency} \propto 1 / \text{current (i)} \qquad [5.15]$$

$$\text{and,} \quad \text{frequency} \propto (n \, F \, A \, C_{A,bulk} \, k_0) \qquad [5.16]$$

With all other things constant, the dimensionless time expression (equation 4.16) shows that the frequency will be inversely proportional to the diffusivity:

$$\text{frequency} \propto 1 / t_{period} \qquad [5.17]$$

$$\text{frequency} \propto k_0^2 / \tau_{period} \mathscr{D}_{Am} \qquad [5.18]$$

$$\text{and,} \quad \text{frequency} \propto 1 / \mathscr{D}_{Am} \qquad [5.19]$$

The evaluation of the sensitivity to the rate constant, k_0, is complicated by the fact that it appears in both equations 4.16 and 5.6. The dimensionless period of the waveform from the oscillogenic instrument will be a function of the dimensionless current:

$$\tau_{period} = F[\text{dimensionless current}] \qquad [5.20]$$

$$\tau_{period} = F[\, i/(nFAC_{A,bulk}k_0) \,] \qquad [5.21]$$

and $\quad \tau_{period} \propto t_{period} k_0^2$ [5.22]

so $\quad t_{period} \propto F[\, i/(nFAC_{A,bulk}k_0) \,] / k_0^2$ [5.23]

and to a first approximation:

$\quad t_{period} \propto 1 / k_0^3$ [5.24]

thus to a first approximation,

$\quad$ frequency $\propto k_0^3$ [5.25]

In this model there are only three other quantities: 1) β (the charge transfer coefficient, see equations 4.7 and 4.8), 2) the size of the potential switch (ΔP, see Listing 22), and 3) the average potential ($P_{average}$, also see Listing 22). The sensitivity to these quantities must be established by numerical experiment. Table 5.1 shows the effect of β on the period and frequency of the oscillogenic system; note that it has a minimal effect and that it does not vary much in electrochemical systems [Brown 1984]. Table 5.2 shows the effect of the potential step (ΔP), which is non-linear and significant. Table 5.3 shows the effect of the average potential ($P_{average}$) on the frequency; note that the frequency passes through a peak at an average potential value of about -0.2. After the experiments described in chapter 6 were performed, a more extensive simulation of a scan of the average potential was calculated. The results of this simulation are shown in Figure 5.41. Since this figure is a simulation of a quasi-reverisible electrochemical reaction, this may explain why the peak potential is not located at a dimensionless potential of exactly zero. Note the remarkable symmetry, which was also observed experimentally.

Table 5.1 The effect of β (the charge transfer coefficient) on the frequency of the model oscillogenic system (in units of dimensionless time).

β	Period	Frequency
0.1	16.48	0.0607
0.2	16.33	0.0612
0.3	16.21	0.0616
0.4	16.07	0.0622
0.5	16.01	0.0625
0.6	16.00	0.0625
0.7	15.99	0.0625
0.8	16.00	0.0625
0.9	16.00	0.0625

Note: $\Delta P=2$, $P_{average}=0$, $q_{low}=1$, $q_{high}=2$, measurements taken after 16 cycles.

Table 5.2 The effect of ΔP (the potential step size) on the frequency of the model oscillogenic system (in units of dimensionless time).

ΔP	Period	Frequency
1	55.7	0.018
2	16.01	0.063
3	9.46	0.105
4	7.25	0.138
5	6.19	0.162
6	5.56	0.180
7	5.23	0.191
8	4.95	0.202
9	4.76	0.210

Note: $\beta=0.5$, $P_{average}=0$, $q_{low}=1$, $q_{high}=2$, measurements taken after 16 cycles.

Table 5.3 The effect of $P_{average}$ (the average potential) on the frequency of the model oscillogenic system (in units of dimensionless time).

$P_{average}$	Period	Frequency
0.0	16.01	0.0625
-0.1	15.32	0.0652
-0.2	15.01	0.0666
-0.3	15.11	0.0662
-0.4	15.60	0.0641
-0.5	16.59	0.0603
-0.6	18.29	0.0547
-0.7	21.04	0.0475
-0.8	25.89	0.0386

Note: $\beta=0.5$, $\Delta P=2$, $q_{low}=1$, $q_{high}=2$, measurements taken after 16 cycles.
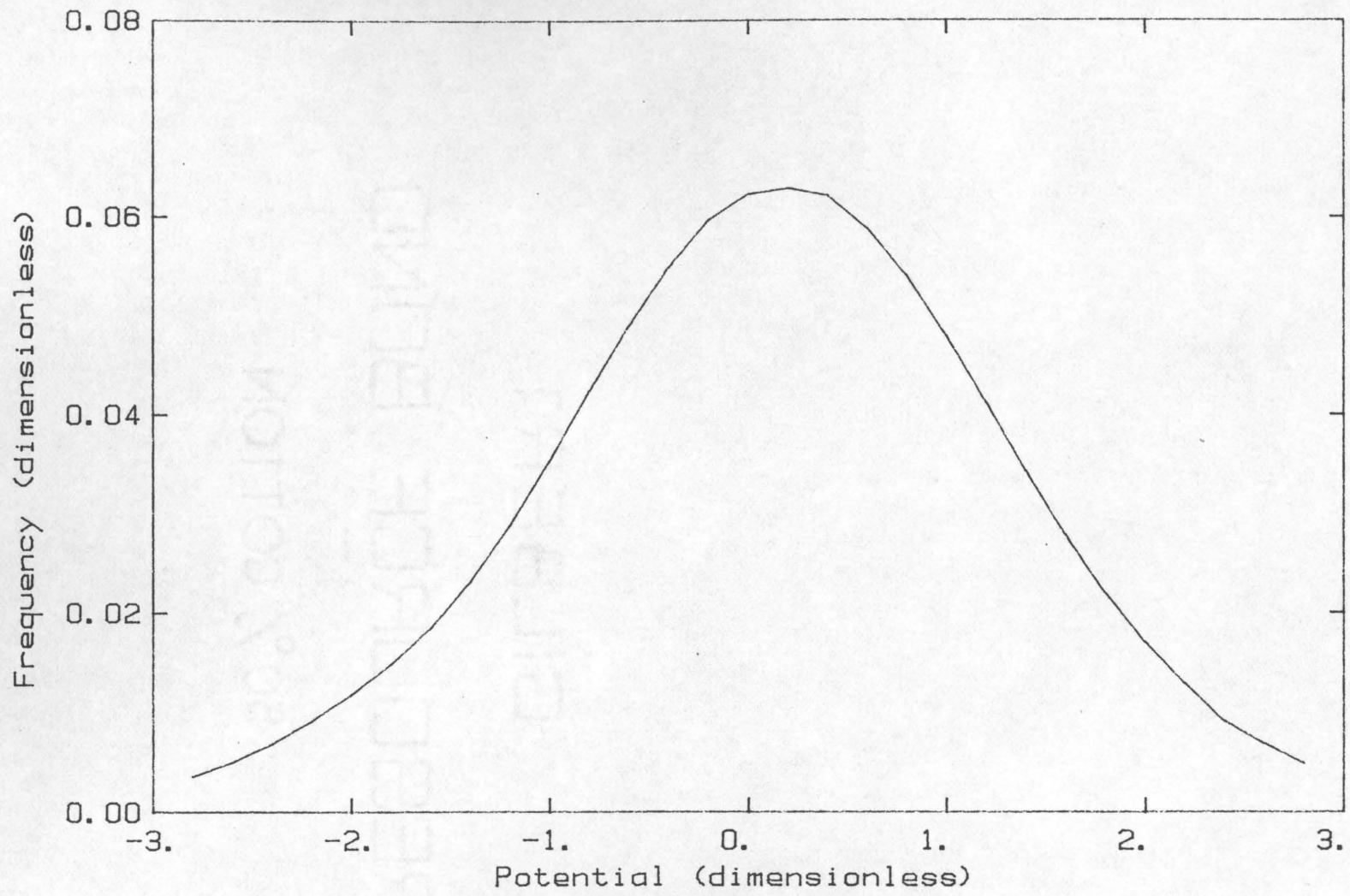
Figure 5.41   Frequency during a scan in potential
for a simulated oscillogenic instrument

In summary, what are the achievements of this chapter? There were a number of improvements in the numerical analysis techniques employed. First, the presentation of the FORTRAN coding of a simple method for redistributing the breakpoints to spread the approximation errors evenly over the region of interest. Second, the use of a boundary condition at the surface of the electrode, which reduced considerably the computational effort required. And third, the proposal of an improved convention for the values of the spline functions at the breakpoints.

Finally, the discovery that a stable oscillator could not be constructed by using the current value as the input to a hysteresis element was the greatest achievement of the model. It saved many hours of experimental work by pointing out a better approach. The mathematical model also indicated that an electrochemical oscillator was indeed a possibility.

Chapter 6: Experimental Implementation of an Oscillogenic Technique

> *"When you make a thing, a thing*
> *that is new, it is so complicated*
> *making it*
> *that it is bound to be ugly.*
> *But those that make it after you,*
> *they don't have to worry*
> *about making it.*
> *And they can make it pretty, and so*
> *everybody can like it*
> *when the others*
> *make it after you."*
>
> *Pablo Picasso (as quoted by Gertrude Stein)*

The mathematical model of an electrochemical oscillogenic system indicated the direction for synthesizing an experimental verification of the technique. An electrochemical cell and supporting electronics were needed for the experiments. As with the mathematical model, the experimental procedure was validated by the accepted technique of scanning voltammetry.

The electrochemical cell was constructed after a design by H. O. Finklea of the Chemistry Department, VPI&SU [Finklea 1982]. Figure 6.1 shows a drawing of the cell. It was constructed of borosilicate glass with a polyfluoroethylene three-way stopcock valve. The cell was constructed with cylindrical symmetry, and the side well for the reference electrode was connected to the working cell electrolyte by a 2 mm inside diameter tube (a *Luggin probe*). Because of the cylindrical symmetry of the cell and the fact that the Luggin probe was placed at the axis of the cylinder, the large diameter of this reference electrode probe would not affect the symmetry of the lines of constant electrical
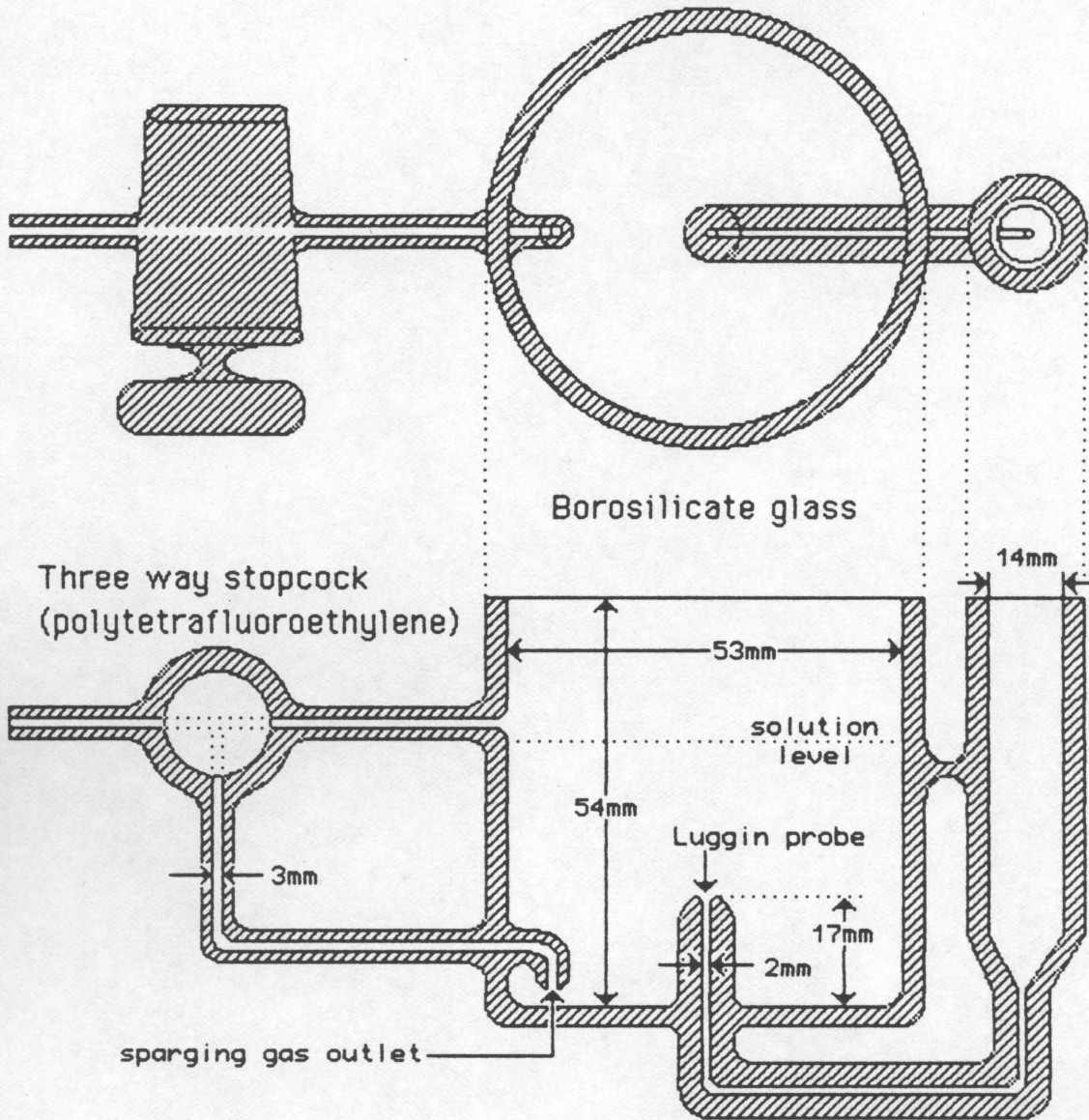
Figure 6.1 Glass component of electrochemical cell

potential around the working electrode. The reference electrode was a saturated Calomel electrode (Beckman).

The cell was filled with electrolyte to a level just below the upper gas inlet. The cell held from 60 to 70 ml of liquid. For future work, the cell diameter might well be reduced by a factor of two or three, to reduce required solution volume without affecting the performance of the cell.

A cap for the cell was constructed of polytetrafluroethylene (donated by H. O. Finklea of the Chemistry Department, VPI&SU). The construction is illustrated in Figure 6.2. The cap served as a support for the working and counter electrodes, as well as a seal to allow a blanket of inert gas in the space above the electrolyte in the cell.

The counter electrode was a piece of platinum gauze (25 mm by 50 mm, 52 mesh, Fisher Scientific Co., Springfield, NJ 07081) formed into a cylinder. The gauze was held in a cylindrical form by spot welding the two ends together, and it was suspended from the cell cover by two lengths of heavy gauge platinum wire (70 mm each, 26 gauge, Fisher Scientific Co., 711 Forbes Ave., Pittsburgh, PA 15219), which were also spot welded to the platinum gauze. These support wires were held in vertical holes by means of a threaded plastic rod, which clamped the platinum wires in place. The threaded rod was screwed into a tapped hole which intersected the vertical hole at a right angle.
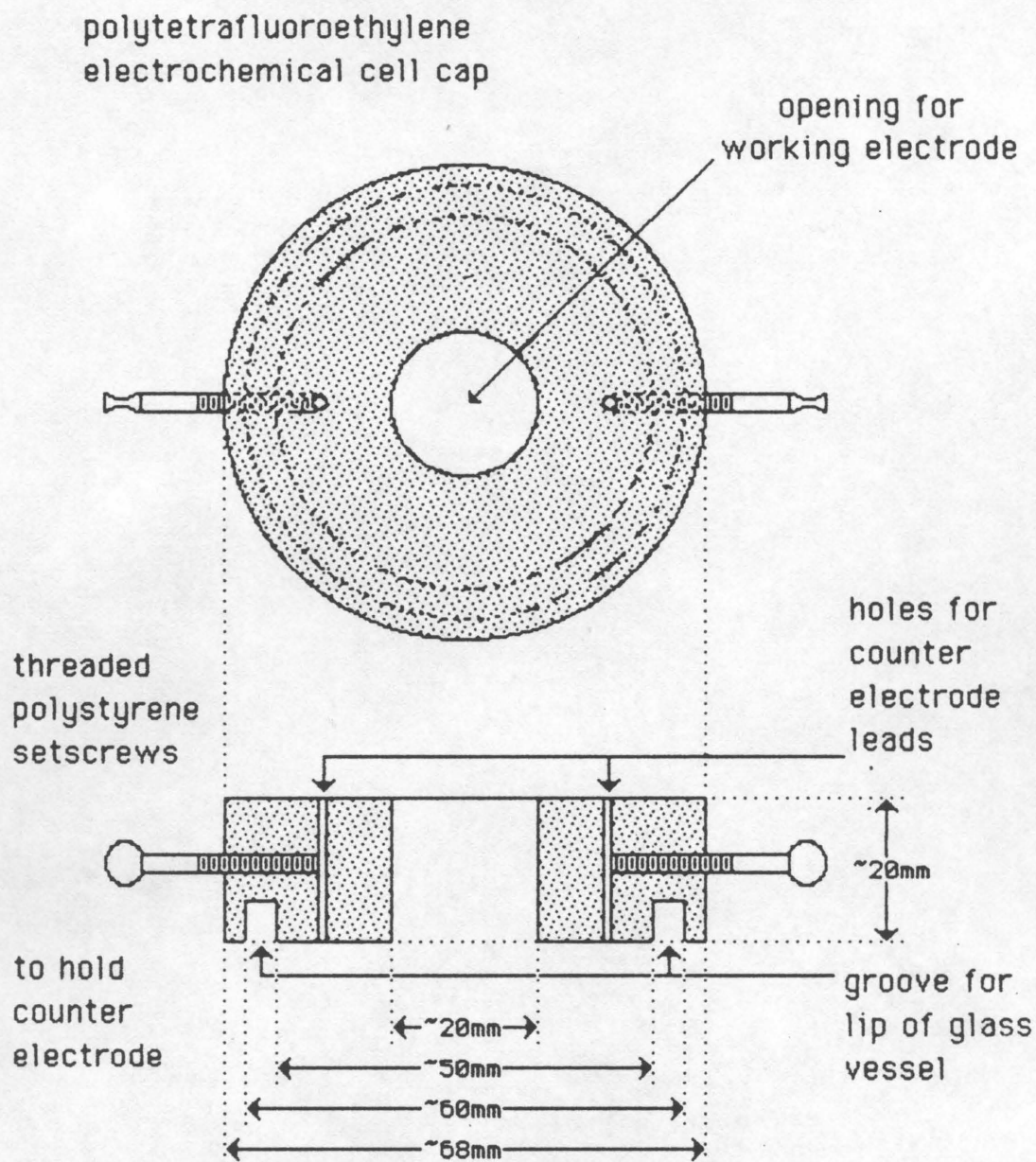
Figure 6.2 Cap for electrochemical cell

Glassware was cleaned before the first use by the following procedure:

1. Soaking overnight in "Nochromix" cleaning solution.

2. Rinsing with distilled water.

3. Filling with methanolic KOH, and rubbing the surfaces well.

4. Rinsing with distilled water.

5. Rinsing with concentrated HNO_3.

6. Rinsing with distilled water.

7. Drying at 103 °C in an oven.

After this initial cleaning, when the solution in a piece of glassware was changed, the glass was rinsed with a dilute solution of HNO_3 and then with distilled water.

The water used was city water passed through a pair of ion-exchange columns and then distilled in glass (Corning Mega-pure automatic still). Since the purity seemed adequate for these purposes, no further purification was attempted. Reagents were A.C.S. certified grade and no further purification of these was attempted.

The working electrode was either a short length of heavy gauge platinum wire, or a hanging mercury drop electrode, depending on the nature of the electrochemistry. The two electrodes used are sketched in Figure 6.3.

The platinum working electrode was constructed of a piece of heavy gauge platinum wire with a "Kynar wire wrap wire" lead soldered to one end. The other end of the platinum wire was sealed with "Duco" cement

Platinum
electrode

hanging
Mercury drop
electrode
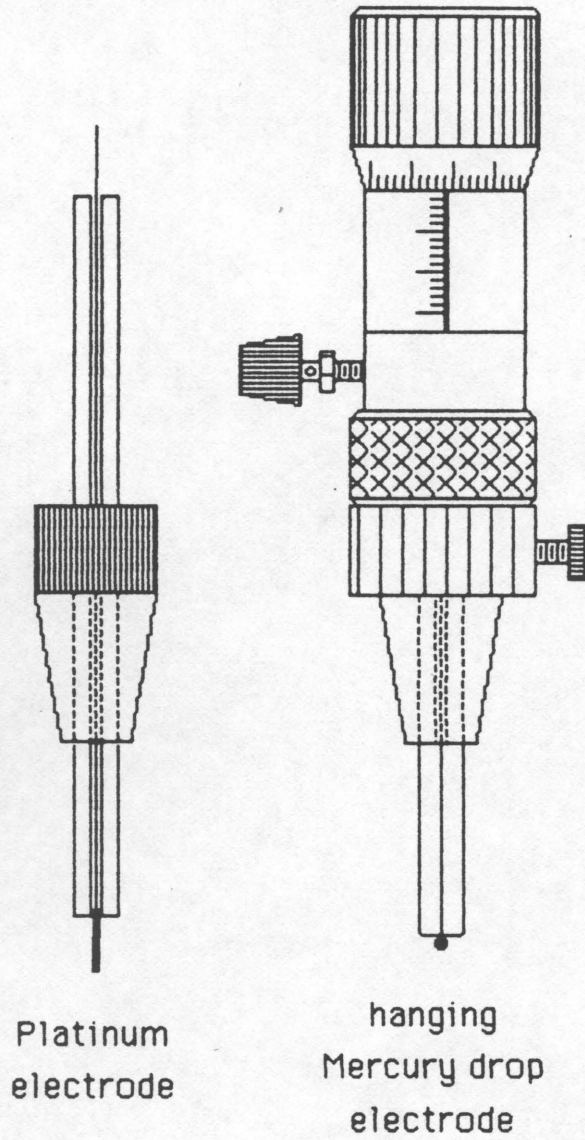
Figure 6.3 Working electrodes

into the end of glass capillary such that about 5 millimeters of bare end protruded from the cement (wire 0.48 mm diameter, 4.1 mm exposed length, 6.2 mm^2 exposed area).

The hanging mercury drop electrode was a commercial device from Metrohm (model E410, 'microfeeder' electrode, Metrohm AG, CH-9100 Herisau, Switzerland), kindly provided by H. O. Finklea of the Chemistry Department of Virginia Polytechnic Institute and State University. This electrode was equipped with a mercury reservoir the volume of which was controlled by a micrometer piston. This reservoir was connected to a glass capillary tube and the mercury drop exuded and hung from the end, suspended in the electrolyte. The size of the mercury drop was measured by the number of divisions the micrometer was moved for each drop (this work used 4 divisions per drop, which from the literature on the E410 electrode is 0.83 mm in diameter and 2.22 ± 0.07 mm^2 in area). Electrical contact was made through the metal body of the electrode to the mercury, and a metal stud was attached to the side of the metal body for connection to the potentiostat. Particular care must be exercised in the cleaning and handling of the glass capillary. The same cleaning procedure used above for all of the glassware was employed. A polyvinyl-chloride plastic tubing was used to connect the upper end of the capillary to a water aspirator in order to suck the cleaning solutions through the capillary. After drying, and while the glass was still hot, a low-viscosity silicone oil was sucked through the capillary. After sucking air through the tube for a while to clear the excess silicone oil, the electrode was assembled.

The silicone oil acts as a hydrophobic material that helps to keep the electrolyte solution from wetting the interior of the capillary. Caution must be exercised to keep the electrolyte solution out of the capillary. This caution includes always keeping a droplet of mercury hanging on the end of the capillary while it is in the solution; when changing droplets on the end, by extruding the mercury, this was not always possible. If the droplets of mercury cannot be made to the desired size, or they fall off of the tip of the capillary too easily, then the capillary should be demounted from the metal body and re-cleaned.

Figure 6.4 shows the overall relationships of the various parts of the electrochemical cell. Figure 6.5 shows the cell after assembly, in the operation of sparging the electrolyte with inert gas (low oxygen content nitrogen, <5ppm, AIRCO). The nitrogen cylinder was equipped with a double stage regulator and a needle valve to regulate the flow rate of the nitrogen; first nickel chromium (stainless) steel tubing and then polyvinyl (Tygon) tubing were used to connect the gas valve to the glass cell.

The electrode assembly was mounted on a ring stand. A white ceramic base was preferred for the ring stand for better viewing of the working/counter electrode placement. Also, a small incandescent light bulb placed behind the glass electrochemical cell helped in positioning and using the electrodes. The assembly was shielded from electrical noise by a Faraday cage constructed from copper window screen. The screen was cut and bent into a box shape just big enough to enclose the base of the ring stand and the clamps holding the electrode assembly. The joints of the screen were soldered together with Lead/Tin solder. A box shaped lid

working
electrode
assembly

reference
electrode

KCl
electrolyte

Polytetrafluoroethylene cap

Platinum
counter
electrode

reference
electrode
well

three-way stopcock

borosilicate
glass cell
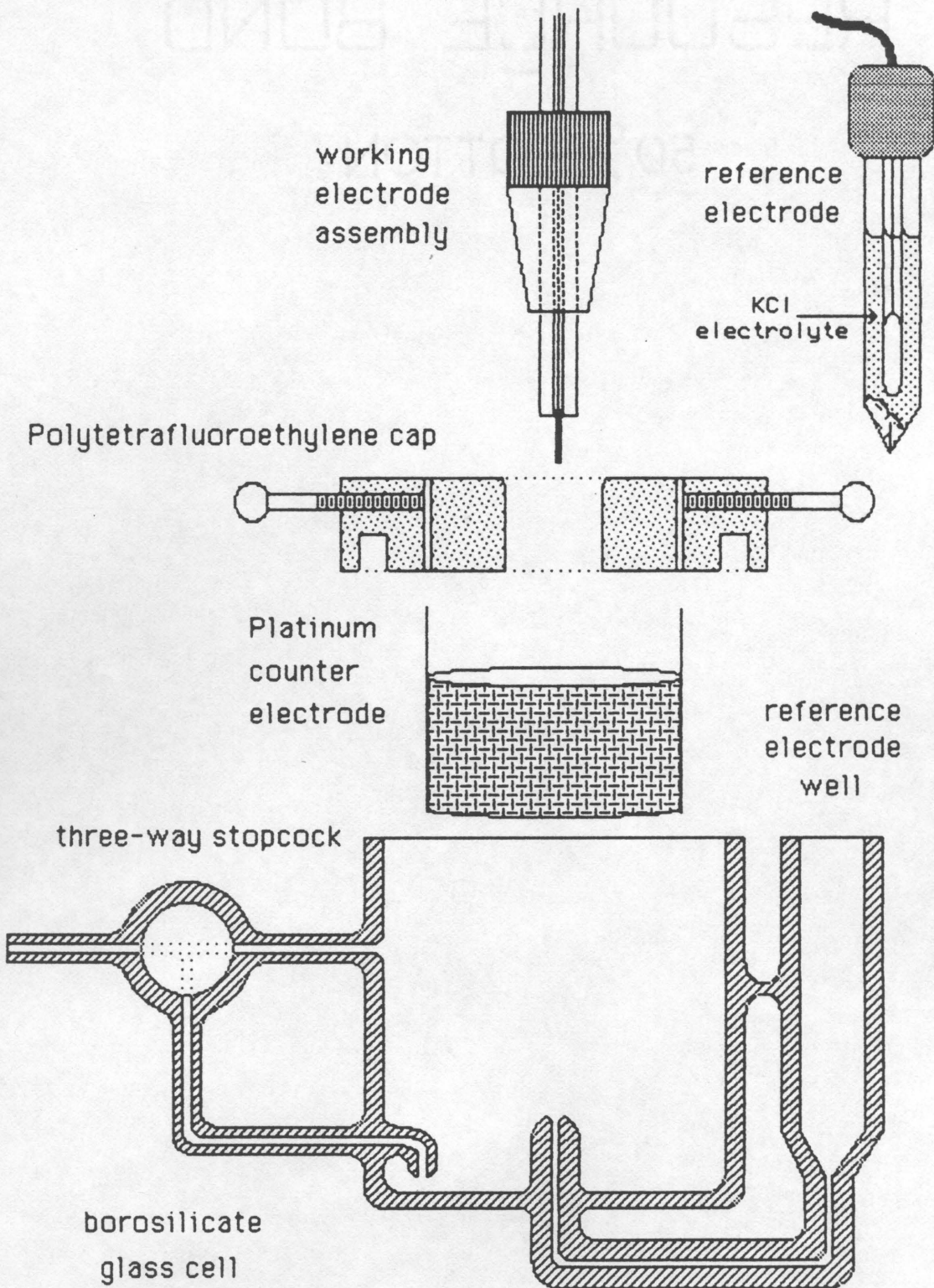
Figure 6.4 'Exploded' view of electrochemical cell

counter
electrode

working
electrode

reference
electrode

N₂
gas

electrolyte
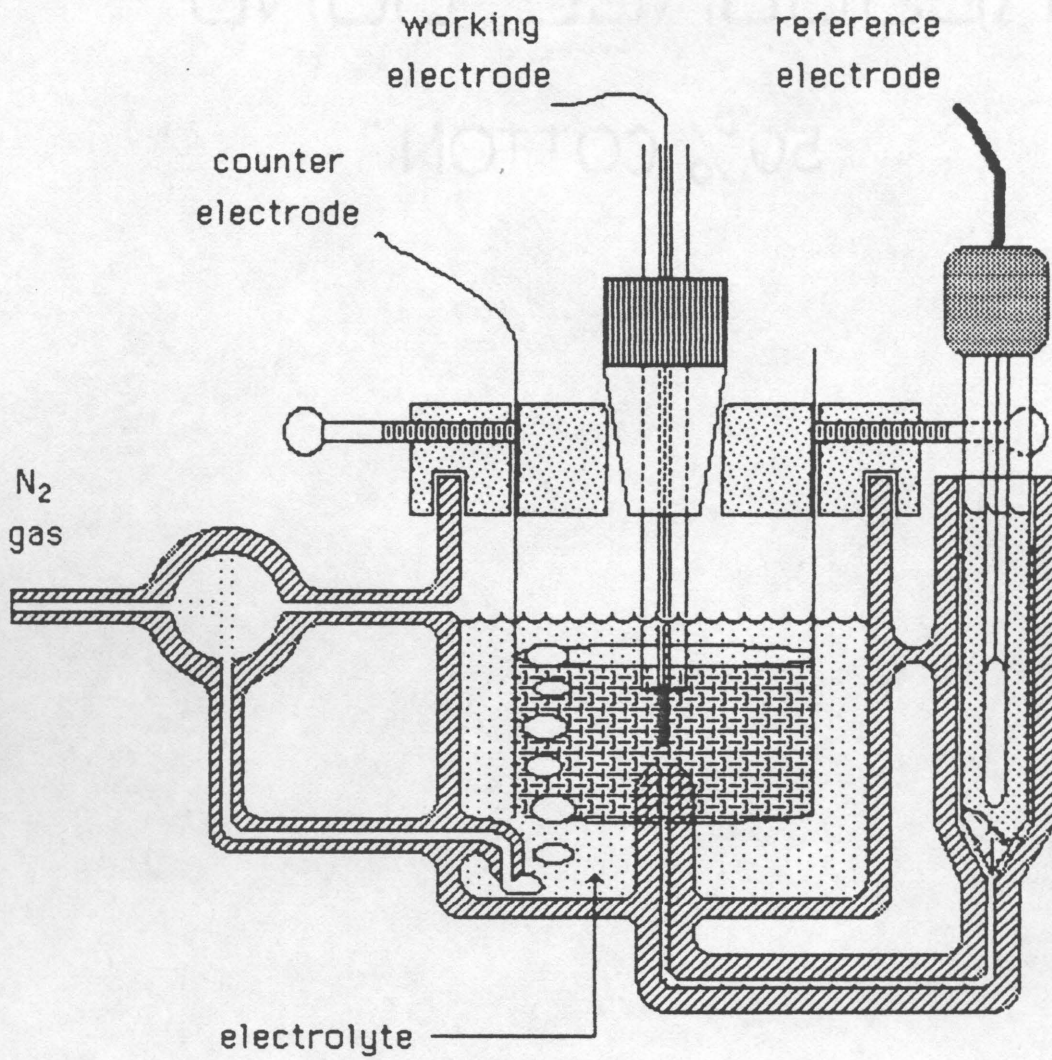
Figure 6.5 Assembled view of electrochemical cell
with Nitrogen sparging

was similarly constructed. The metal rod of the ring stand extended up through the lid, and particular care had to be taken to electrically connect the rod to the copper cage. This was done via a small "alligator clip lead". The cage was electrically connected to the shield on the cable leading to the potentiostat.

The supporting electronics for the electrochemical cell are shown in Figure 6.6 below.

Figure 6.6 Schematic view of electronic connections
to electrochemical cell.
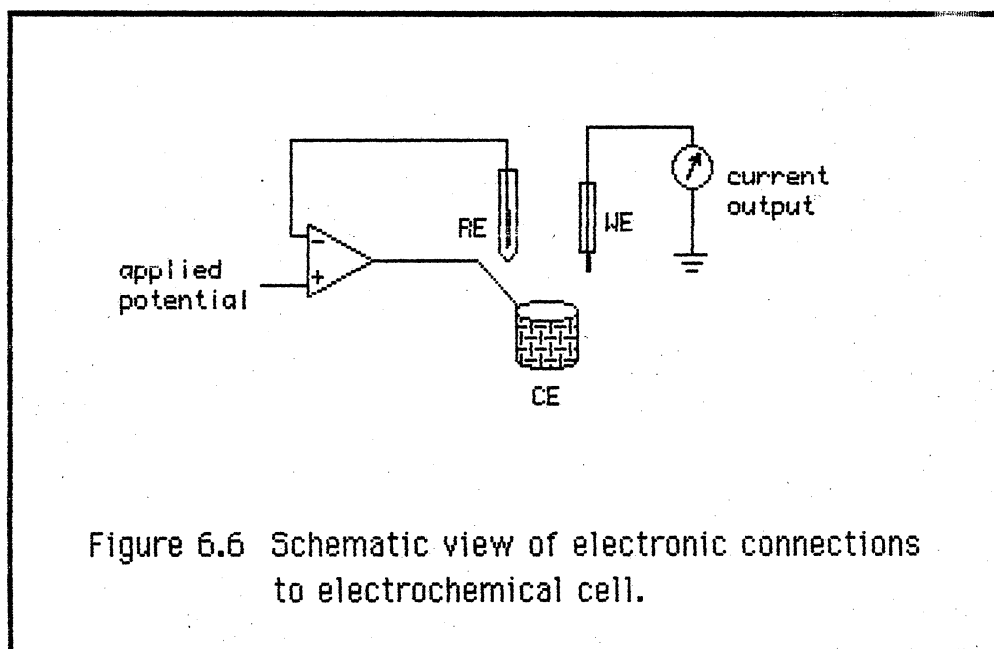
For both scanning voltammetry and the oscillogenic instrument, the potential of the working electrode (WE) is controlled and the current flowing to that electrode is measured. The device which performs this function is called a *potentiostat* by electrochemists. The potentiostat is comprised of three parts: a voltage control or regulator section, a current measuring section, and a power supply.

The schematic of the voltage control circuit is shown in Figure 6.7. The reference electrode (RE) is input to a voltage follower amplifier (U1c, input impedance $>10^{12}\Omega$). The amplifiers in all of the circuits of this work are FET input, bipolar output operational amplifiers (Texas Instruments' TL084 or TL094 quad op amp packages, Radio Shack). The output of the voltage follower buffer amplifier goes to the minus input of a high gain operational amplifier (pin 2 of U1a, gain > 15,000). The output of the buffer amplifier also is connected to the shield around the lead to the reference electrode as a *guard* circuit to further reduce leakage currents.

A standard differential amplifier (U1b) is used to buffer the *potential setpoint* of the potentiostat. This setpoint can come from an internal voltage divider or an external source (E_{ext}). The output of the differential buffer goes to the positive input of the high gain amplifier. Across the input terminals of the high gain amplifier (pins 2 and 3 of U1a), are a capacitor to improve stability, and two diodes (type 1N914) to protect the amplifier input from transients. Although the operational amplifiers are protected against output short circuits, the output of the high gain amplifier is also fused (0.25 ampere) to protect the electrodes and the rest of the circuit.

The output of the high gain amplifier (U1a) is the gain times the difference between the inputs. This output is connected to the counter electrode (CE) which in turn affects the potential of the electrolyte solution around the working electrode. The reference electrode, via the Luggin probe, measures the electrical potential of the electrolyte solution
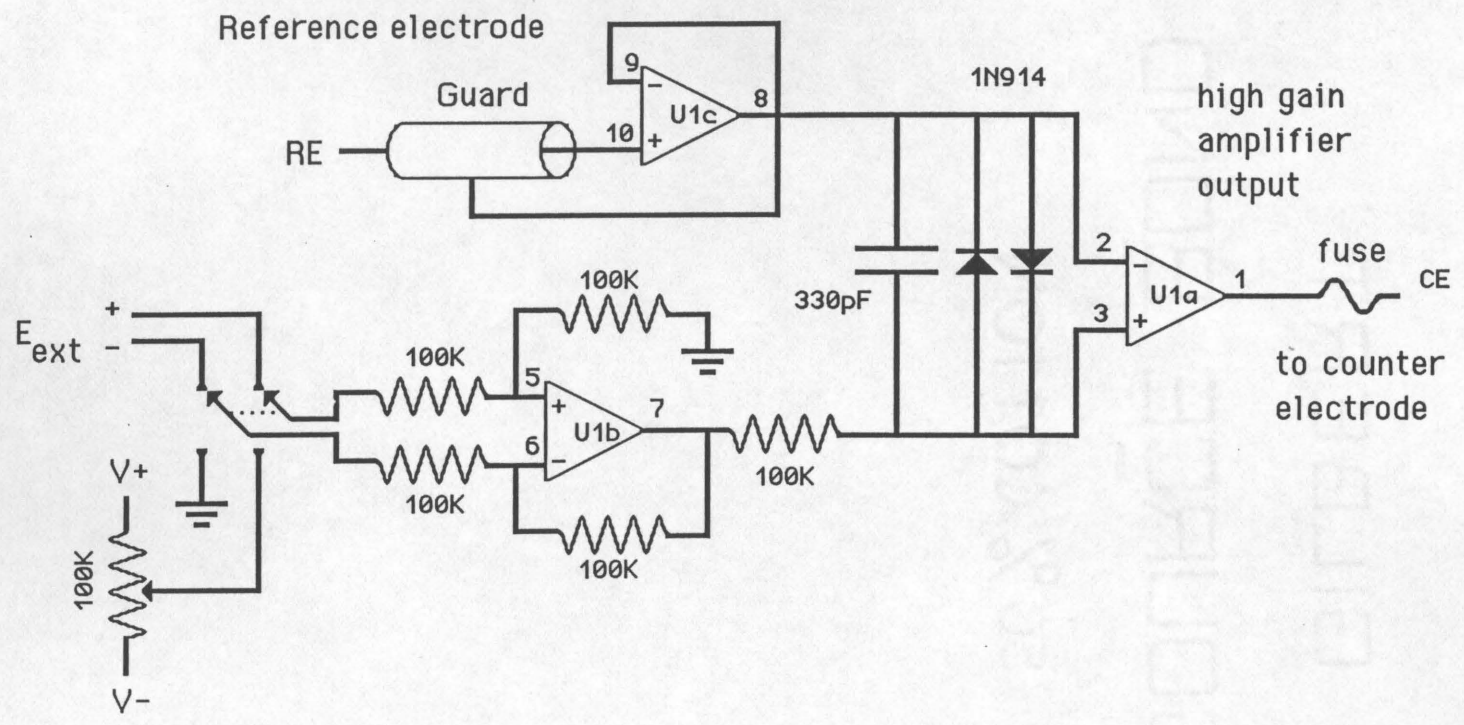
Figure 6.7 Potential output section of potentiostat

in the vicinity of the working electrode, thus closing the loop. The potential difference between the reference electrode and the setpoint is very small (<0.0001 volt) because of the high gain of the amplifier.

Figure 6.8 shows the current measuring circuit of the potentiostat. The working electrode is connected to the input of a low impedance current follower circuit (pin 13 of U1d, <60Ω on the most sensitive scale to <0.006Ω on the least sensitive scale). In the feedback path of the circuit, a rotary switch is provided to select the current range, from 1 volt/microampere to 10 volts/milliampere. Also in the feedback path is a 0.001 microFarad capacitor to improve the stability of the circuit. A meter is also included in the potentiostat to monitor various internal voltages, from top to bottom: (1) the signal proportional to the current, (2) the potential setpoint, (3) the reference electrode potential, (4) the counter electrode potential, (5) the output of the high gain amplifier, (6) the positive supply voltage, and (7) the negative supply voltage.

A possible power supply for the potentiostat is shown in Figure 6.9. The power line supply is fused and the output has series diodes for protection. A switch is provided to completely remove power from the potentiostat, and hence the electrodes. The power supply connections for the operational amplifiers are also shown.

The experiments were controlled by a microcomputer (IBM PC, IBM Entry Level Systems, Boca Raton, FL). The interface for voltage setpoint output was a commercially available *digital-to-analog* converter board (model DAC-02, MetraByte Corporation, 254 Tosca Drive, Stoughton, MA 02072) which was installed in the microcomputer. The connections and

current signal output

100
Current Range Select

1K

10K

100K

1M

0.001μF

current signal
330K

U1 pin 7 from E
37K

U1 pin 8 from RE
37K

from CE
330K

amp U1 pin 1
330K

from V+
330K

from V−
330K

meter select

polarity switch

meter
(50μA full scale)

13 Working Electrode
14 U1d 12

217
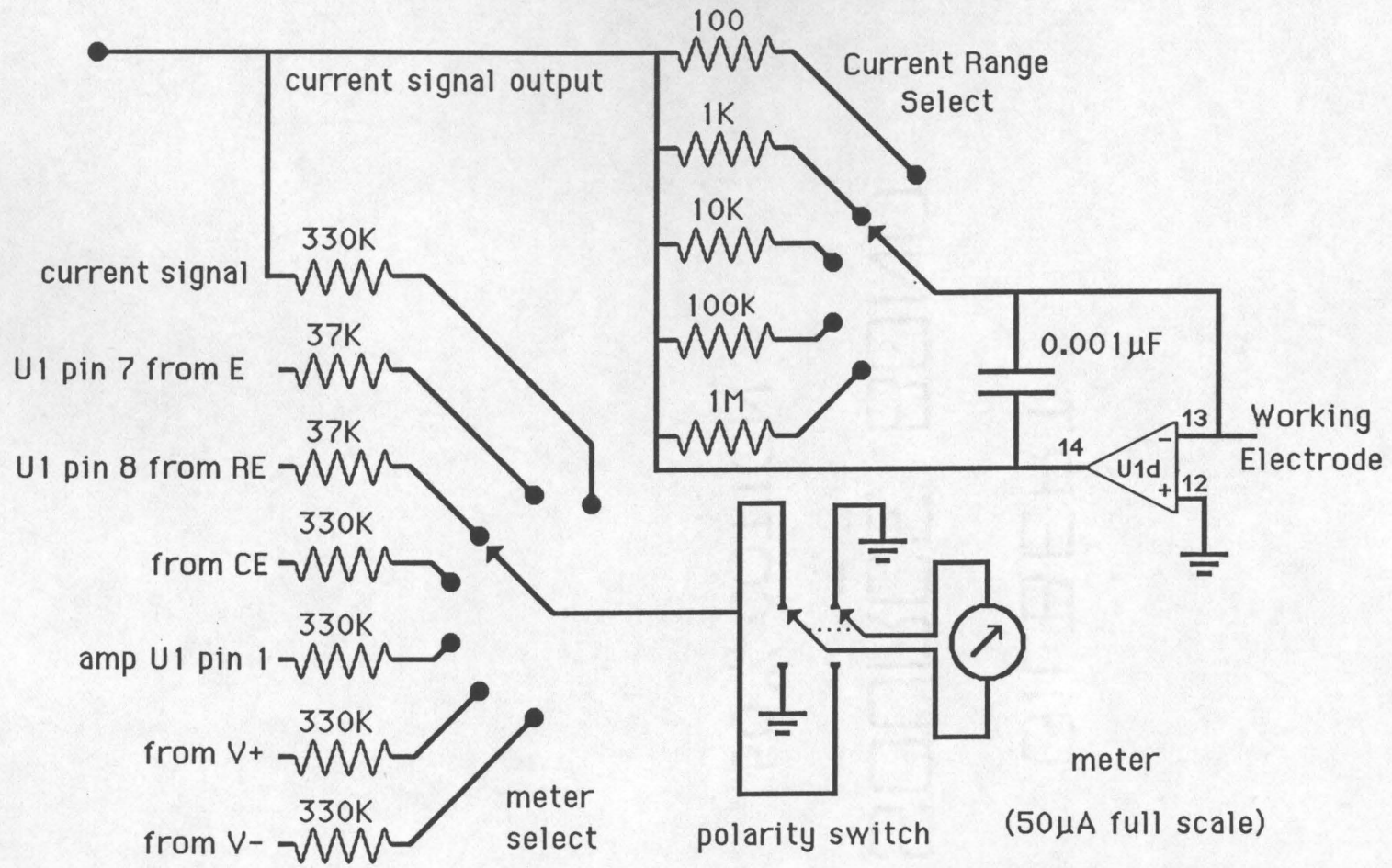
Figure 6.8 Current measuring and output

1 amp
120 VAC

1000μF

7815 (+15v) V+

47μF

LED

all op amps:

V+

14

+

- 11 TL084 or TL094

V- Quad op amp packages

1000μF 7915 47μF (−15v)

V−
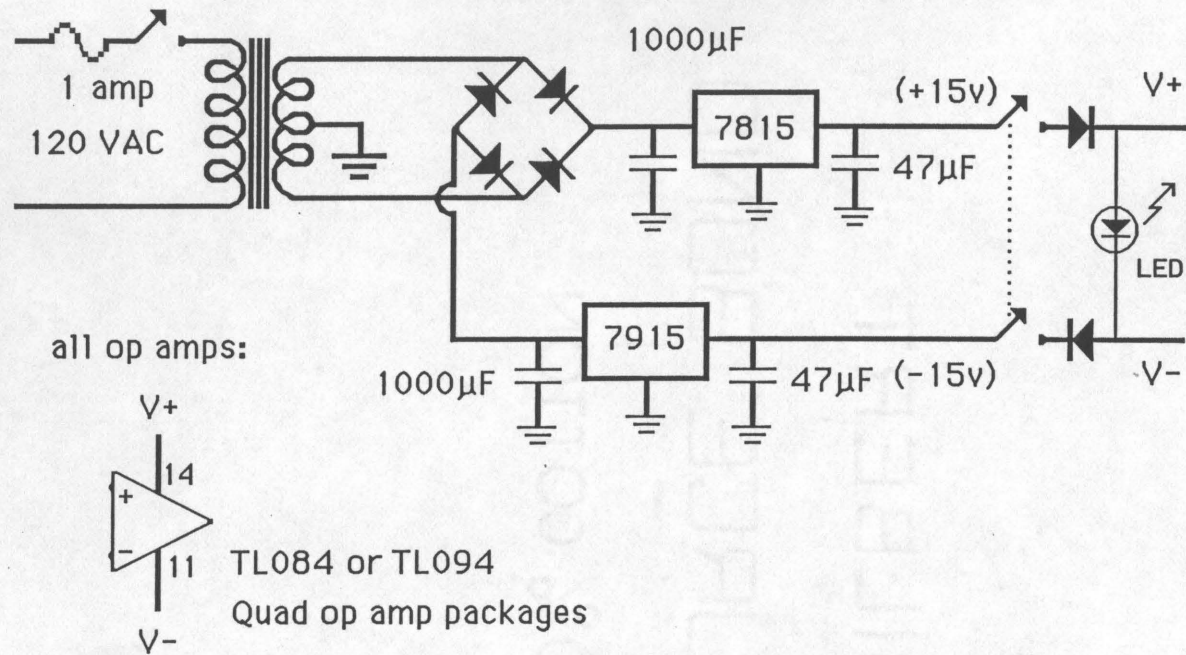
Figure 6.9 Power supply for potentiostat

DAC-02   Base address = 340 Hex

MetraByte

DB25S connector

13

Base Address
9876543

1

Calibration pots

IBM PC bus

analog
ground   10

e_out
23

jumper
for
range:
21&22

1 14

13 25

ribbon cable
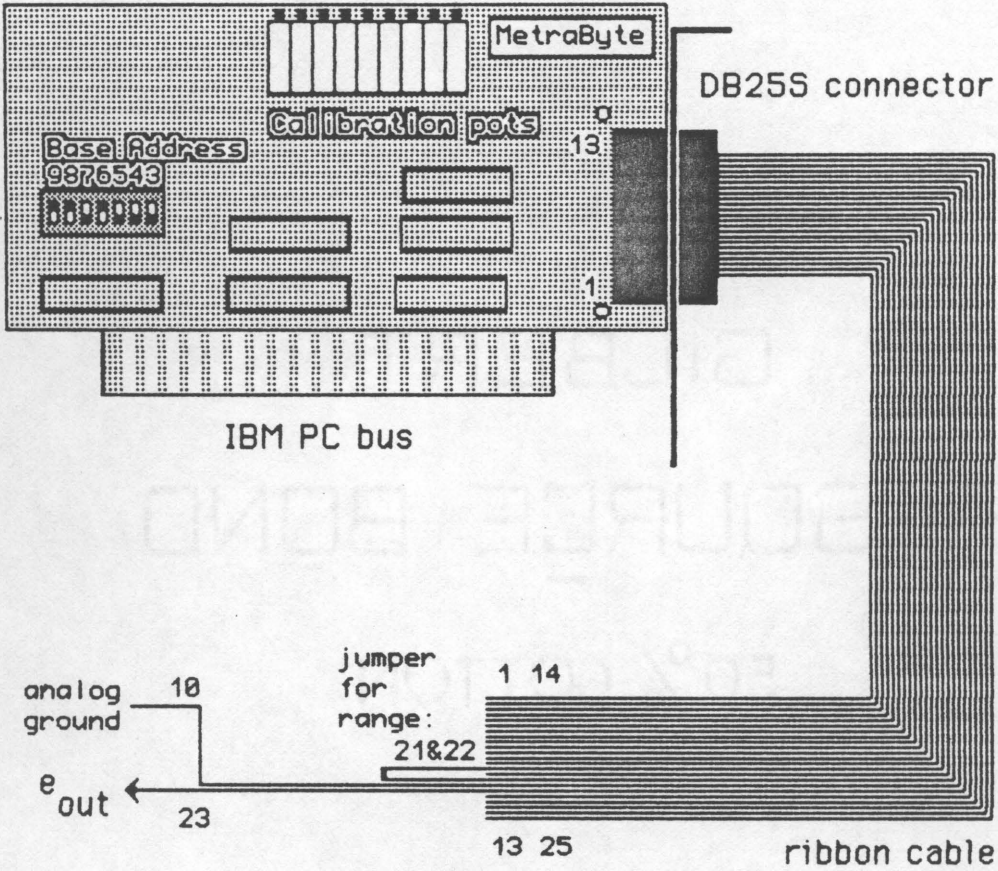
Figure 6.10 DAC-02 digital to analog converter

DASH-8   Base address = 330 Hex

DASH-8   Base address = 330 Hex
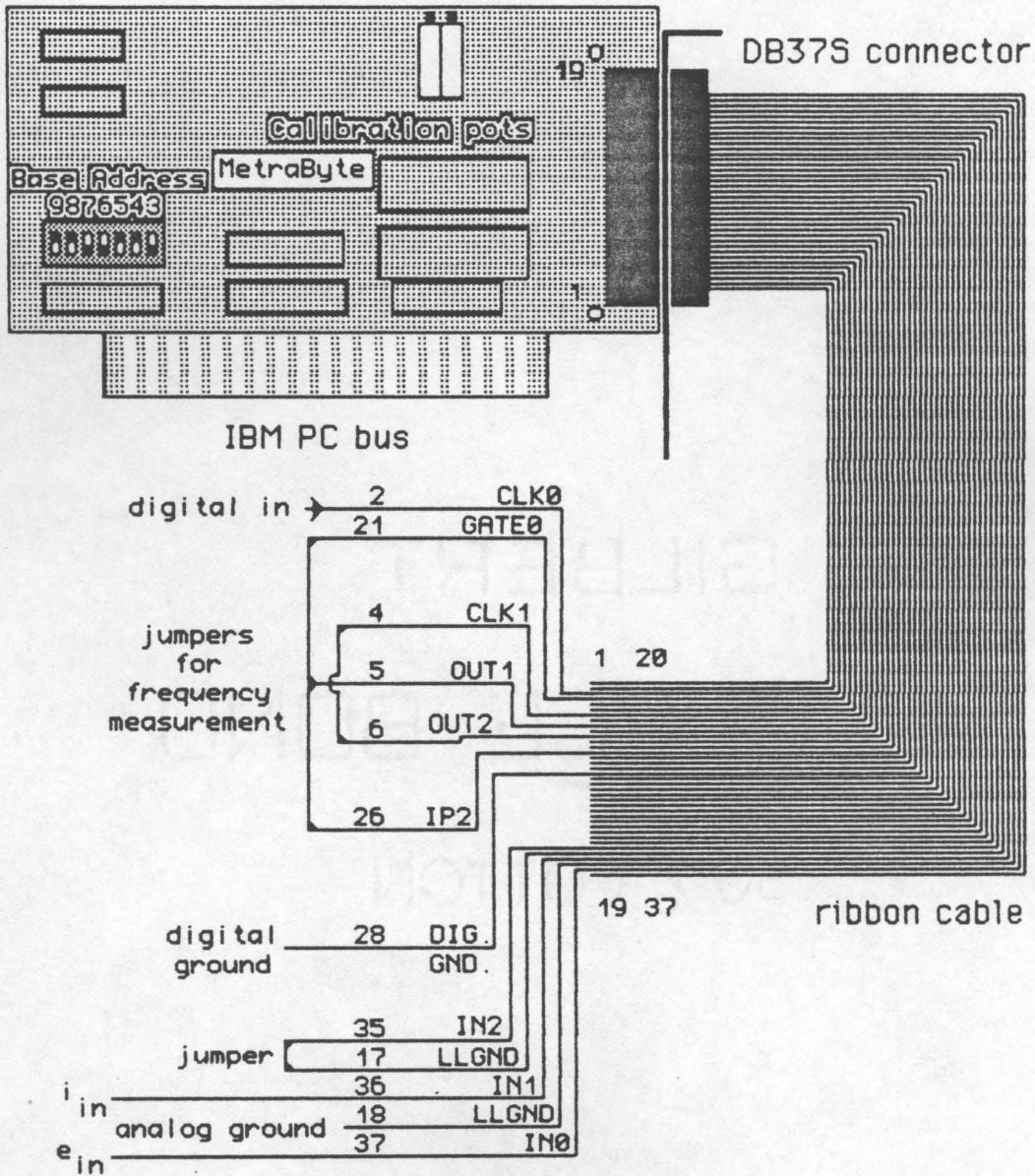
Figure 6.11 DASH-8 Analog to digital converter

configuration of this board are shown in Figure 6.10. The interface for measuring the current signal from the potentiostat (and the frequency of the oscillogenic instrument) was a commercially available *analog-to -digital* converter board (model DASH-8, MetraByte Corporation, 254 Tosca Drive, Stoughton, MA 02072) which was also installed within the microcomputer. Figure 6.11 shows the configuration of and connections to this board.

For experiments in scanning voltammetry, the connections shown in Figure 6.12 were used.

Figure 6.12 Experimental setup for scanning voltammetry
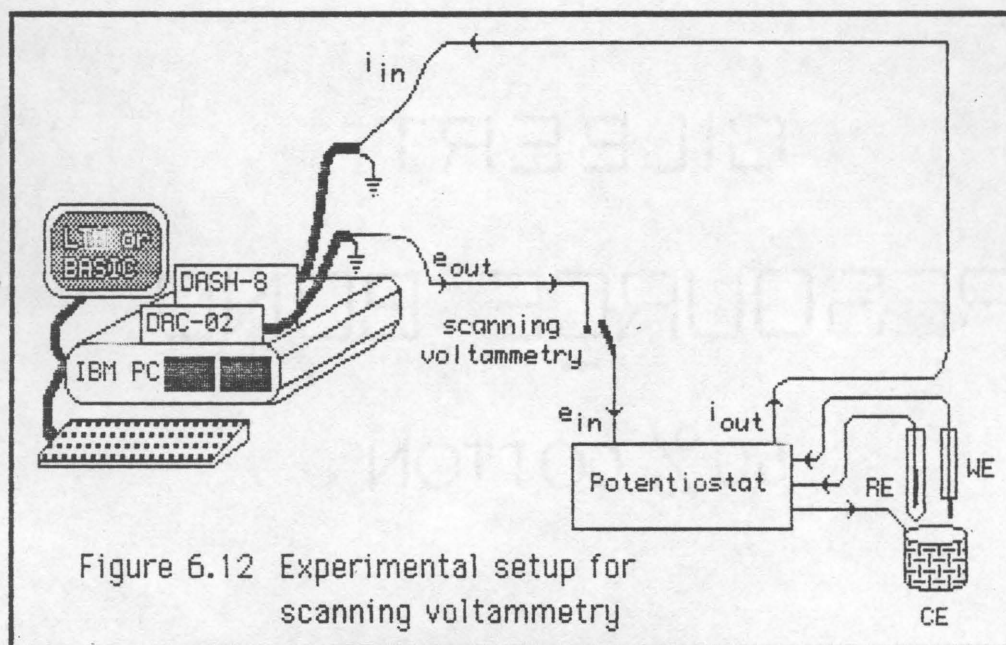
The experiments in scanning voltammetry were controlled by a commercially available program called: LABTECH NOTEBOOK (Laboratory Technologies Corporation, 255 Ballardvale Street, Wilmington, MA 01887). The configuration files for LABTECH NOTEBOOK are shown in listing 23 in

the appendix. The voltage waveform is controlled by a file called:
WF1.PRN. Listing 24 shows a short BASIC program that generates the
appropriate data for linear scanning voltammetry. The scan rate is
controlled by the values in the file and by the sample time setting for the
output channel in LABTECH NOTEBOOK. The output channel is channel 0 for
these experiments and is called an "open loop" output channel by the
LABTECH NOTEBOOK program. This program also graphs the results of the
experiment on the screen and outputs the data to a disk file for later
analysis. The quality of the graphical display (with an IBM Color/Graphics
Monitor Adaptor and an IBM Color Display) is sufficiently good to observe
gross errors in an experiment, such as no power to the potentiostat or a
disconnected working electrode. However, the quality of the graphical
display is not high enough to distinguish such problems as contaminated
solutions or excessive (>5% of full scale) noise in the signal from the
potentiostat.

The raw data, which the computer obtains from the interface
boards, differs somewhat from the conventions that electrochemists use
in reporting results. The signal from the current measuring circuit of
Figure 6.8 will be a positive voltage for a flow of electrons from the
working electrode. Electrochemical convention uses a positive current to
indicate a flow of electrons to an electrode, so the signal that the
computer records is inverted before graphing. The potential applied to the
working electrode is always stated in the electrochemical literature as a
potential "with respect to the reference electrode". As discussed above,
the potential of the reference electrode is virtually the same as the

"setpoint potential" applied to the potentiostat circuit of Figure 6.7. There is a contrast between the expressions "voltage with respect to a reference electrode" and "applied potential". The circuit of Figure 6.8 holds the working electrode at a *virtual* ground, that is, the action of the amplifier (U1d) is such that the working electrode is held very close (±0.0002 volts) to the ground potential of the power supply for the potentiostat. It is this same ground potential which is the reference for all other potential measurements. In reporting the results of an electrochemical experiment, one is interested in the potential of the working electrode versus the potential of the surrounding electrolyte. If the potential applied to the input of the potentiostat (the potential *setpoint*) is 1 volt, then the reference electrode potential will be very nearly 1 volt, while the working electrode is at a potential of 0, with respect to the circuit ground. But the potential of the working electrode with respect to the reference electrode will be −1 volt! Thus, the *potential setpoint input* to the potentiostat is the negative of the *potential with respect to the reference electrode*. The potentiostat circuit of Figures 6.7 and 6.8 is functionally the same as circuits in the literature [Macdonald 1977], where the difference between the potential applied to the potentiostat and the potential with respect to the reference electrode is not mentioned. So, the potential reported in the following graphs is the negative of the voltage output by the digital to analog converter card.

A typical result from the scanning voltammetry experiments is shown in Figure 6.13. The form of the curve and the position of the peaks
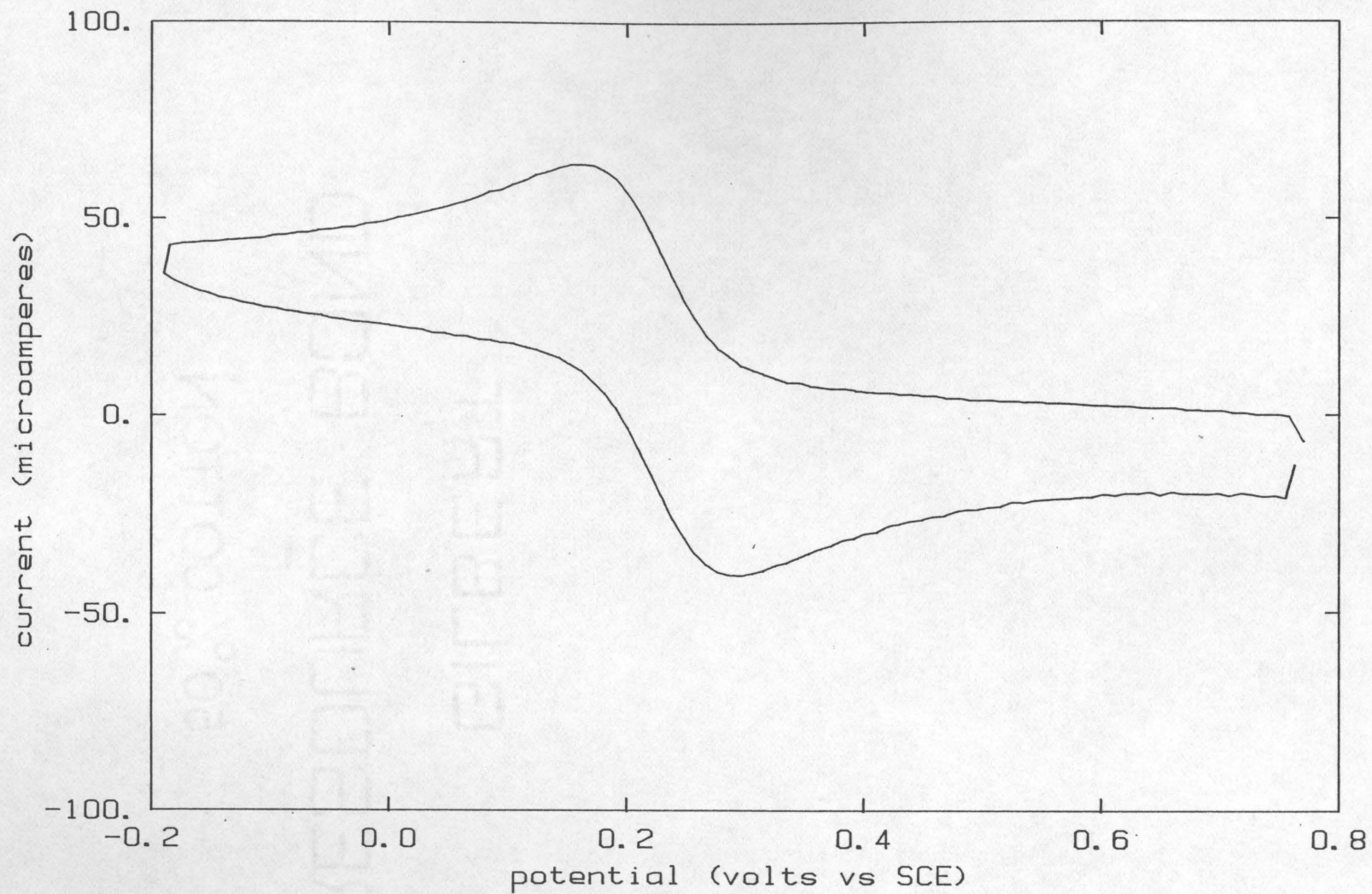
Figure 6.13   Typical results from scanning voltammetry
(ferricyanide, 6 millimolar)

are similar to results reported in the literature for this system [DeAngelis 1976, Mabbott 1983, Kissinger 1983], so the apparatus performed properly.

The experimental procedure for scanning voltammetry and for the electrochemical oscillogenic instrument are similar, and follow these steps:

1. Fill the clean glass cell, with the desired electrolyte for the experiment, to just below the level of the gas inlet at the top of the cell. Clamp the cell on the ring stand in the open Faraday cage.

2. Turn the stopcock such that the nitrogen can only pass to the sparging outlet at the bottom of the cell. Adjust the gas flow rate to 3-4 bubbles per second.

3. Take the cell cap (with the clean counter electrode) and place it on the cell. Make sure that the counter electrode is centered in the cell, and adjusted vertically such that the tip of the Luggin probe is centered between the top and bottom of the counter electrode. Put a ground glass stopper in the hole where the working electrode will be. Leave the nitrogen sparging for 15 minutes. This was found to be sufficient to remove residual oxygen from the solution such that it was not detectable as backgound current, to the precision of these experiments.

4. Prepare the microcomputer to run the experiment. This includes renaming the files from the previous experiment -- the LABTECH NOTEBOOK program will use the same filename for each run and write over the previous results, unless one takes the time to reconfigure the

experiment for each run.

5. Switch the potentiostat to its internal setpoint voltage and adjust the voltage to the starting voltage of the experiment.

6. Clean and install the working electrode in the cell. Check the alignment of the working electrode; it should be centered in the cell and just above the Luggin probe.

7. Put the reference electrode into the side well of the cell. Make sure that the level of KCl electrolyte in the reference electrode is above the level of the electrolyte in the rest of the cell; this is essential to avoid contamination of the reference electrode. The following operations must be done quickly, because the electrolyte from the reference electrode continues to flow into the test solution, and the clean surface of the working electrode will also change with time.

8. Attach the leads to the electrodes, and plug the leads into the potentiostat. The leads are provided with a plug which allows this connection to the potentiostat to be made in one motion. Put the top on the Faraday cage and attach what ever grounding connections are necessary.

9. Start the computer program and then switch the potentiostat to external voltage input. This is necessary because the LABTECH NOTEBOOK program does not allow the standby potential of the digital to analog converter card to be set, and the default output voltage is not often the desired starting potential.

10. When the program stops, disconnect the potentiostat. If replicate runs are being performed, then go to step 4.

11. Remove the reference electrode and place it in a standby container (a test tube of saturated KCl solution will do). Remove the working electrode and rinse it off with dilute nitric acid and then distilled water. Remove the cap and the counter electrode, rinse with distilled water, and store in a beaker of dilute nitric acid.

12. Drain the test electrolyte from the cell, rinse it with dilute nitric acid, and then with distilled water.

The first test system for the apparatus was the reduction of ferricyanide ion to ferrocyanide. The source of the ferricyanide was $K_3Fe(CN)_6$ (ACS certified grade crystals, Fisher Scientific Co., Springfield, NJ 07081). Following the procedure outlined by Benschoten [1983], the supporting electrolyte was a 1 molar aqueous solution of KNO_3 (ACS certified grade crystals, Fisher Scientific Co., Springfield, NJ 07081). Stock solutions of 1 molar KNO_3, and 10 millimolar $K_3Fe(CN)_6$ in 1 molar KNO_3 were made. Each solution of a dilution series was made by taking an aliquot of one solution by pipet, placing it into a volumetric flask, and adding the other solution to the mark. The working electrode was the platinum wire. Between runs, the electrode was cleaned by polishing with a paste of alumina (Conaco) and distilled water.

The scanning voltammograms of a series of solutions is shown in Figure 6.14, and a plot of the peak currents (on the negative going potential scan) versus concentration is shown in Figure 6.15. Thus the quantitative use of scanning voltammetry to measure concentration is demonstrated.
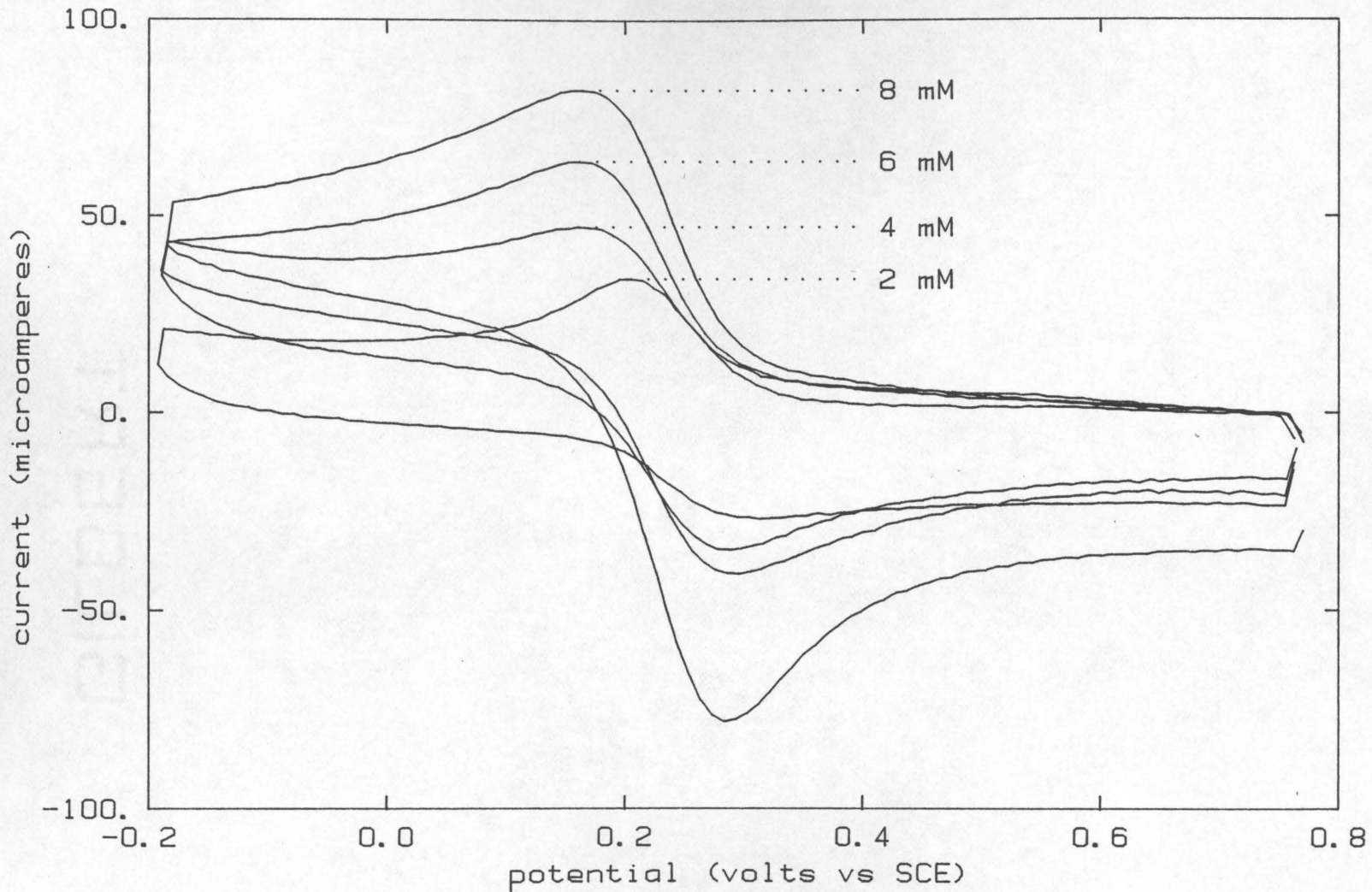
Figure 6.14 Series of scanning voltammograms
(ferricyanide, 2 to 8 millimolar)
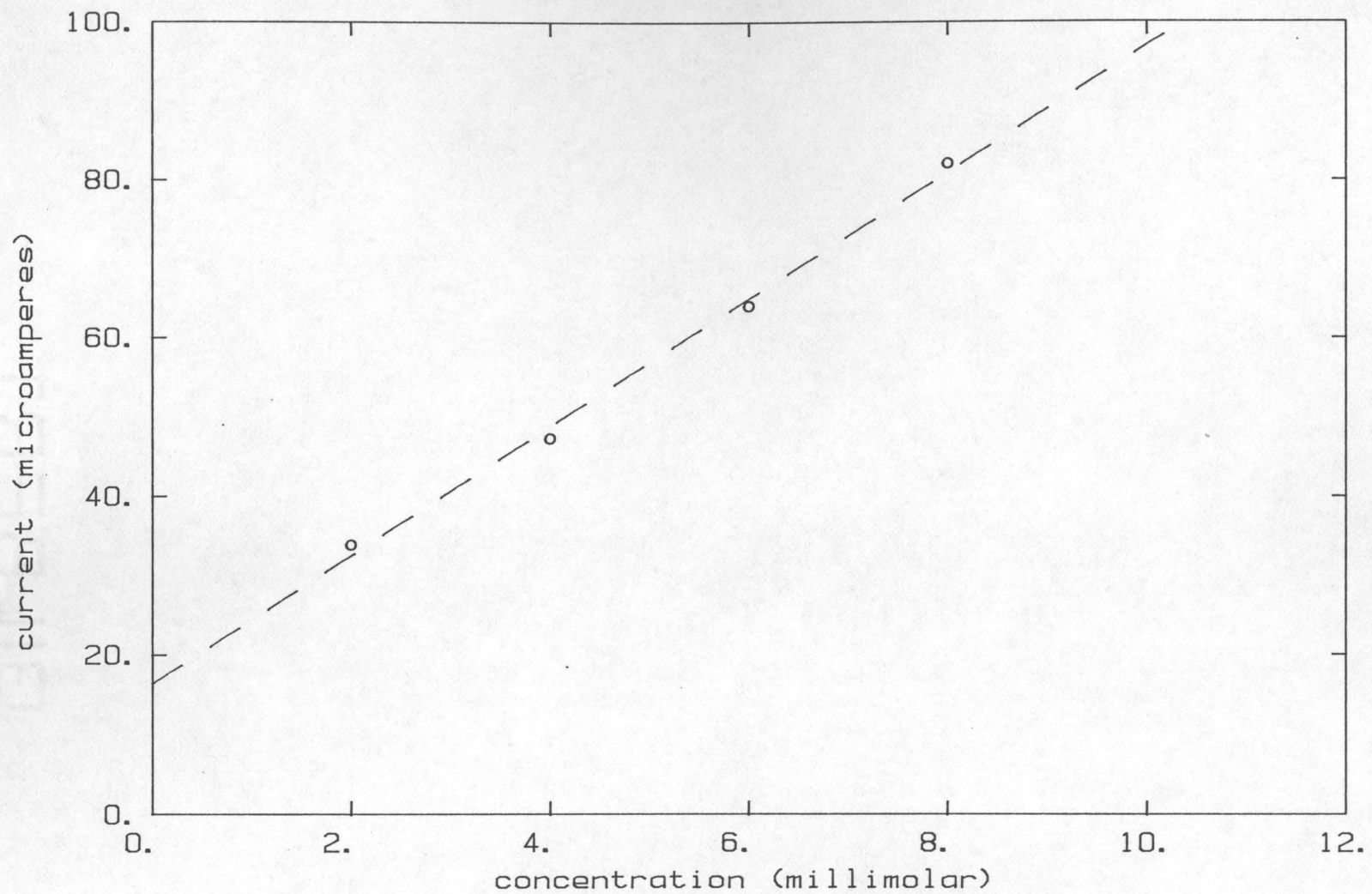
Figure 6.15 Peak scanning voltammetry current
as a function of concentration (ferricyanide)

Next, an attempt was made to synthesize an oscillogenic circuit for this electrochemical system. Following the design recommendations which resulted from the mathematical model of chapters 4 and 5 of this work, a simple oscillogenic feedback circuit was devised. Figure 6.16 shows a sketch of the first circuit tried.

Figure 6.16 Simple schematic of oscillogenic feedback
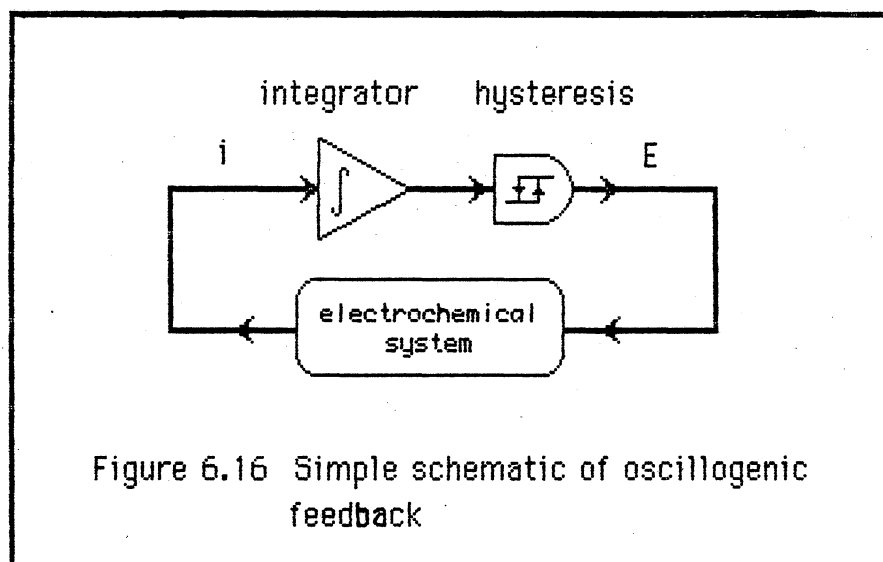
The integrator was constructed using an operational amplifier with a feedback capacitor (polystyrene, Radio Shack). The hysteresis element was a Schmitt trigger (CMOS 4093, Radio Shack). Since the output voltage of the integrator was not compatible with the input voltage of the Schmitt trigger, an operational amplifier was used as a level shifter to shift the voltage linearly. Since the output voltage of the Schmitt trigger was not the desired input to the electrochemical cell, another linear level shifter was used. This combined circuit is shown in Figure 6.17.

Figure 6.17 First oscillogenic feedback attempted
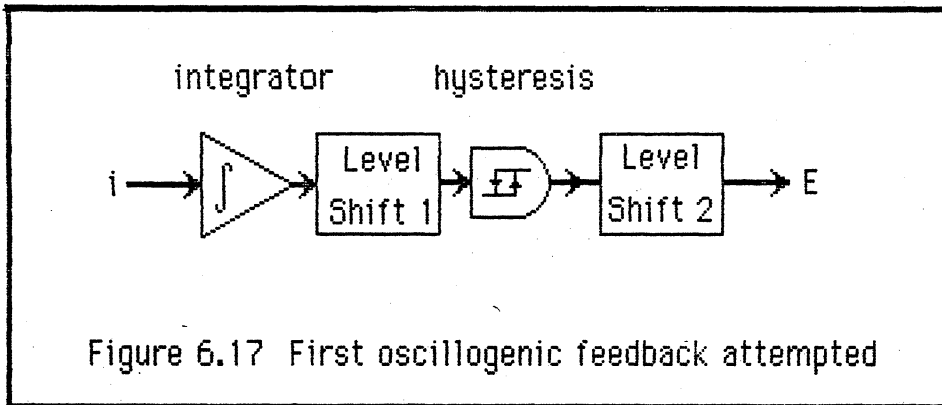
This circuit, with the electrochemical cell, would give bursts of oscillation, but it was a very unstable oscillator. The current output of the potentiostat, observed with an oscilloscope, indicated that the integrator output drifted to a high or low extreme and stopped the oscillation. Since analog integrators are known to have drift, a balancing circuit was added to the integrator. But, this did little to improve the oscillator. The major source of the drift seemed to be a background current in the electrochemical cell. So, the input to the integrator was changed to a potential representing the instantaneous current minus the average current. An operational amplifier was used as an active first order filter, and the system of Figure 6.18 was tried.

Figure 6.18 Improved oscillogenic feedback
(integral of the deviation of the current from average)
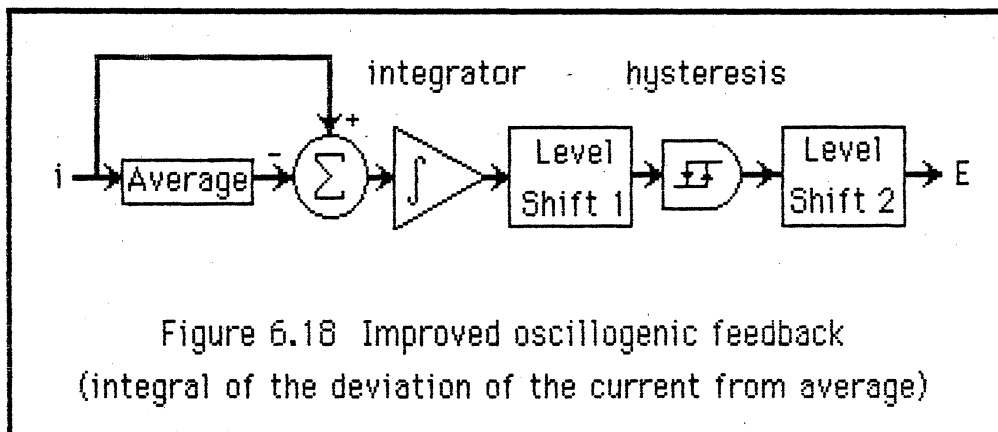
While this circuit improved the stability of the oscillator, it still drifted and stopped oscillating. Instead of taking tenths of seconds to stop, it stopped in seconds. By observing the current output of the potentiostat with an oscilloscope, a distinct trend was noticed in the pattern of the waveform, as shown in Figure 6.19.
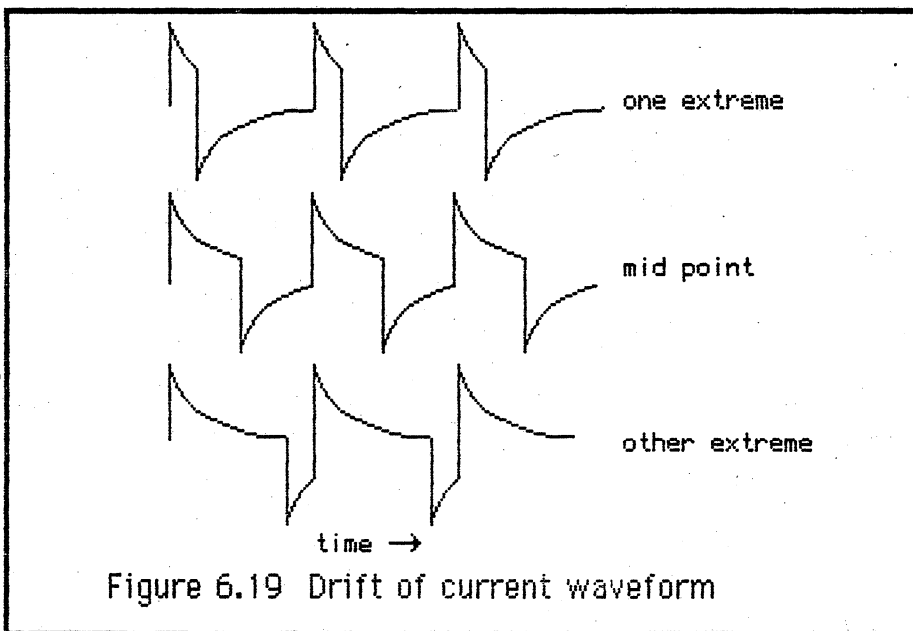
Figure 6.19 Drift of current waveform

The waveform started out at one of the extremes, slowly changed until it passed through the midpoint waveform, continued to the other extreme, and then stopped oscillating. The waveform could be made to start at either extreme depending on the setting of the average potential and the zero setting of the integrator. The drift seemed to be related to the balancing required earlier by the analog integrator circuit, but the amount of compensation needed changed with time. This suggested that some type

of dynamic integrator balance was required. What was changing with time in Figure 6.19 was the *duty cycle.*

The duty cycle of a wave form is defined as the percent of the total period of the wave that the wave is on the higher of two different modes. In the waveforms of Figure 6.19, there are two distinct modes: (1) start high and decay lower and (2) start low and decay higher. The duty cycle is the percent of the time that the wave is in mode number (1). At the top of Figure 6.19, the duty cycle is low. In the middle, the duty cycle is 50 percent. And at the bottom, the duty cycle is high. The required dynamic trim in the oscillogenic feedback circuit was added in the form of a *duty cycle controller*. A signal was developed which was a measure of the duty cycle. This signal was then fed into the integrator to trim the duty cycle appropriately. A sketch of the resulting system is in Figure 6.20.
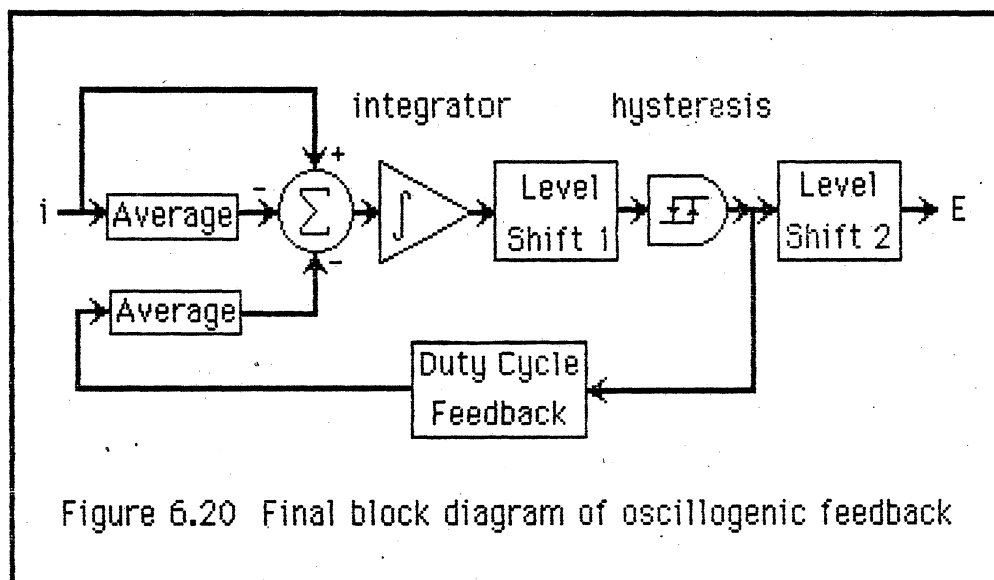
Figure 6.20 Final block diagram of oscillogenic feedback

The circuit based on the block diagram of Figure 6.20 did yield a stable oscillogenic instrument.

Figure 6.21 shows a test of the stability of the oscillogenic system. The first section of Figure 6.21 shows the stability of the electrochemical oscillogenic instrument with the ferricyanide ion and the platinum electrode. There is an initial transient, but the frequency becomes stable with some long term (tens of minutes) drift. This drift is the result of a common bête noire of electrochemistry: a changing electrode surface. In the second section of Figure 6.21, the oscillogenic feedback was disabled (the potentiostat was switched to an internal potential setpoint), and then in the third section the oscillogenic feedback was re-enabled. The oscillogenic system restabilizes but at a different frequency; the electrode surface was probably changed. To verify this, the platinum electrode was cleaned and two new tests attempted. For test number one, the oscillogenic feedback remained on continuously. For test number two, the working electrode was left disconnected until a measurement of the frequency was required, and the working electrode was connected only long enough to measure the frequency (by the use of a Hewlett Packard frequency meter). Figure 6.22 shows the results of the two tests; it is the electrochemical action that causes the change in frequency, not merely the exposure of the electrode to the electrolyte solution. If a liquid electrode is used (including mercury), then the electrode surface can be continually replaced by means of the system shown in Figure 6.23.
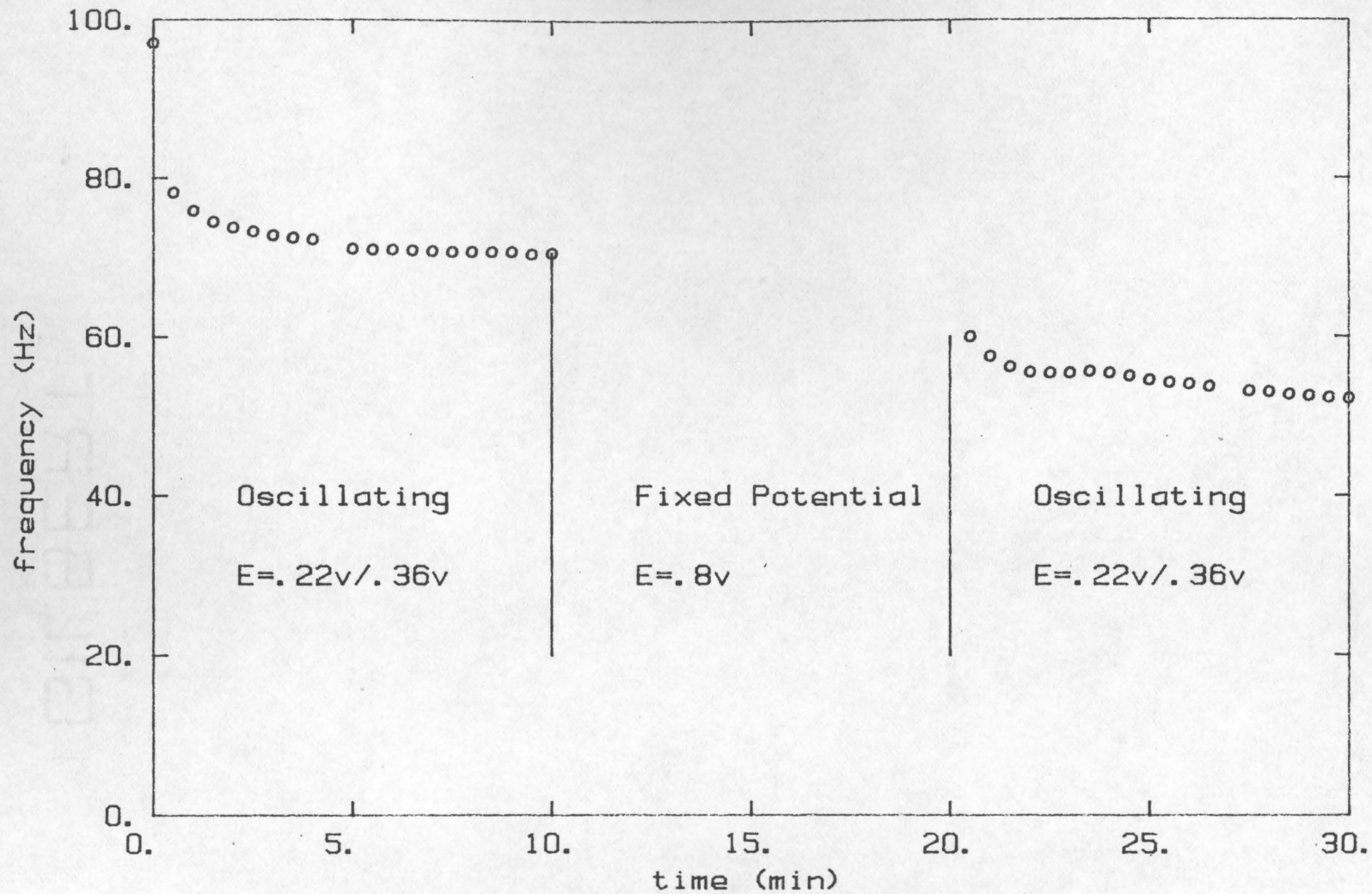
Figure 6.21 Stability of oscillogenic system, part 1
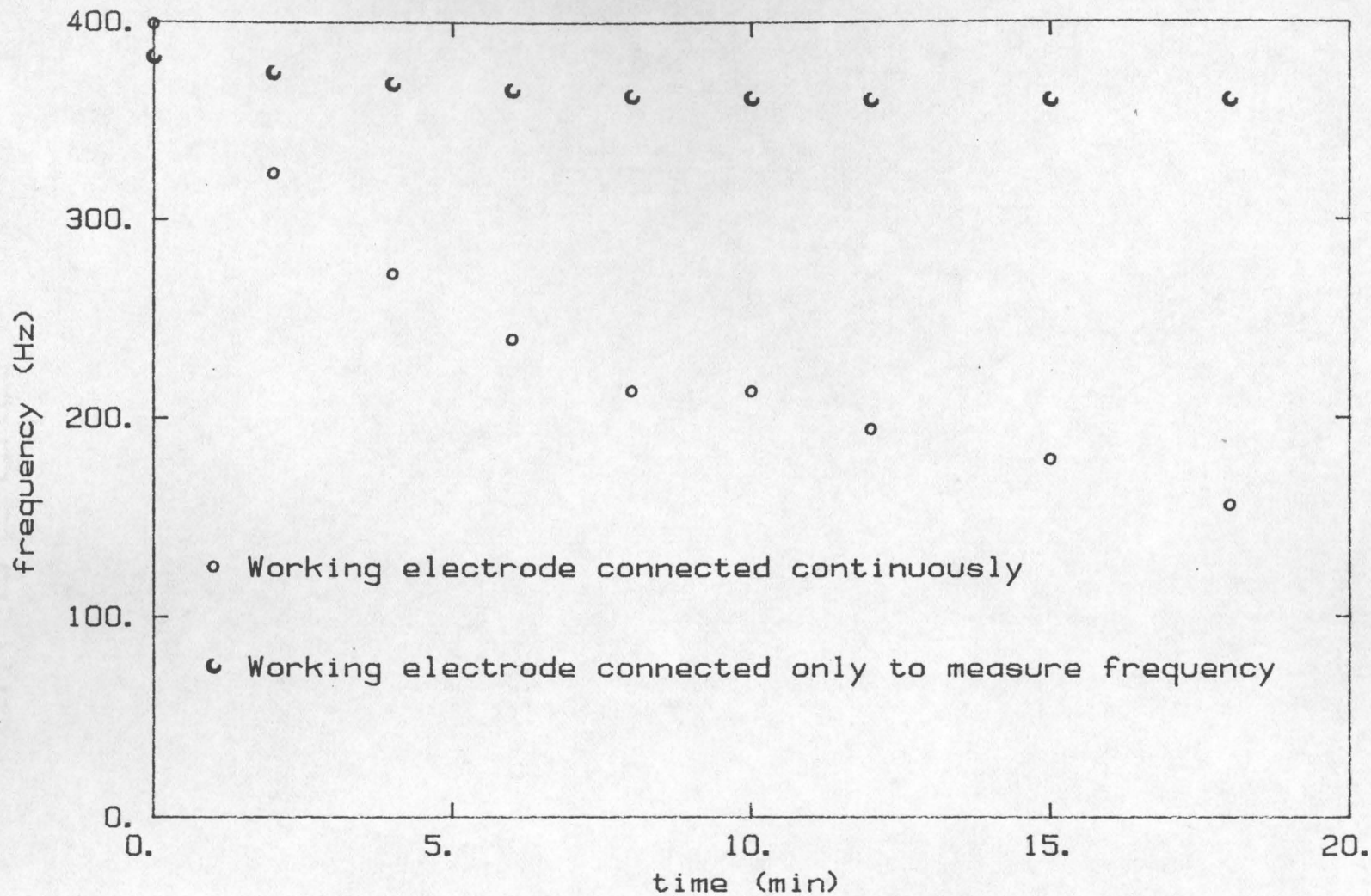
Figure 6.22 Stability of oscillogenic system, part 2
(ferricyanide with platinum electrode)

Figure 6.23 Trapped Mercury Drop Electrode
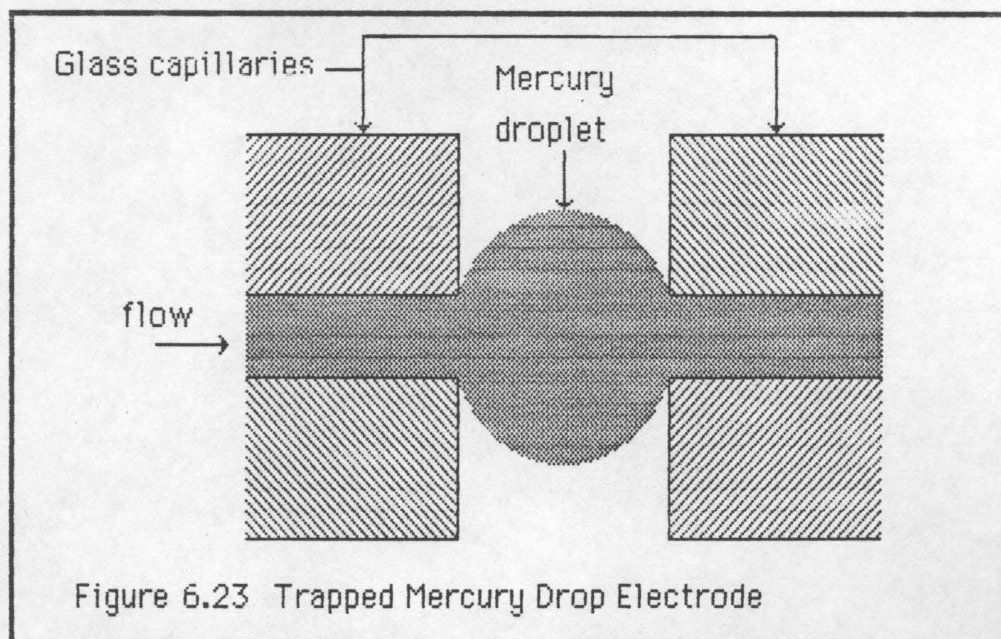
For this work, it was found that if the measurements were made rapidly enough (within a number of minutes), then the oscillogenic instrument was stable enough.

The detailed operation of the successful oscillogenic circuit can now be explained. Each block in Figure 6.20 has an electronic circuit associated with it. Beginning from the left, the first blocks are the "averaging" blocks. Since these blocks are linear, they can be and were combined into one electronic circuit, which is shown in Figure 6.24.

Figure 6.24 Averaging circuit
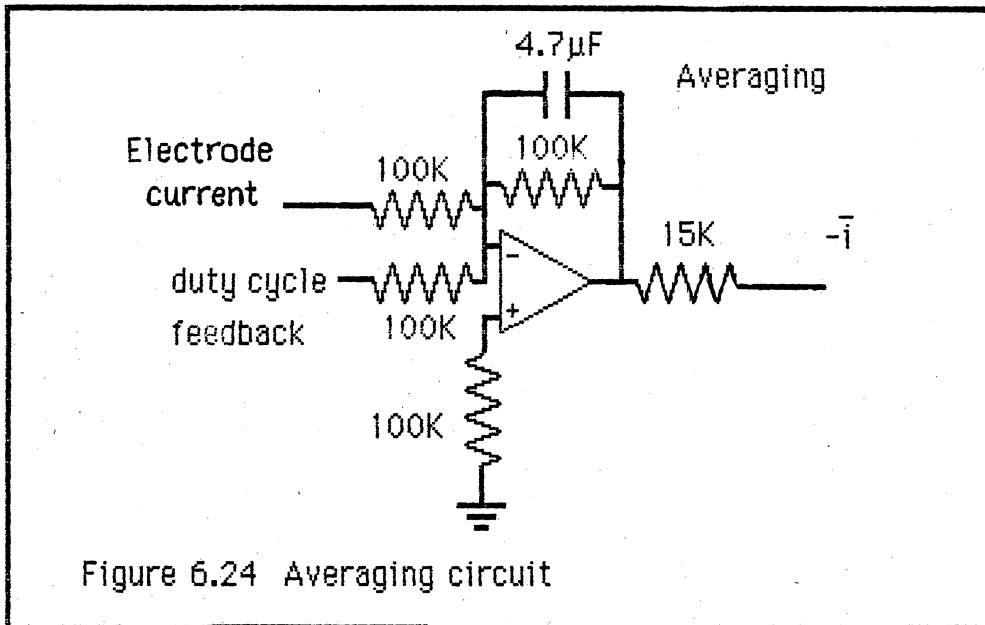
This circuit is a simple first-order summing filter; the time constant is 0.47 seconds. The next block in the block diagram is the summing integrator; the circuit is shown in Figure 6.25.

Figure 6.25 Integrator
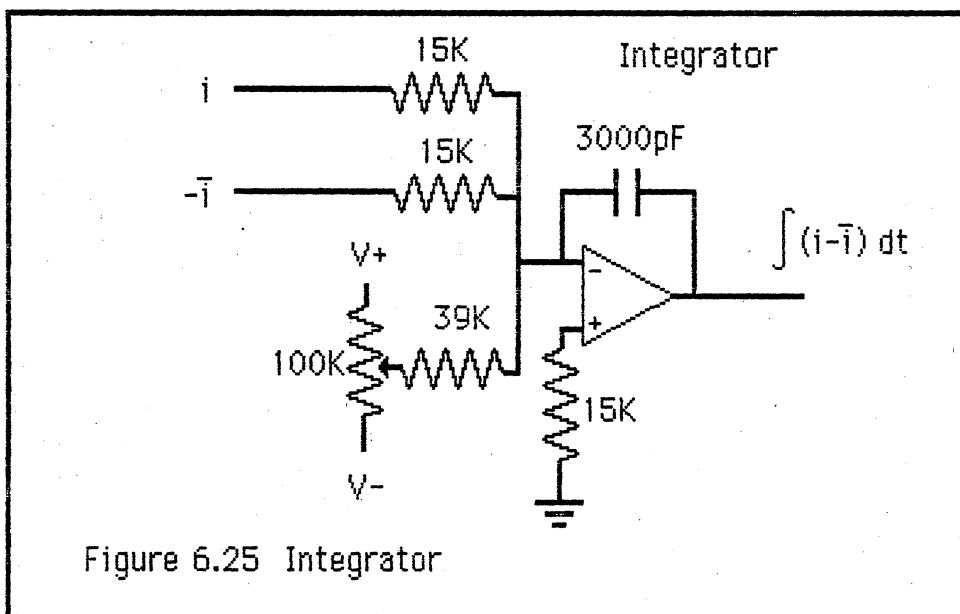
The 3000 picoFarad capacitor is a polystyrene capacitor used for its low leakage characteristic. The potentiometer is a balance adjustment; if desired, it is adjusted for minimal drift in the output when the input is zero. The output of the integrator must be scaled or level shifted to provide the proper input signal for the Schmitt trigger. The circuit of Figure 6.26 does this.

Figure 6.26 Level shift from integrator
to Schmitt trigger
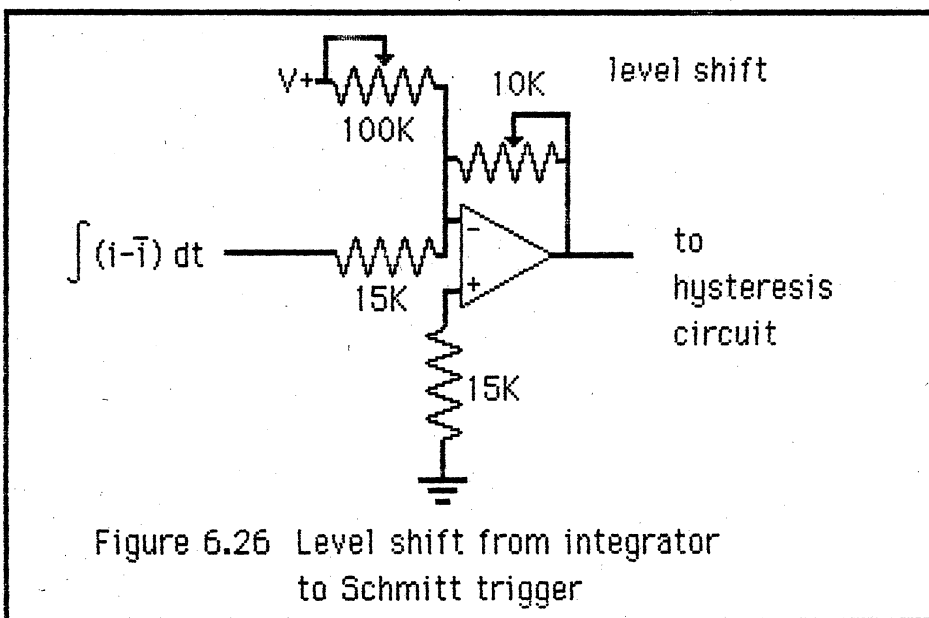
The two potentiometers are adjusted until an input of +10 volts will give an output of -10 volts and an input of -10 volts will give an output of 0 volts. The hysteresis element, the Schmitt trigger, was a commercially available circuit the CMOS 4093 quad Schmitt nand gate. It is much easier to use a pre-designed device than to construct one out of an operational amplifier. A CMOS (Complementary-Metal-Oxide-Semiconductor) device was chosen, rather than a TTL (Transistor-

Transistor-Logic) device, because the power supply range of the former is much higher (the same supplies as the operational amplifier can be used). Figure 6.27 shows the resulting circuit.

Hysteresis

Note:
Vpp = 0 volts,
and Vss = -15 volts

from
level
shift
1

to
level
shift
2

to duty cycle feedback

Figure 6.27 4093 Schmitt trigger circuit
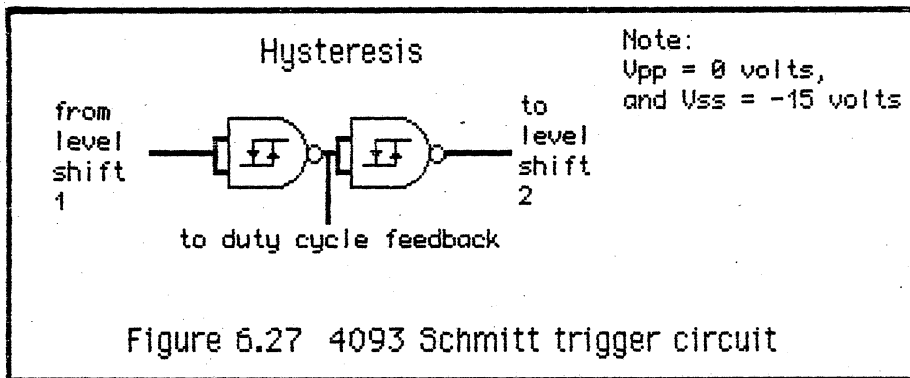
The power supply connections in this circuit are unusual, but this made the overall circuit design easier. The second level shifter is required to convert from the Schmidt trigger logic levels to the actual potential applied to the electrochemical cell. Figure 6.28 shows the actual circuit.

from
microcomputer
D-to-A

1K

1K

Level Shift

from
hysteresis
circuit

100K

Electrode
Potential

100K

Figure 6.28 Level shift from Schmitt trigger
to potential output to cell
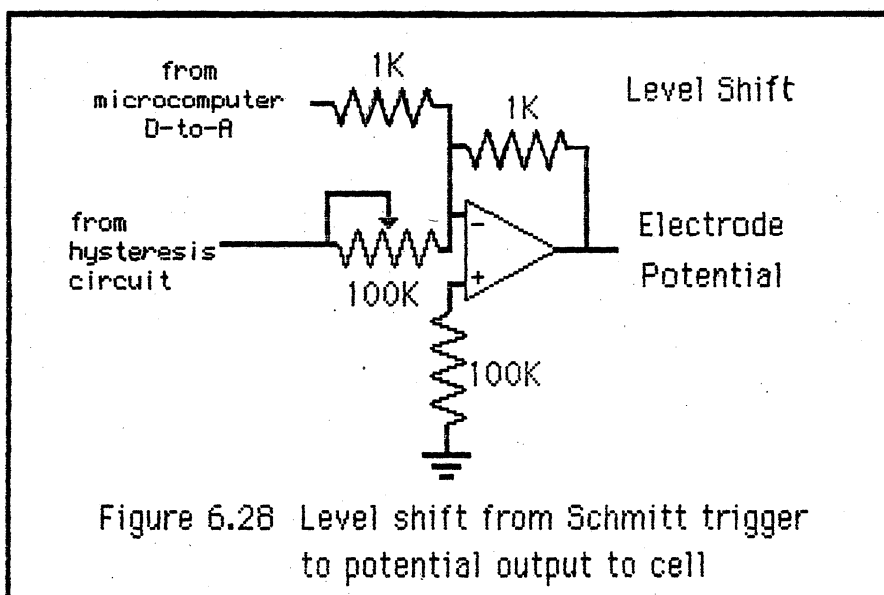
The variable resistor adjusted the voltage step size (ΔE), and the microcomputer digital-to-analog converter set the average voltage (E_{avg}).

The final element, the duty cycle controller, is shown in Figure 6.29.

Figure 6.29 Duty cycle feedback circuit
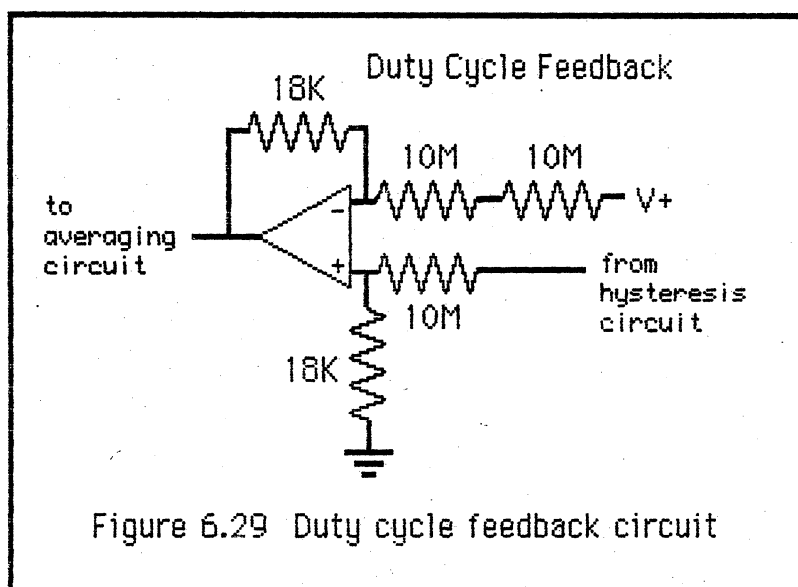
This circuit takes the logic level of the Schmitt trigger and subtracts an average value for the duty cycle desired (50%). This difference is then an error signal. The gain of this circuit is very low (0.018); the adjustment to the integrator is small, a *trim*. The total circuit for the oscillogenic feedback is shown in Figure 6.30.

A concentration series was used to compare the oscillogenic instrument to the accepted technique of scanning voltammetry. The average potential was scanned through a range of values, while the resulting frequencies were measured. The digital-to-analog converter
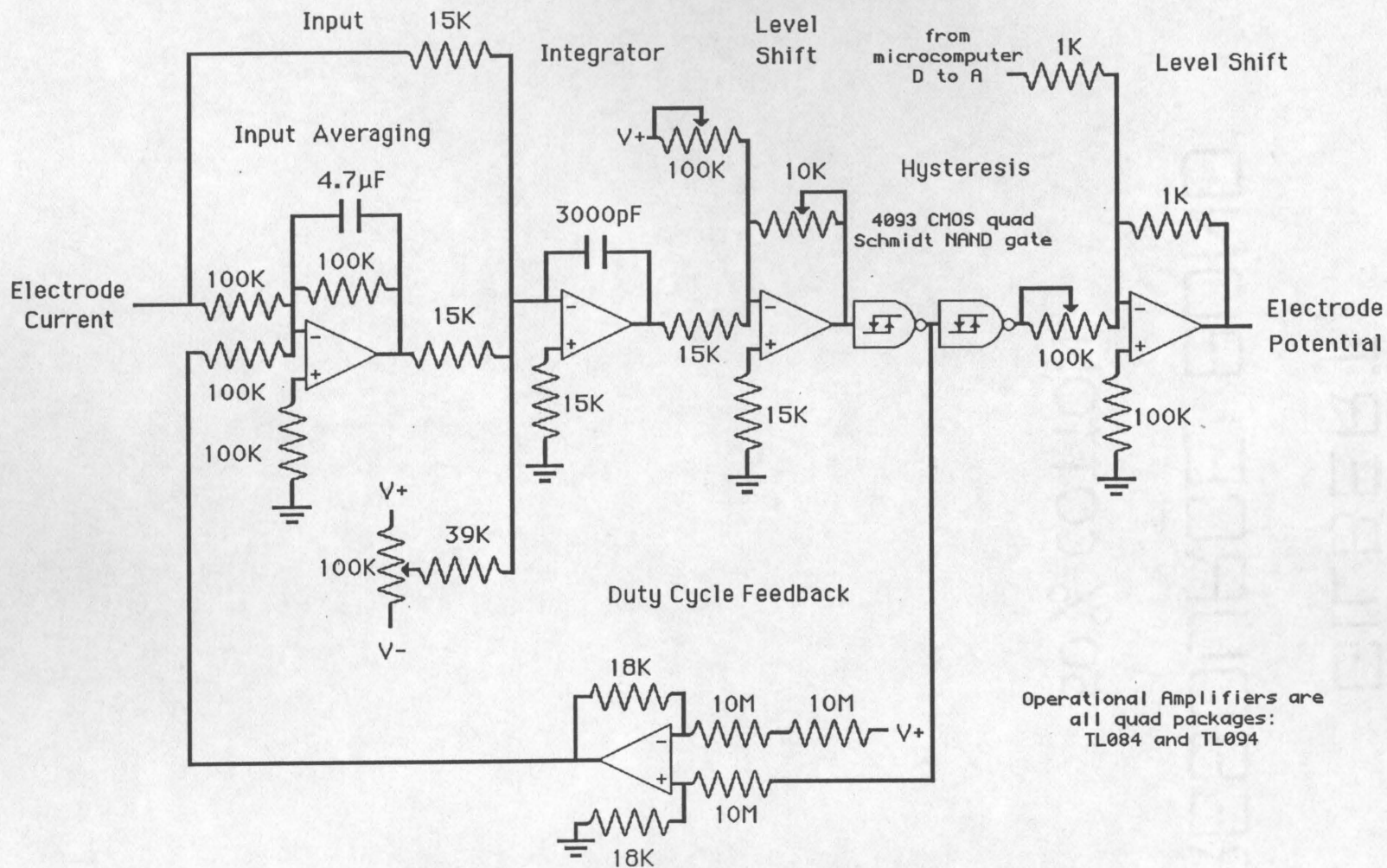
Figure 6.30 Complete oscillogenic feedback circuit

board had a digital input capable of measuring frequencies, but the LABTECH NOTEBOOK program could not make use of this ability. So a BASIC program was written to step the average potential and record the frequencies. This program is in listing 25 in the appendix. The connections between the interface boards in the microcomputer, the oscillogenic feedback circuit, the potentiostat, and the electrochemical cell are shown in Figure 6.31.
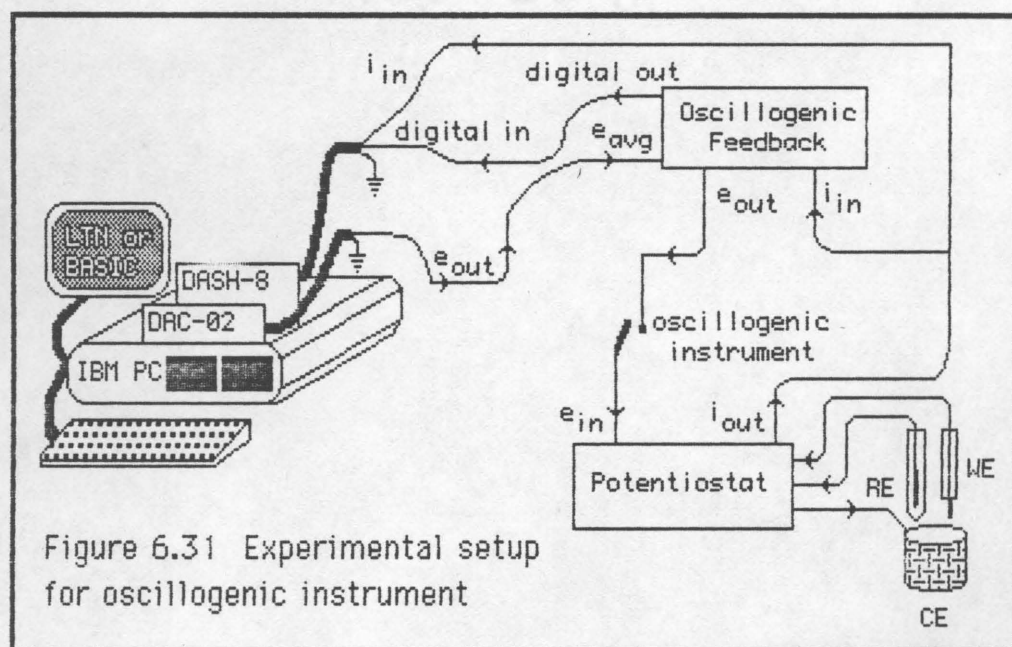
Figure 6.31 Experimental setup for oscillogenic instrument

The results of scanning the average applied voltage used by the oscillogenic feedback, and measuring the frequency, are shown in Figure 6.32. These curves have a striking symmetry. There is a peak, as predicted in Table 5.3 and Figure 5.41, and that peak is proportional to the concentration. The peak frequencies are shown as a function of concentration in Figure 6.33. Compare these results to those of Figure
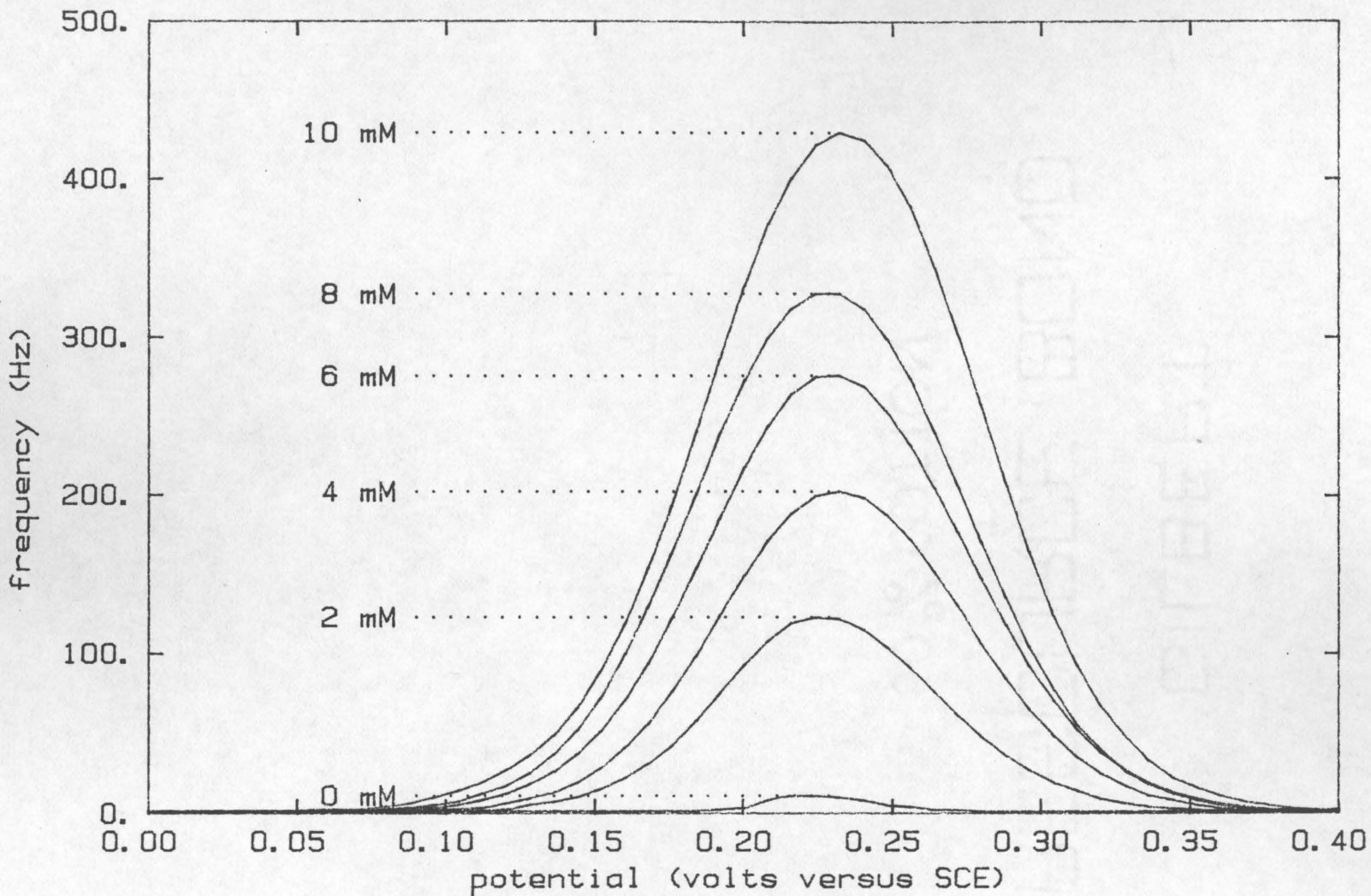
Figure 6.32 Electrochemical Oscillogenic Scan
(ferricyanide with platinum electrode)
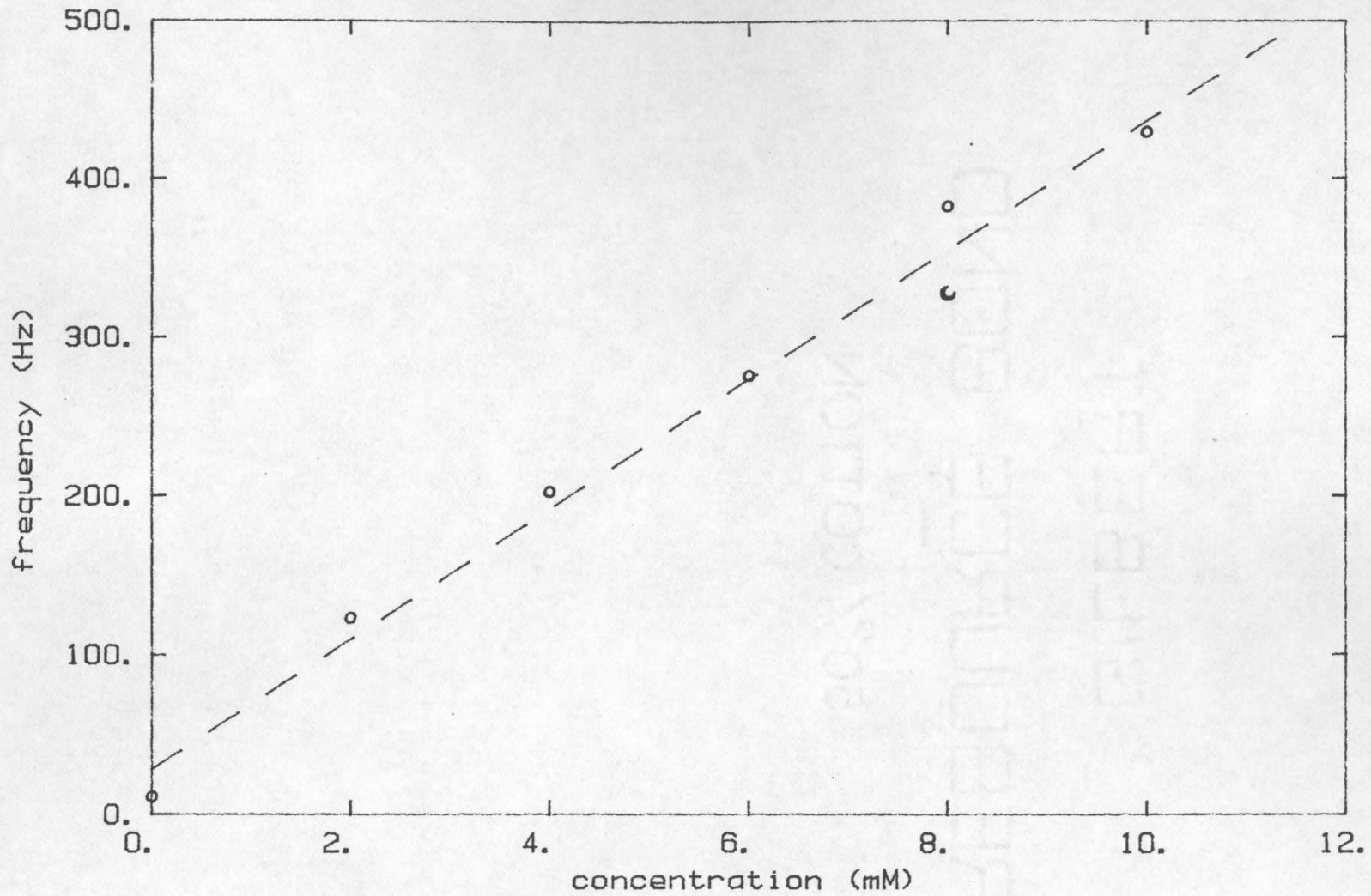
Figure 6.33 Electrochemical Oscillogenic Scan
Peak frequency as a function of concentration.

6.15; the oscillogenic technique gives similar results when measuring the peak frequency rather than current; as expected from the discussion of the mathematical model in chapter 5.

The symmetry of the curves of frequency versus average potential can be explained based on the classic Nernst equation. The electrochemical reaction of the ferricyanide ion is generally considered to be a "reversible" reaction, that is, a very fast reaction [Benschoten 1983, Kissinger 1983]. As such, the concentration at the surface of the electrode should follow that predicted by the Nernst equation [Macdonald 1977]. The current which flows on a small step change in applied potential will be proportional to the change in concentration. Since the frequency of the oscillogenic system will increase as the current increases (as discussed at the end of chapter 5), the frequency will also be proportional to the change in concentration at the surface. Thus, the frequency should be proportional to the derivative of the Nernst equation with respect to potential. The Nernst equation is [Macdonald 1977]:

$$E = E_0 + \frac{RT}{nF} \ln \left[\frac{c_A}{c_{A,bulk} - c_A} \right] \qquad [6.1]$$

Using dimensionless variables and rearranging:

$$f = \frac{e^P}{1 + e^P} \qquad [6.2]$$

where P is the dimensionless potential as defined by equation 5.5:

$$P = (nF/RT)(E-E_0) \qquad [5.5]$$

and f is dimensionless concentration as defined by equation 4.15

$$f = C_A/C_{A,bulk} \tag{4.15}$$

The derivative of the dimensionless concentration with respect to dimensionless potential is:

$$\frac{df}{dP} = \frac{e^P}{[1 + e^P]^2} \tag{6.3}$$

A graph of the derivative as a function of the potential is shown in Figure 6.34, which shows the same symmetry as the oscillogenic peaks.

Figure 6.34 The first derivative of the Nernst function with respect to potential.
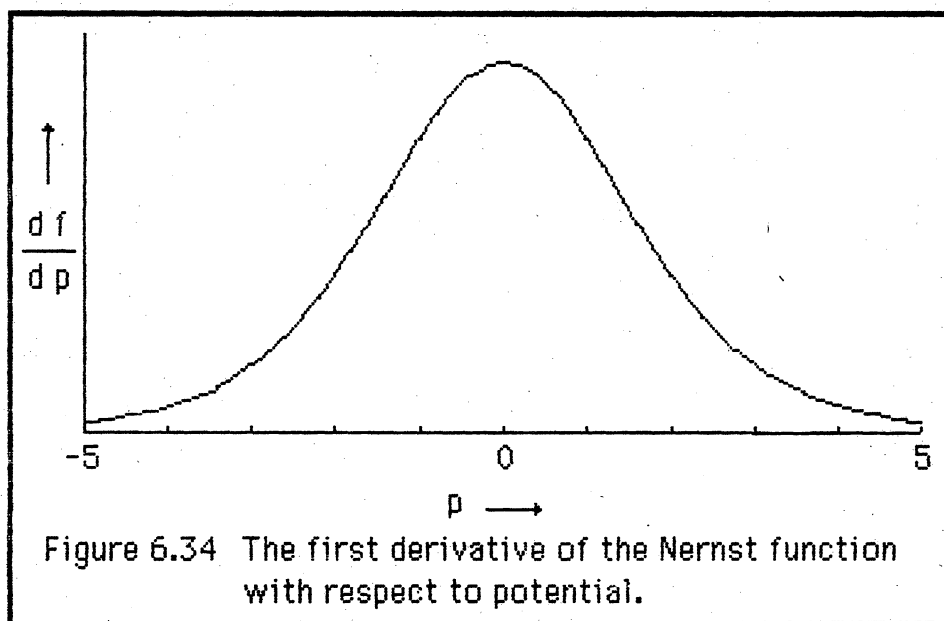
The peak width at half height can be calculated for the function of equation 6.3:

$$P_{1/2} = 3.526 \text{ (dimensionless potential units)} \tag{6.4}$$

$$E_{1/2} = 0.091 \text{ (volts)} \hspace{2cm} [6.5]$$

The measured value in Figure 6.32 is 0.11 volts. The peak width at half height for scanning voltammetry varies with the scan rate, and is not symmetrical (in extreme cases there is <u>no</u> half height on the negative side of the peak). But a comparison of Figures 6.32 and 6.14 shows that the oscillogenic technique has a sharper peak than scanning voltammetry. This may provide the ability to more easily discriminate between chemical species.

A combined apparatus for conducting both scanning voltammetry and scanning oscillogenic tests can be constructed. Figure 6.35 shows the connections needed for this.

Figure 6.35 Combined experimental setup
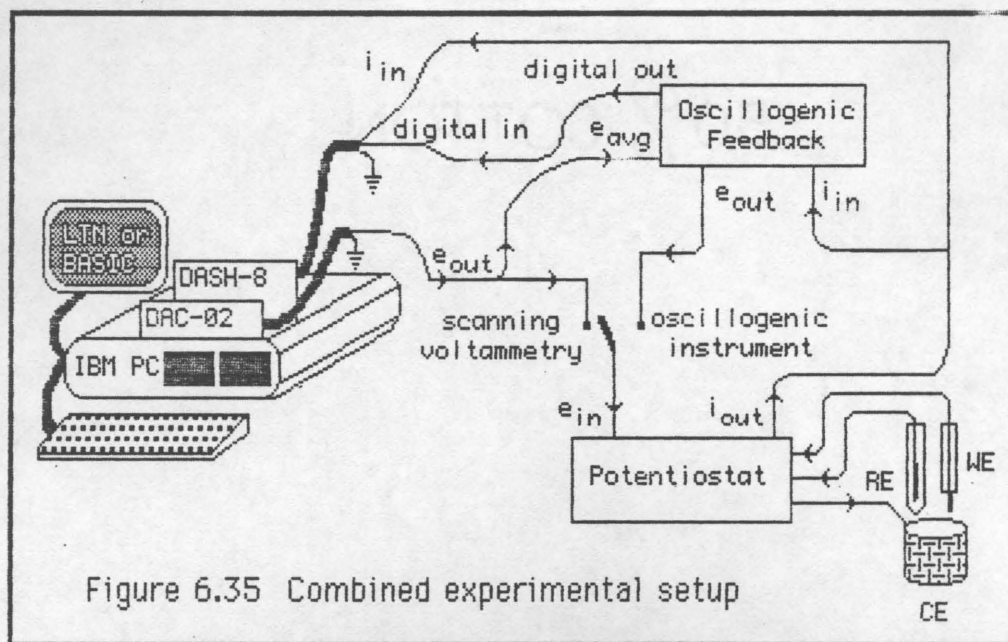
Note that a single switch, in Figure 6.35, converts the experiment from scanning voltammetry to scanning oscillogenic. Of course, the software in the microcomputer that controls the experiment must be changed as well, but this is just as easy as flipping a switch.

This work has shown that an oscillogenic system is able to measure concentration in a system with one component. How selective is the oscillogenic method compared to scanning voltammetry? In a system of two or more components, what effect does changing the concentration of component A have on the measurement of the concentration of component B? A different electrochemical system was chosen to investigate this question. The second system was also used to illustrate the technique with a different type of electrode, in a different potential region, using different types of ions, and a different type of electrochemical reaction (an insoluble product rather than a soluble one). The system chosen was a mixture of Lead and Cadmium ions in a perchlorate electrolyte [Brown 1984].

Stock solutions of aqueous NaOH (ACS certified grade, Fisher Scientific Co., Springfield, NJ 07081), and perchloric acid (from ACS reagent grade, 70% solution, Fisher Scientific Co., Springfield, NJ 07081) were prepared. The NaOH was standarized against potassium hydrogen phthalate. Stock solutions of $Pb(NO_3)_2$ (ACS certified grade crystals, Fisher Scientific Co., Springfield, NJ 07081) and $Cd(NO_3)_2 \cdot 4H_2O$ (ACS certified grade crystals, Fisher Scientific Co., Springfield, NJ 07081) in distilled water were also prepared. A burette was used for the NaOH and perchloric acid solutions to dispense appropriate quantities into 100 ml

beakers. Pipets were used to dispense the Lead and Cadmium solutions. The pH of the mixture was adjusted to 1.7 by the addition of perchloric acid stock. The mixtures were transferred to 100 ml volumetric flasks and diluted to the mark with distilled water. The final solutions were nominally: 0.5 M $NaClO_4$, 0.01 M $HClO_4$, and 0.004 M Cd^{++}. Concentrations of Lead ion were: 0.002, 0.004, 0.006, 0.008, and 0.010 M Pb^{++}. The experiments were done in triplicate. The peak position should have remained constant, and if a run deviated significantly (>0.012 volt) from the average peak potential it was rejected. No more than one run out of the three was ever rejected.

Caution should be exercised in the use of the perchlorate ion because some organic materials when mixed with perchlorates form substances which will detonate [Varyu (1985), Eisenbaumer (1985)]. In particular, cellulosic materials should be avoided. The perchlorate materials should be kept in dilute solutions and should not be allowed to dry out.

The averages of the replicate scanning voltammetry results for the concentration series are shown in Figures 6.36 to 6.40. The curves in Figures 6.36 to 6.40 are made up of short line segments; this is due to a limitation of the computer hardware/software combination used to collect the data. If more data points were collected during a run, then the curves would be smoother. There are two peaks corresponding to the two electroactive ionic species. The peak resulting from the reaction of Lead is near -0.4 volts, and the peak resulting from the reaction of the Cadmium is near -0.6 volts. The averages of the replicate oscillogenic

scans are shown in Figures 6.41 to 6.45. The curves in Figures 6.41 to 6.45 are composed of short line segments. This is due to the limitations of the computer interface hardware and the nature of the eletrode. The total scan times were limited by the time that a Mercury drop would safely stay on the tip of the hanging Mercury drop electrode. The Metrabyte analog-to-digital card had a frequency measuring mode and a period measuring mode, howver, the frequencies used (100's of Hz) fell in a region where the precision of the card was low. The measurements in Figures 6.41 to 6.45 were made in frequency mode, counting cycles for one second, thus the precision of the measurement was ±1 Hz. The peak values in both the scanning voltammetry and the oscillogenic scan should be proportional to concentration (equations 5.6 and 5.16). The peak values as a function of concentration are shown in Figures 6.46 to 6.49. The Lead concentration changes and the peaks reflect this change; Figure 6.47 for scanning voltammetry and Figure 6.49 for the oscillogenic scan show the dependence of the peak current and the peak frequency on the Lead concentration. Since the Cadmium concentration does not change, any change in the Cadmium peak heights with a change in Lead concentration indicates an interference of the Lead in the Cadmium determination. Figure 6.46 shows the Cadmium peaks in scanning voltammetry and Figure 6.48 shows the Cadmium peaks in the oscillogenic scan.

Both the scanning voltammetric and the scanning oscillogenic methods show similar responses as a function of concentration. Comparing the peak widths in scanning voltammetry (Figures 6.36 to 6.40) to those from the oscillogenic scan (Figures 6.41 to 6.45), the scanning
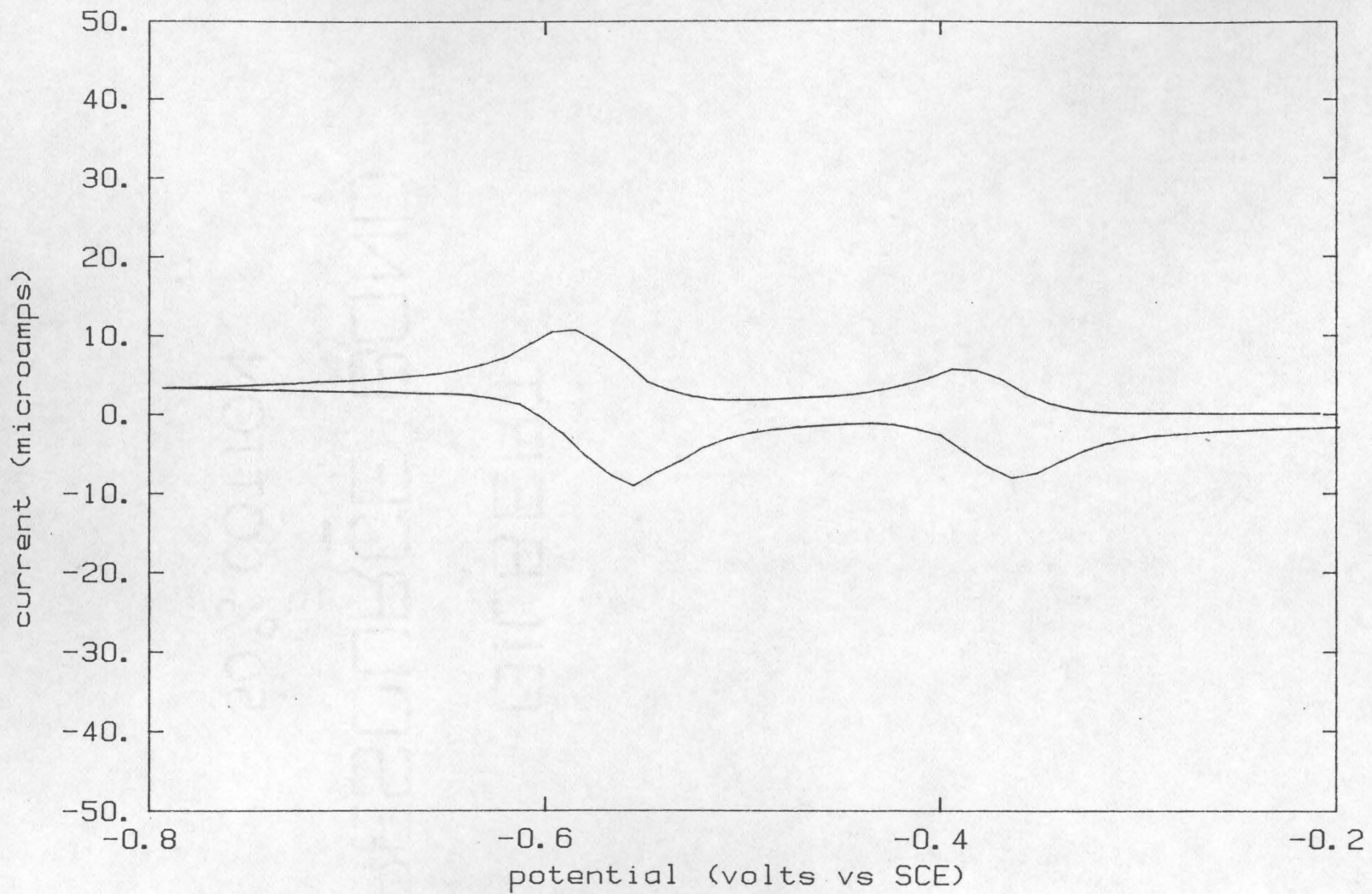
Figure 6.36 Scanning Voltammetry of Cd(II)/Pb(II)
(4 mM Cd, 2 mM Pb)
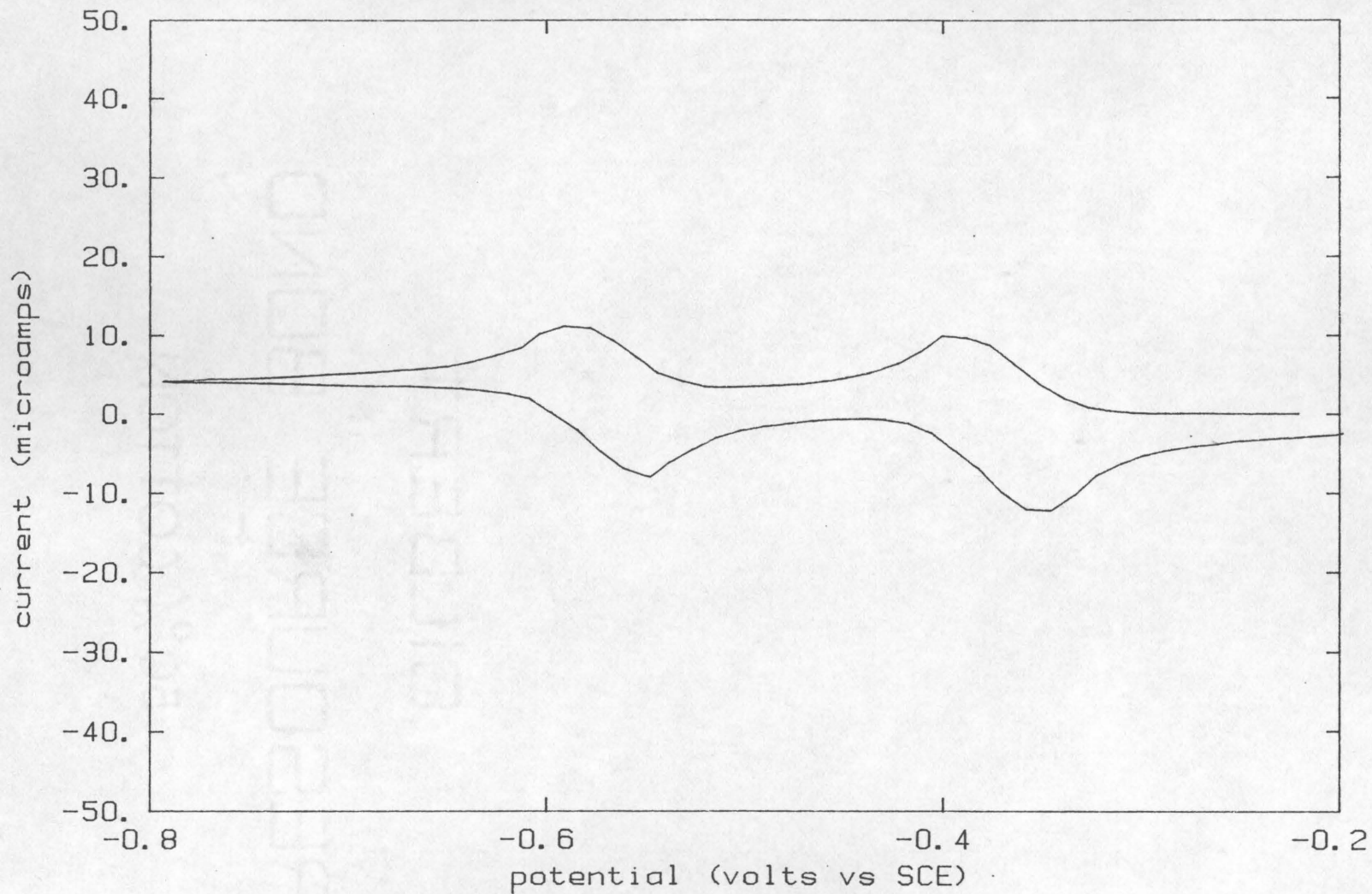
Figure 6.37 Scanning Voltammetry of Cd(II)/Pb(II)
(4 mM Cd, 4 mM Pb)
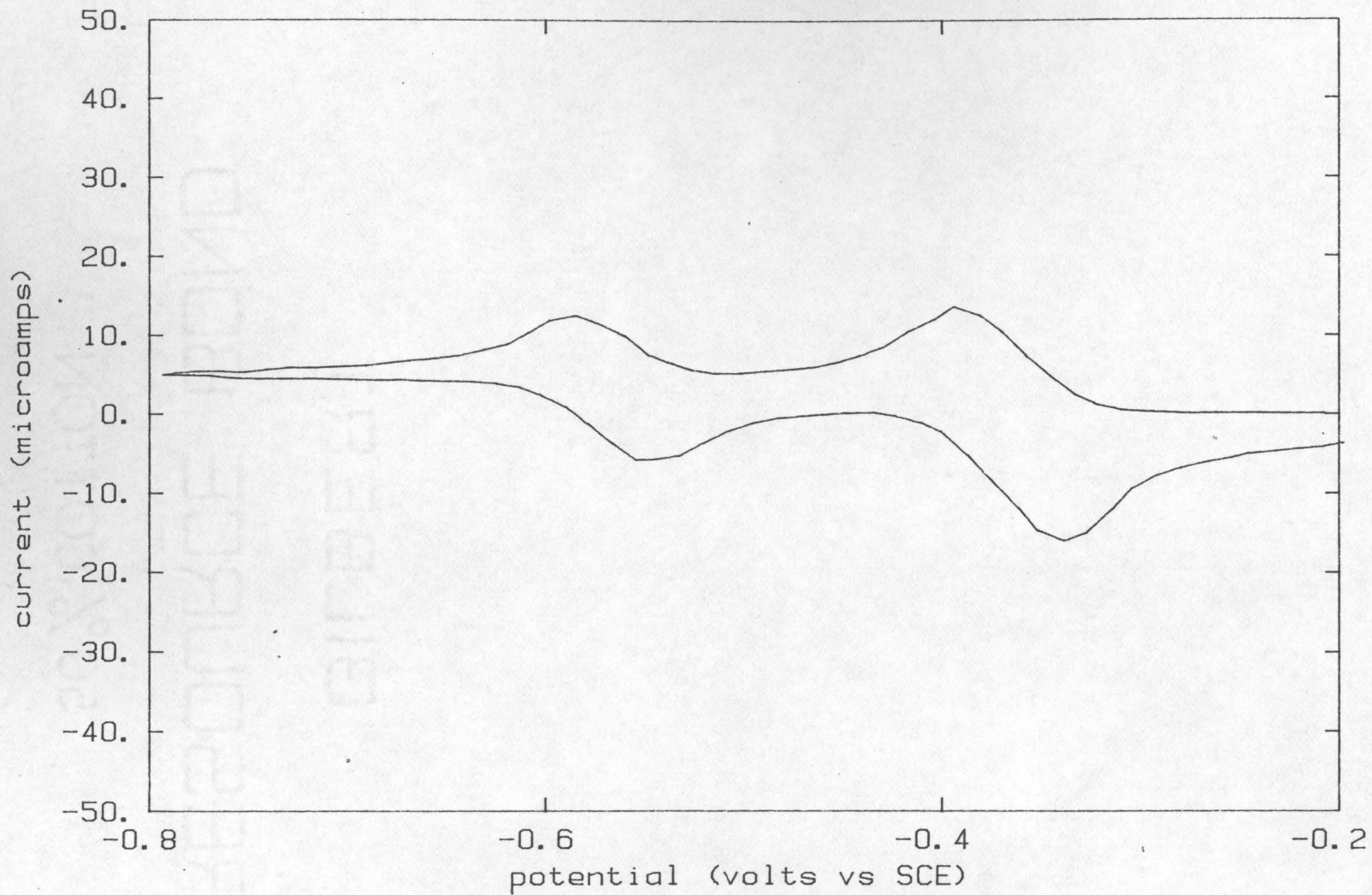
Figure 6.38   Scanning Voltammetry of Cd(II)/Pb(II)
(4 mM Cd, 6 mM Pb)
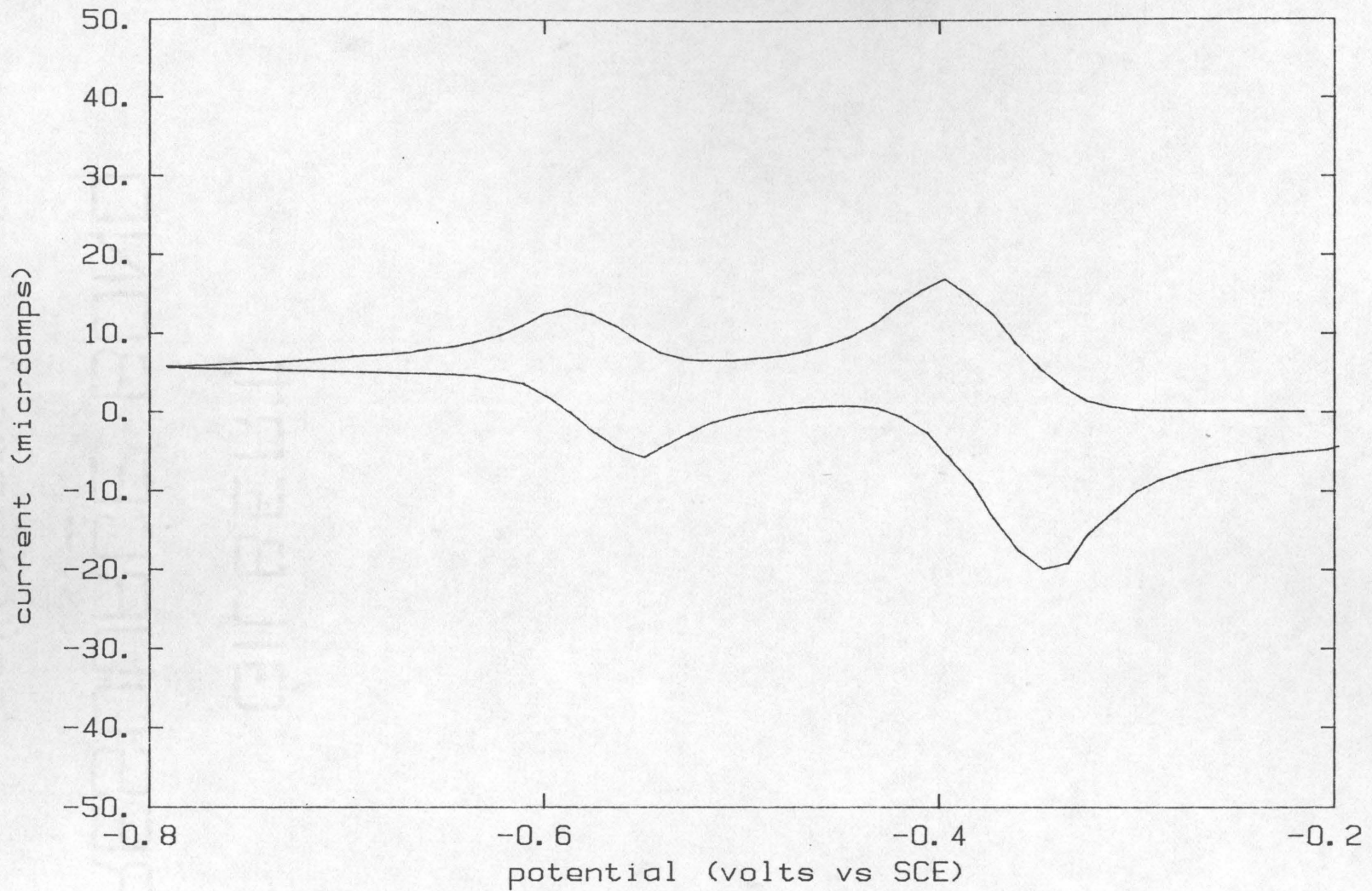
Figure 6.39   Scanning Voltammetry of Cd(II)/Pb(II)
(4 mM Cd, 8 mM Pb)
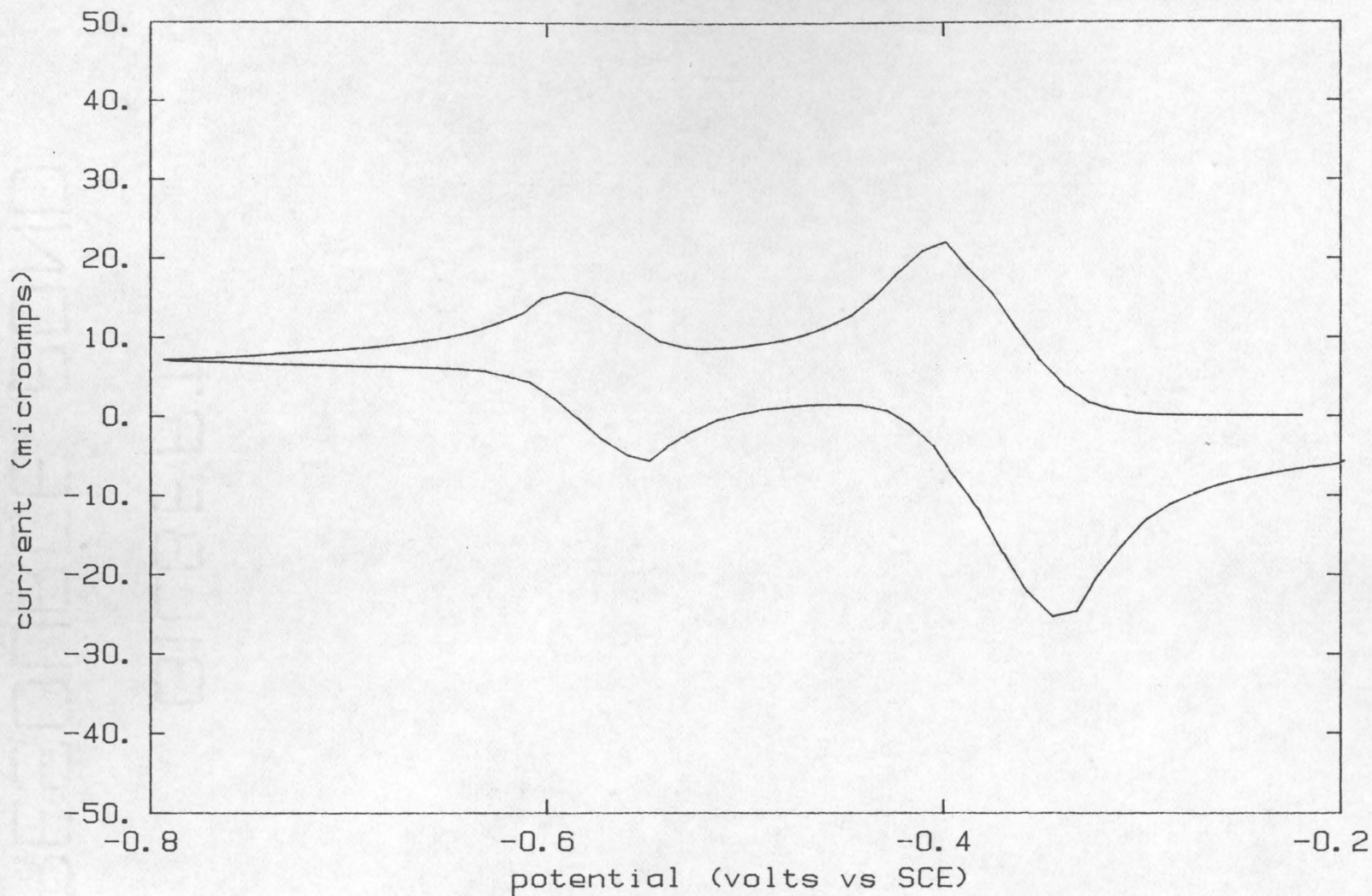
Figure 6.40 Scanning Voltammetry of Cd(II)/Pb(II)
(4 mM Cd, 10 mM Pb)
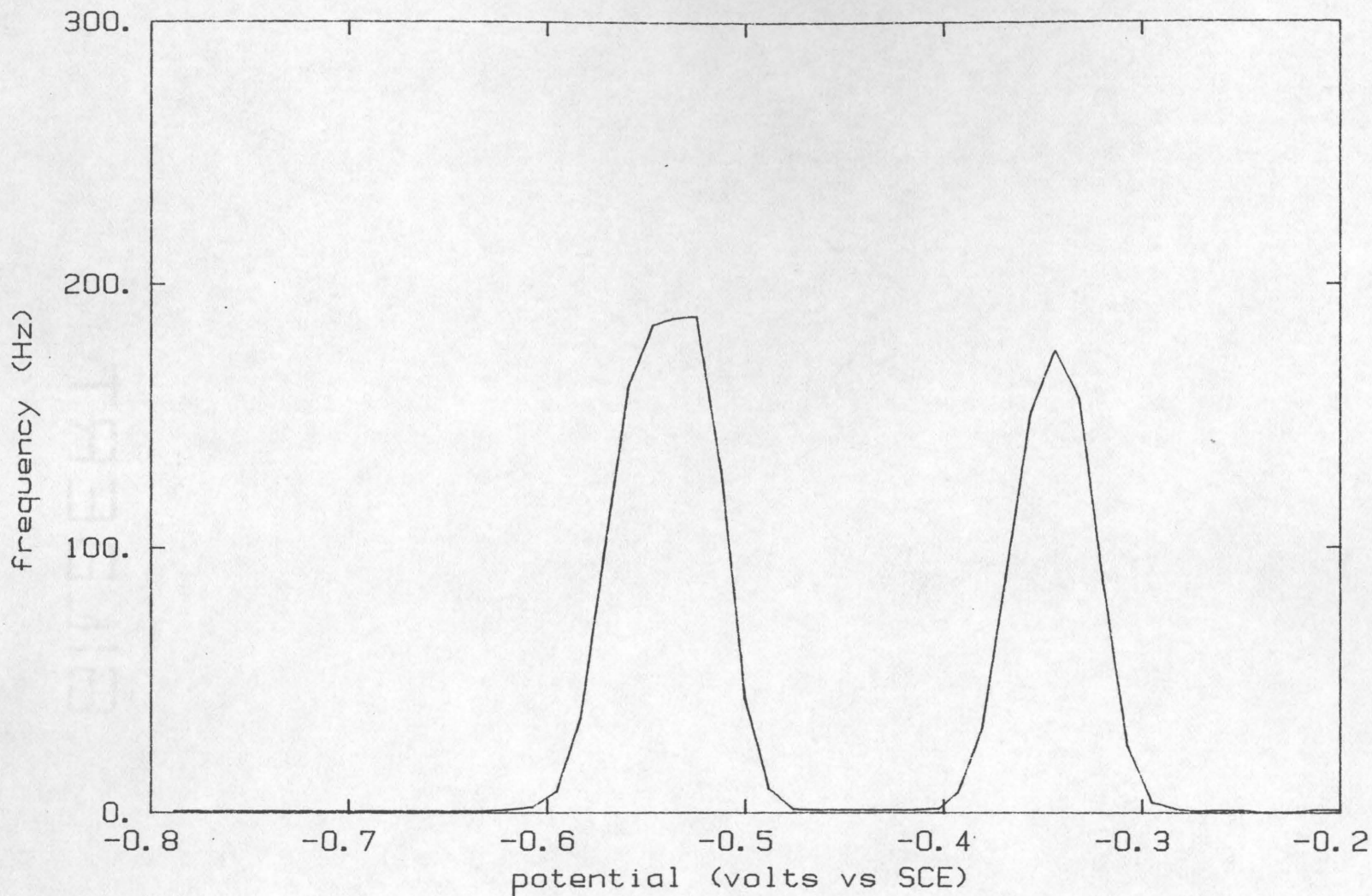
Figure 6.41   Oscillogenic Scan of Cd(II)/Pb(II)
(4 mM Cd, 2 mM Pb)
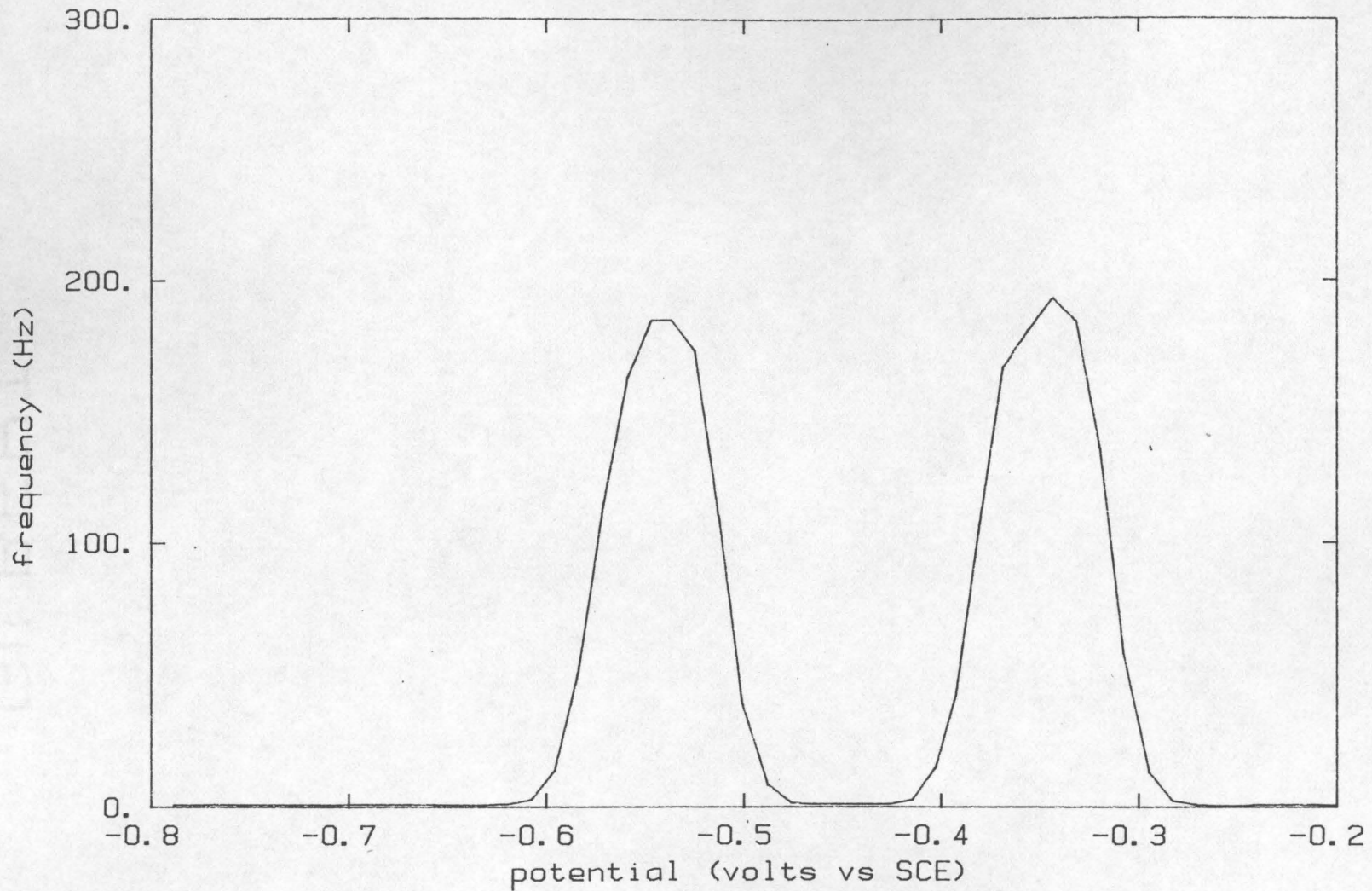
Figure 6.42 Oscillogenic Scan of Cd(II)/Pb(II)
(4 mM Cd, 4 mM Pb)
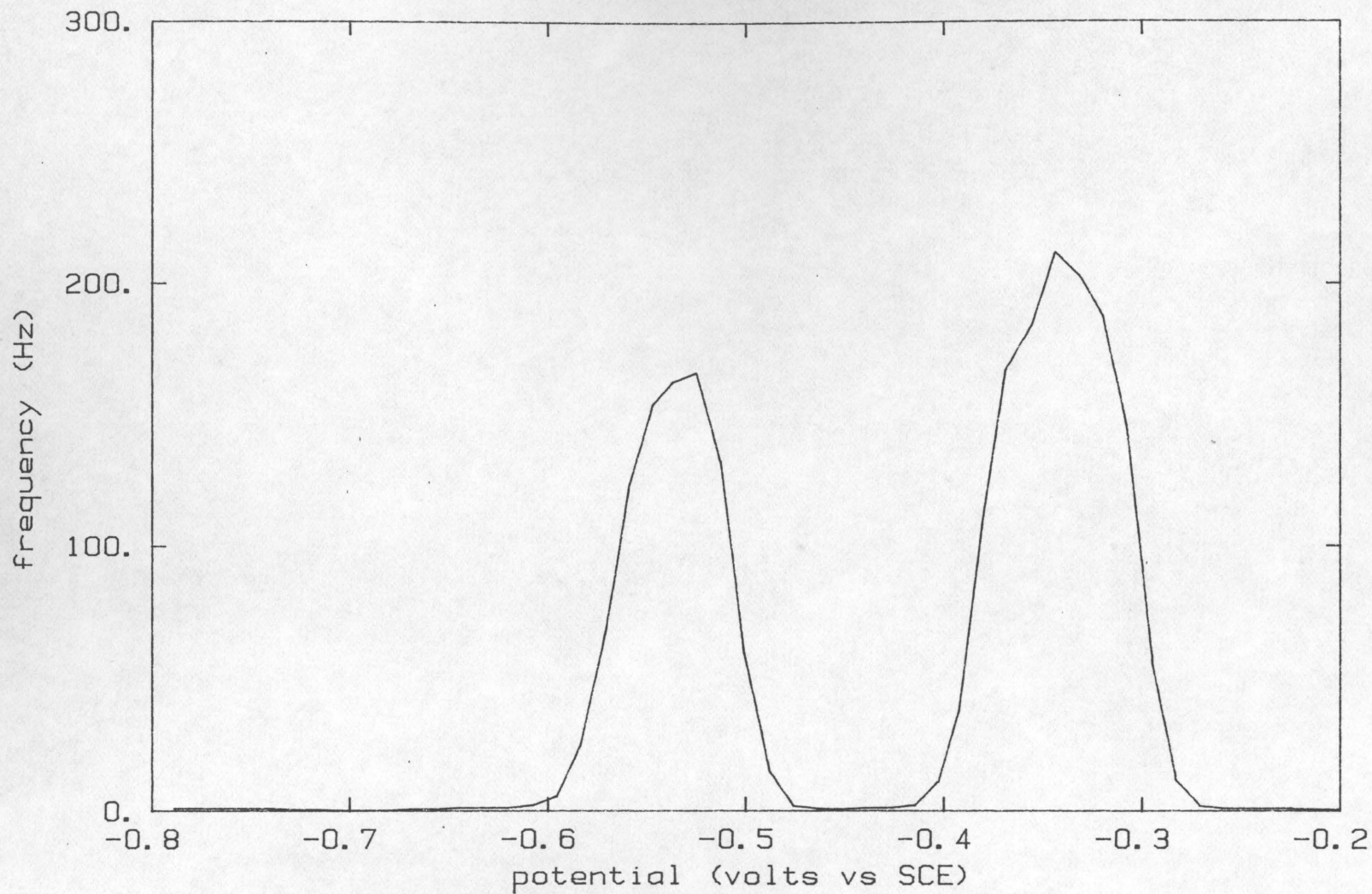
Figure 6.43   Oscillogenic Scan of Cd(II)/Pb(II)
(4 mM Cd, 6 mM Pb)
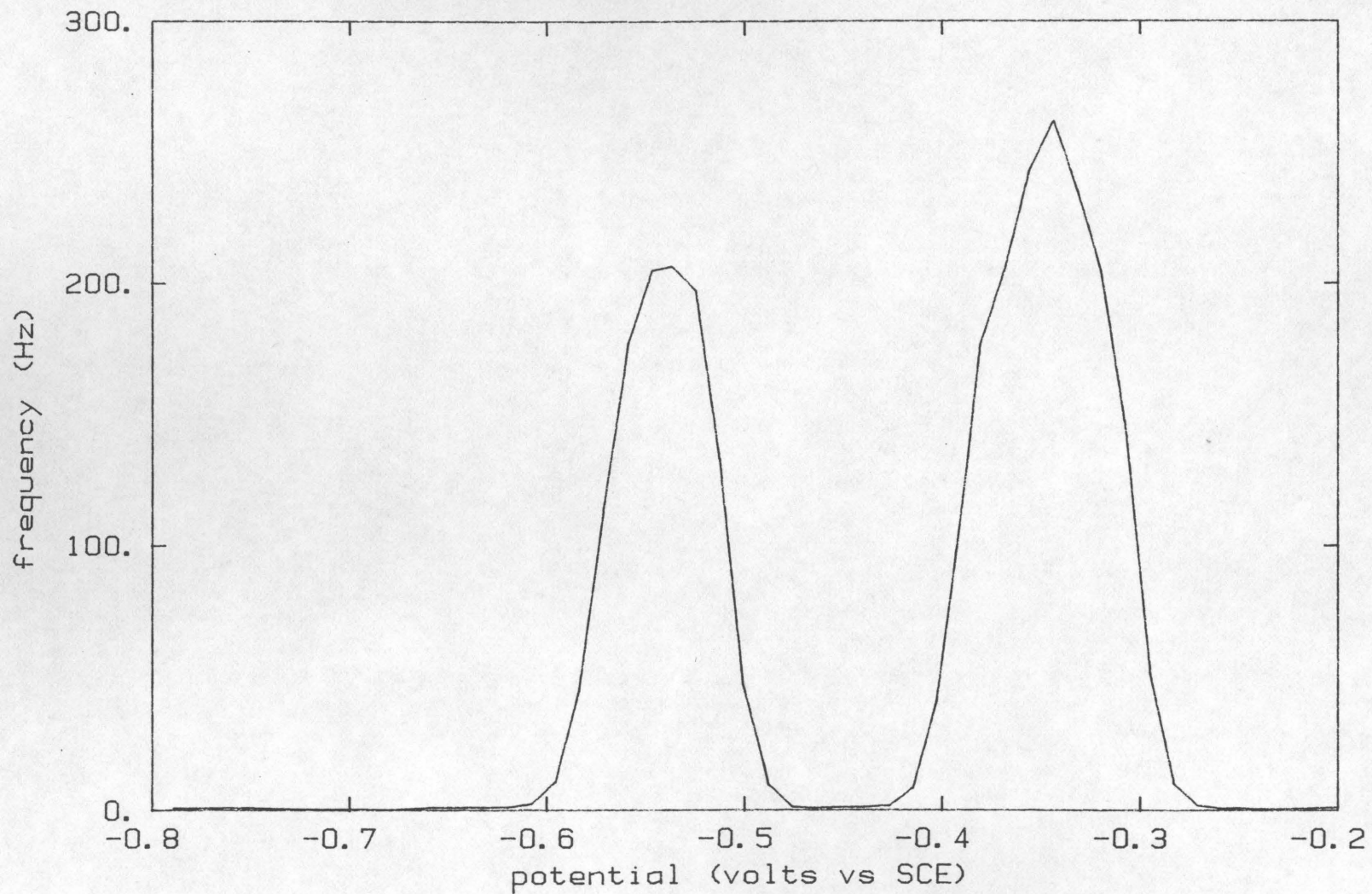
Figure 6.44   Oscillogenic Scan of Cd(II)/Pb(II)
(4 mM Cd, 8 mM Pb)
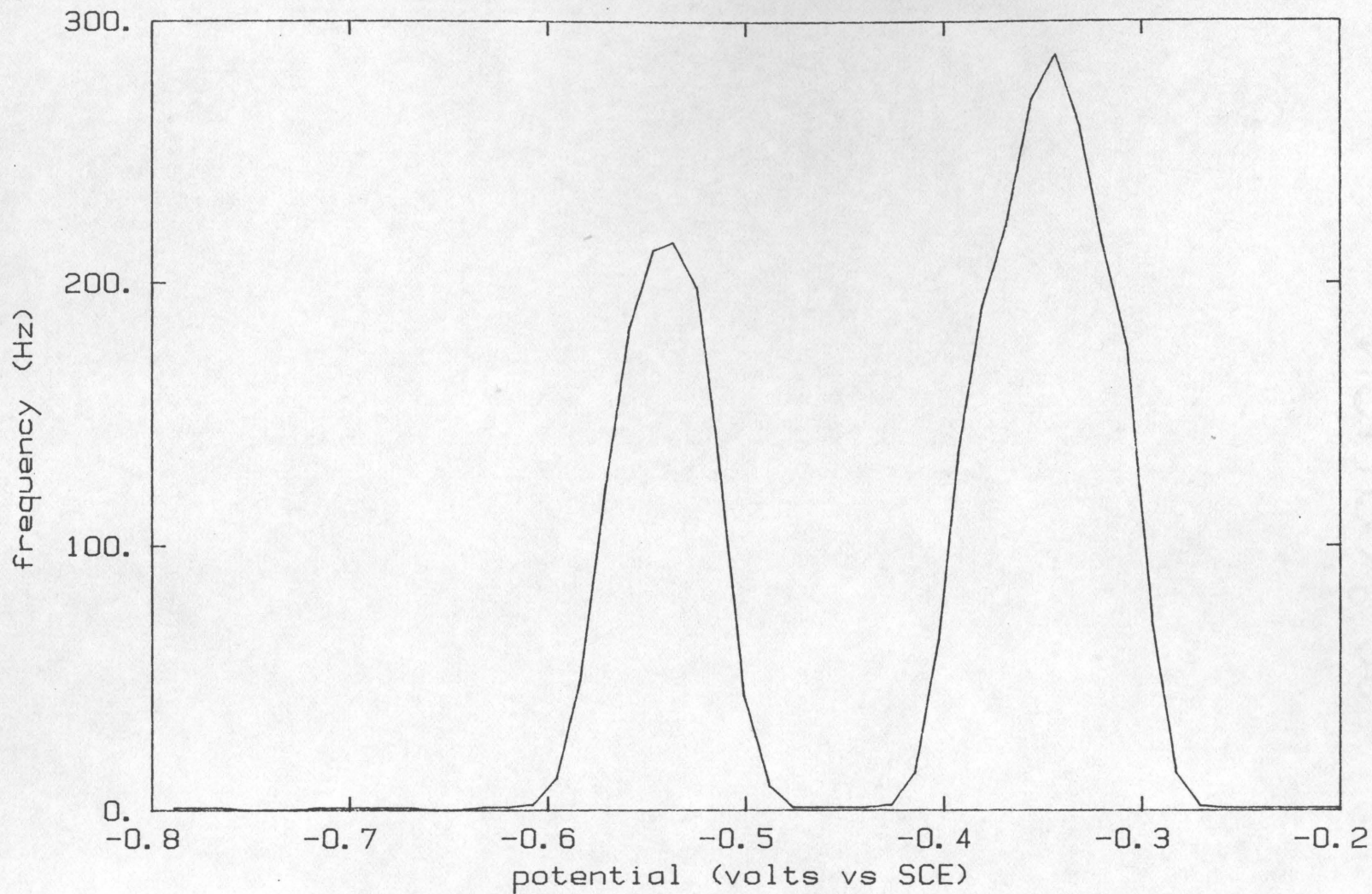
Figure 6.45   Oscillogenic Scan of Cd(II)/Pb(II)
(4 mM Cd, 10 mM Pb)
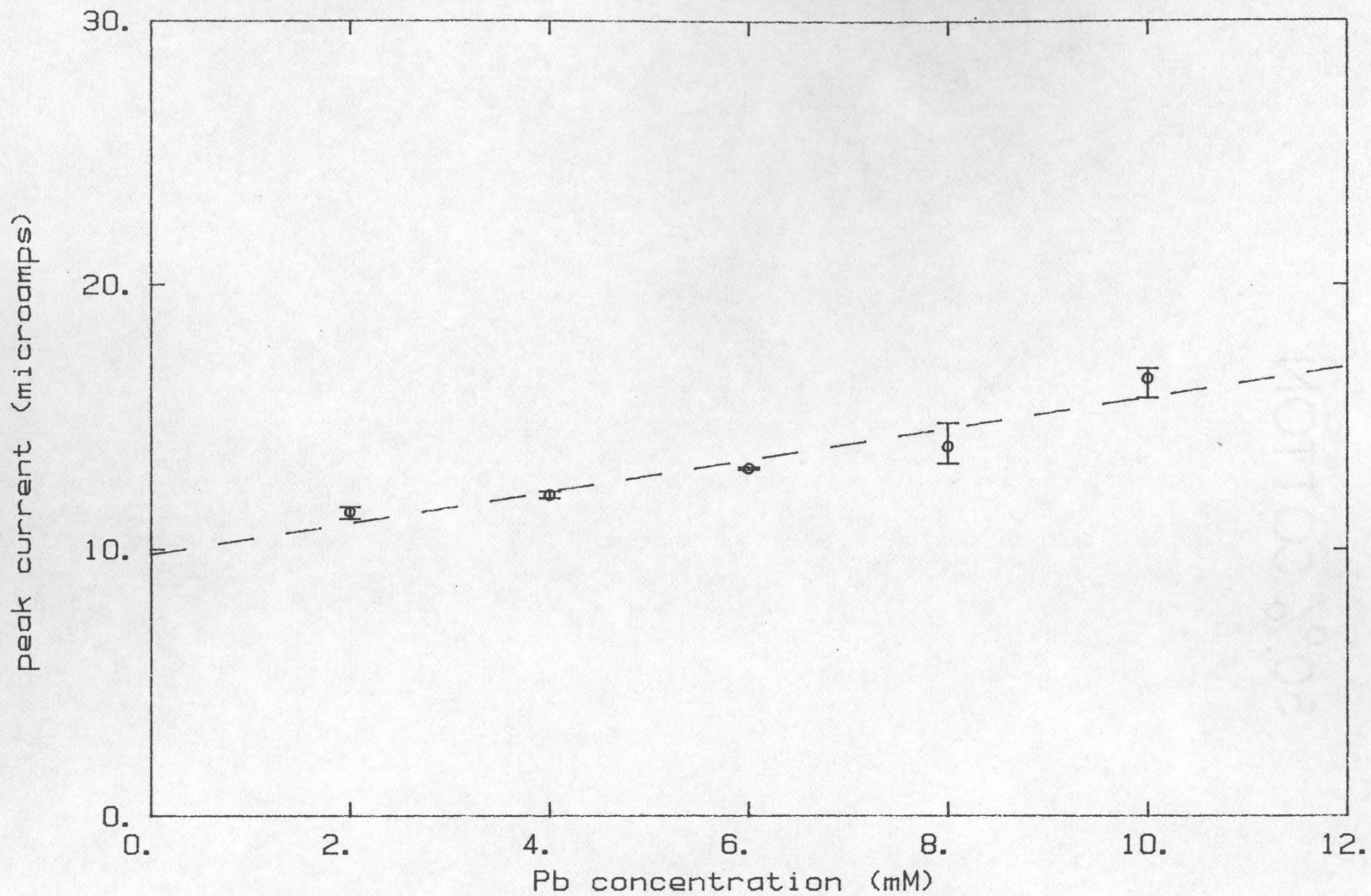
Figure 6.46 Cadmium(II) peak in scanning voltammetry
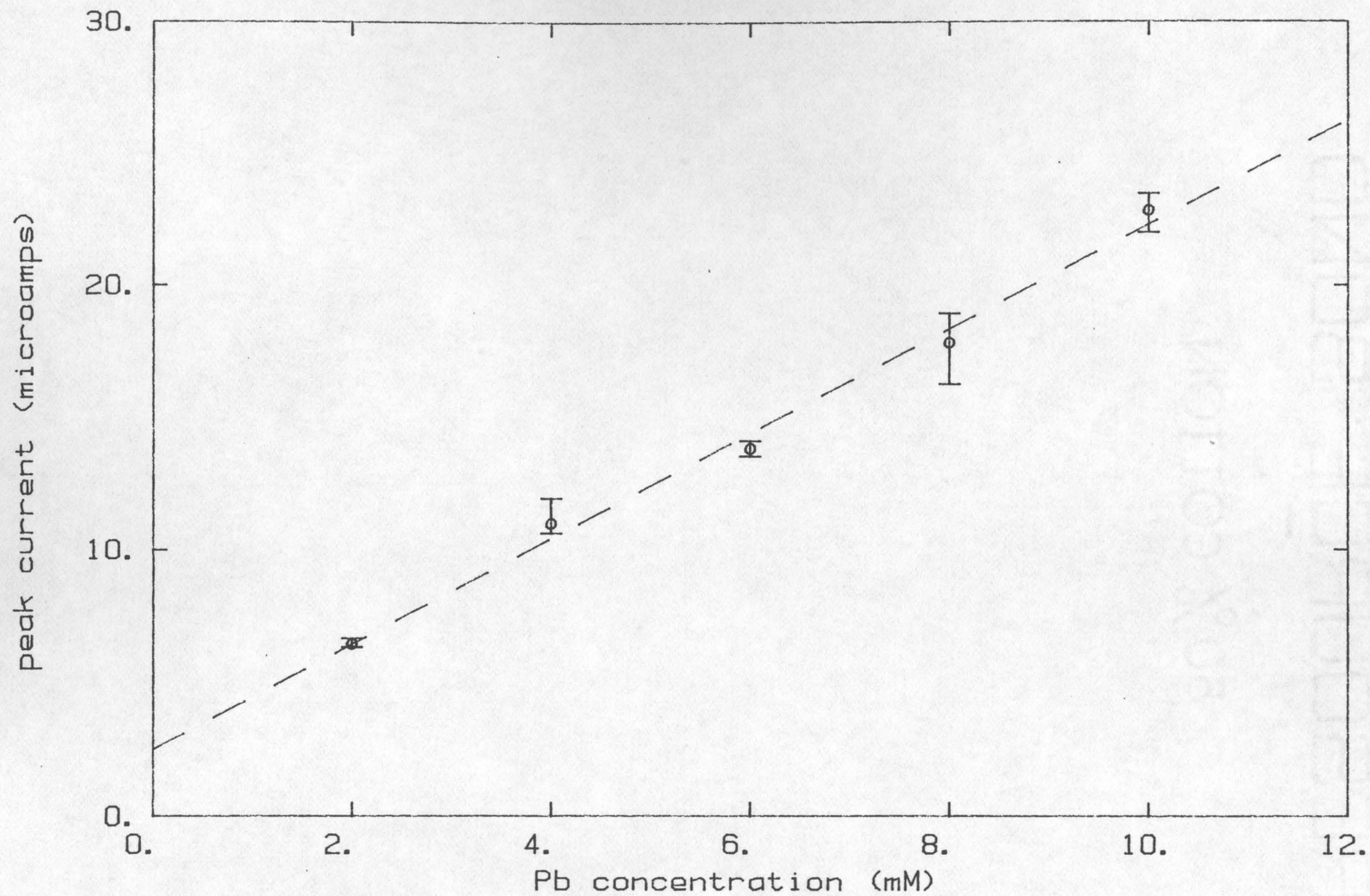
Figure 6.47 Lead(II) peak in scanning voltammetry
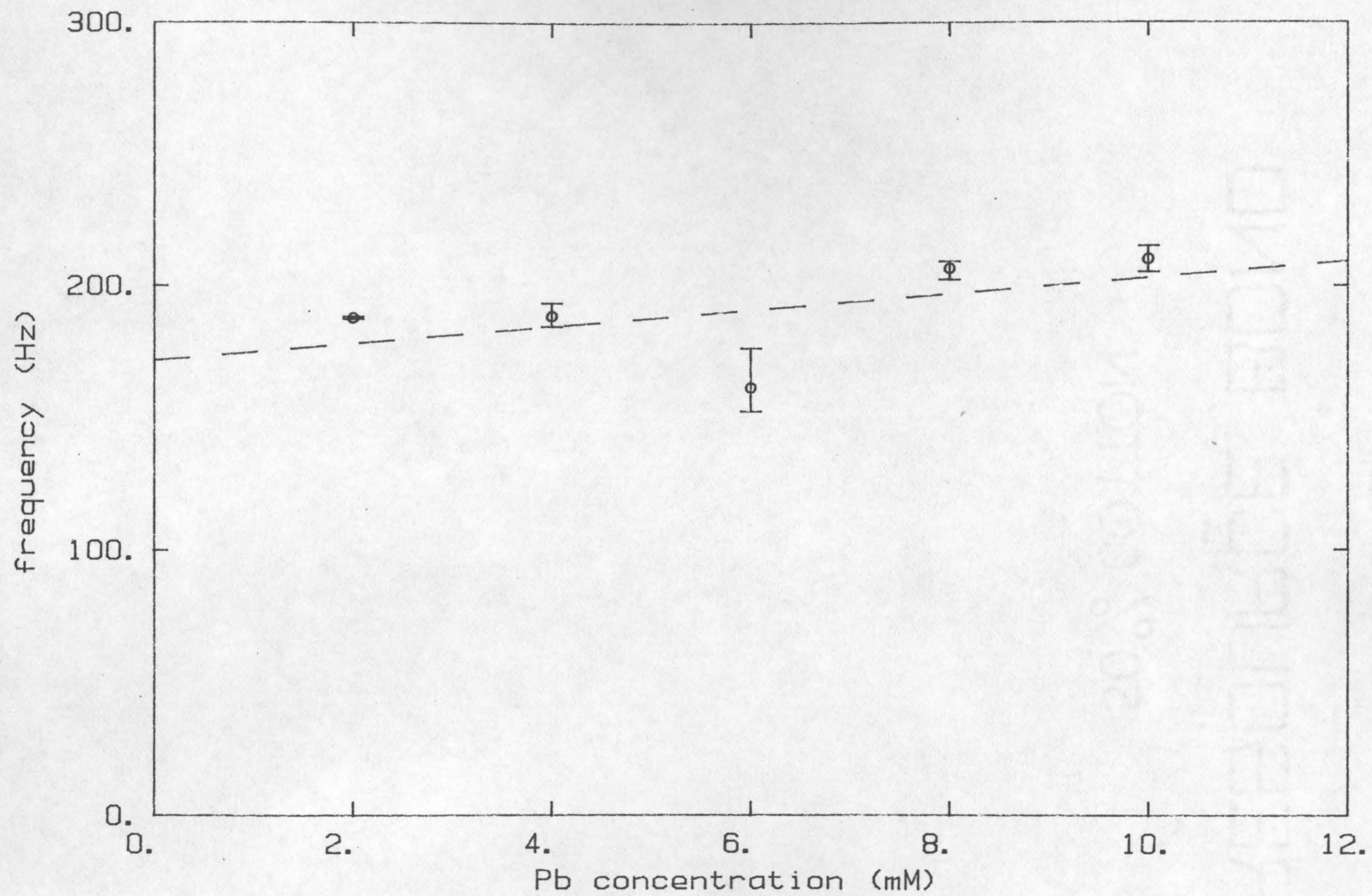
Figure 6.48 Cadmium(II) peak in oscillogenic scan
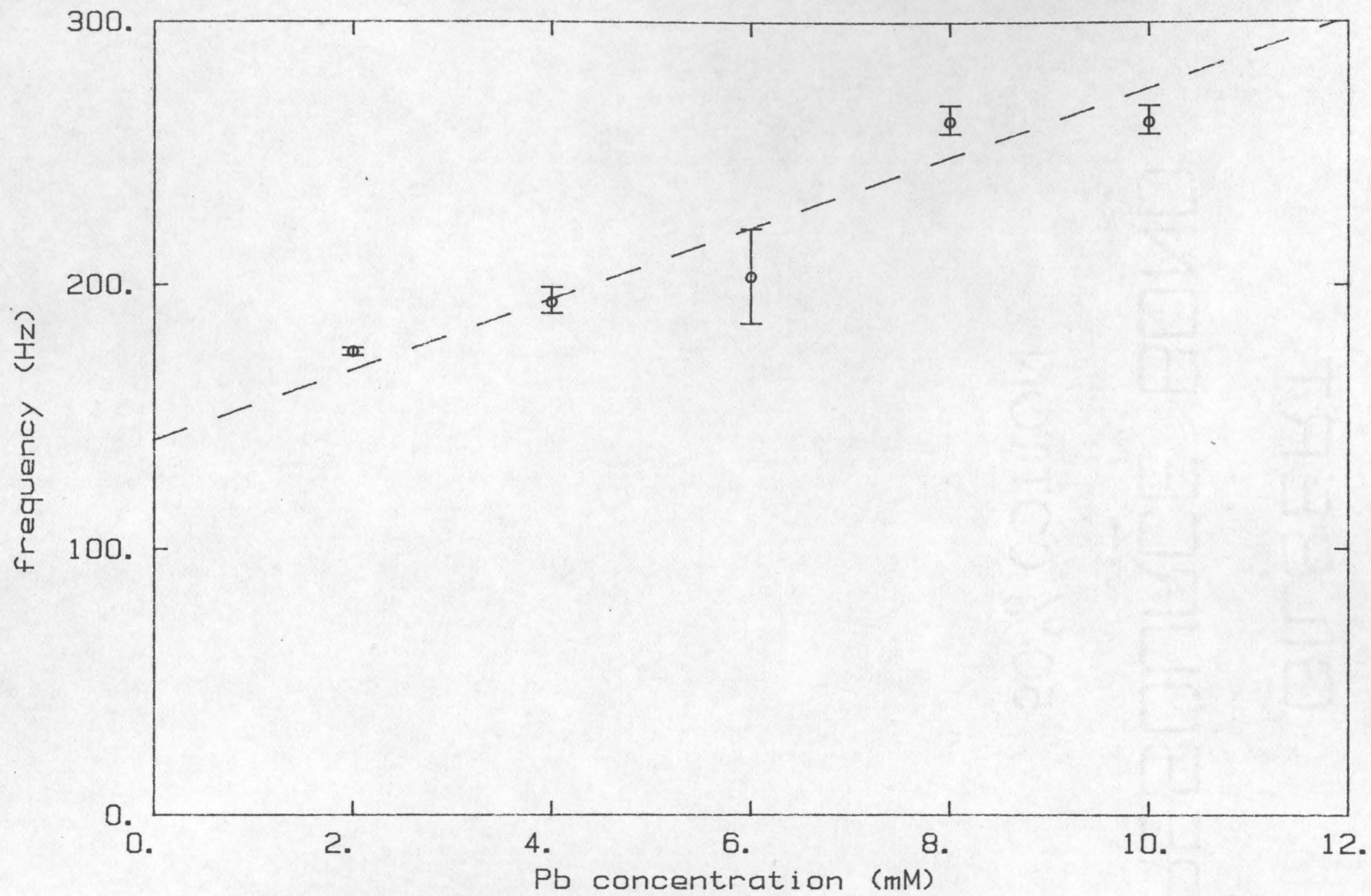
Figure 6.49 Lead(II) peak in oscillogenic scan

voltammetry current peaks appear to be somewhat broader than the frequency peaks from the oscillogenic scan. Because the oscillogenic peaks are marginally better resolved than the scanning voltammetric peaks one might expect the interaction between the Lead and Cadmium responses to be much less in the oscillogenic system. But there are interferences in both methods. While the Cadmium concentration did not change, the peak response in both methods did. In both methods, the change in the Lead peaks was much greater than that of the Cadmium peaks (as expected). This is a fairly rigorous test for both methods. The Lead ion reacts before the Cadmium ion in the potential scan and the Lead will continue to react when the Cadmium begins to react. So a maximum interference effect can be expected. In an effort to compare the relative magnitudes of the interference in both methods, the Cadmium responses for both scanning voltammetry and oscillogenic methods were normalized on the scale of the Lead responses for each method. The results are plotted in Figure 6.50. The minimum value on the vertical axis represents the response for the minimum (0 mM) concentration of Lead (as predicted by the linear regression for both methods); the maximum value on the vertical axis represents the response for the maximum (12 mM) concentration of Lead (as predicted by the linear regression for both methods).

For scanning voltammetry results:

normalized peak response (% of maximum) =

$$(i_{peak} \text{ (microamperes)} - 0.2488)/(2.6248-0.2488) \qquad [6.6]$$
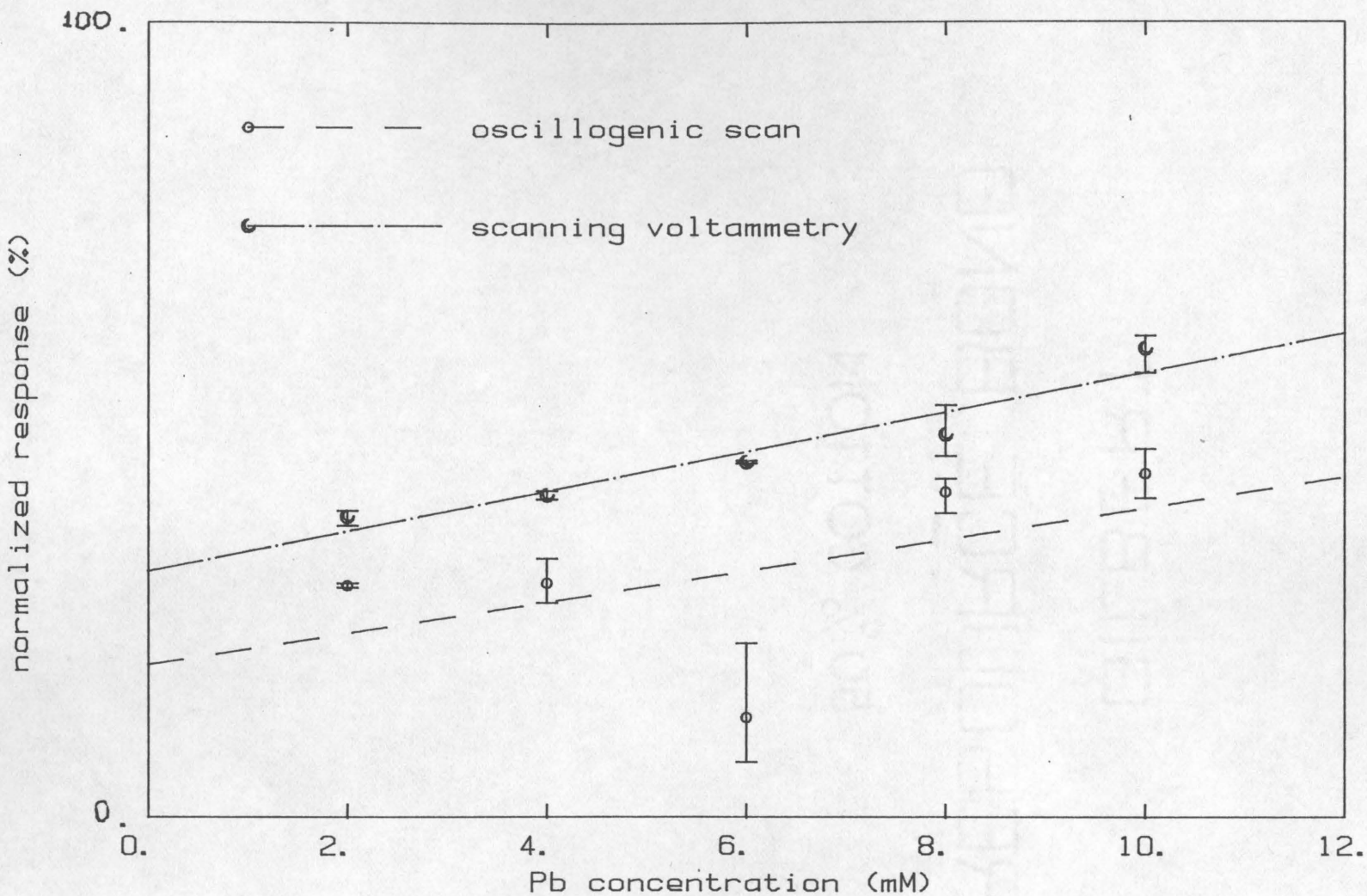
Figure 6.50 Cadmium peaks as a function of Lead concentration
(normalized based on minimum and maximum Lead response)

For oscillogenic scan results:

normalized peak response (% of maximum) =

$$(f_{peak} \text{ (Hz)} - 141.00)/(301.46-141.00) \qquad [6.7]$$

Figure 6.50 shows no significant difference between the curves for the scanning voltammetry and the oscillogenic method. Despite the fact that the peaks are better resolved in the oscillogenic method, this does not appear to reduce the interference observed. But it can be said that the oscillogenic technique will give concentration results just as selective as the scanning voltammetry method.

A video tape (VHS) of the experimental procedure was made by P. K. Mercure and P. R. Rony. Copies of this tape may assist in the duplication of these experiments. P. R. Rony repeated the experiments with $Fe(CN)_6^{-3}$ ion. He noted (24 July 1985) that the frequency versus potential curve scanning in the negative potential direction with the oscillogenic instrument <u>was essentially retraced on the positive potential scan</u> (Figure 6.51). This is in contrast with the scanning voltammetry results, where there is considerable difference between the negative and positive potential scans. One element of symmetry is due to the fact that the frequency is always a positive number (oscillogenic) while the current (scanning voltammetry) swings from positive to negative. With the Pb/Cd ions (Figure 6.52), the reverse potential scans usually had different peak frequencies than the forward scans, but the location of the potential peaks remained the same (Figure 6.52). There was some frequency instability at the peak on the positive going scan with the Pb^{+2}/Cd^{+2} system.
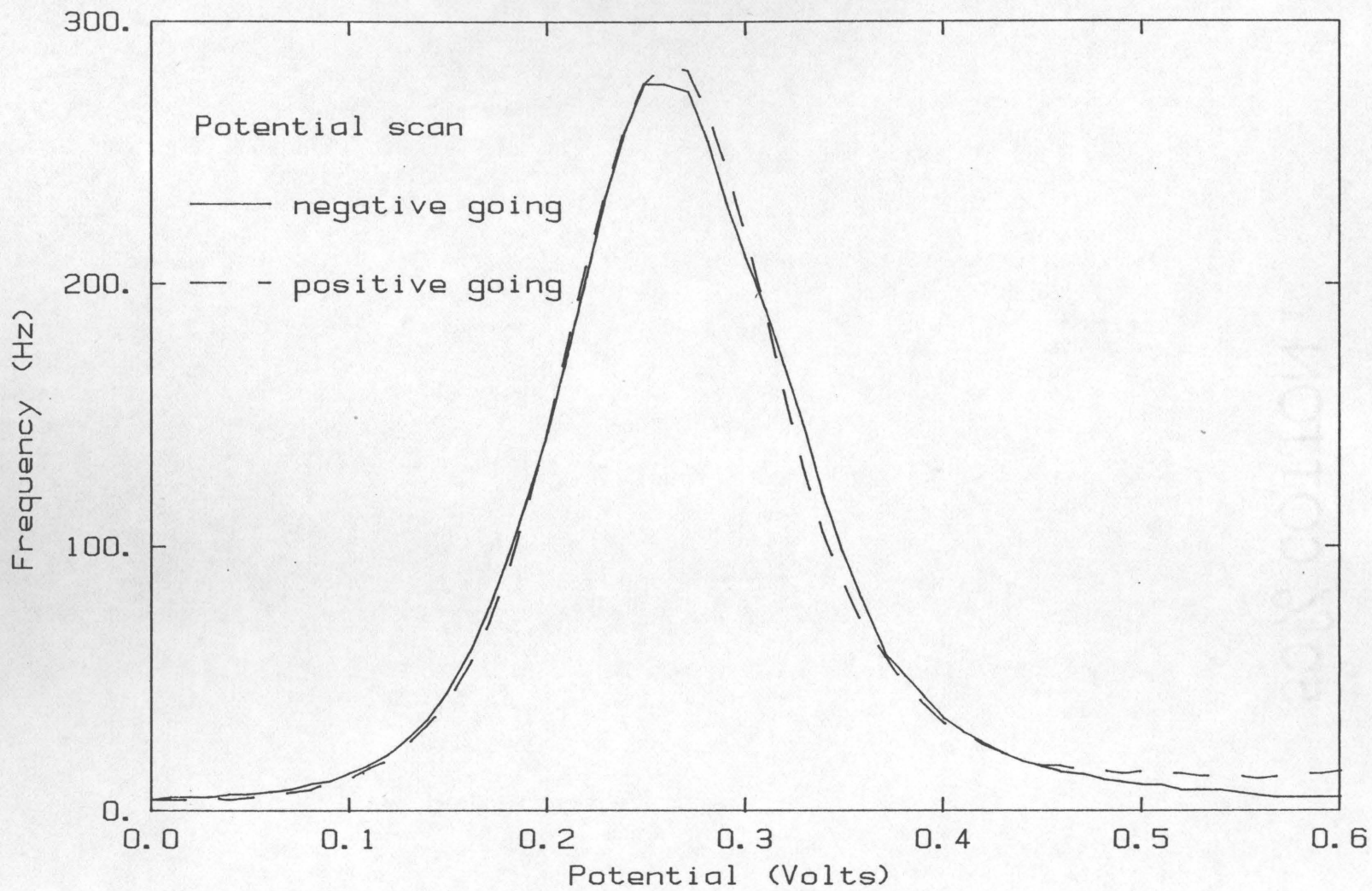
Figure 6.51 Forward and Reverse Oscillogenic Scan
of 10 mM Fe(CN)6 (data by P. R. Rony)
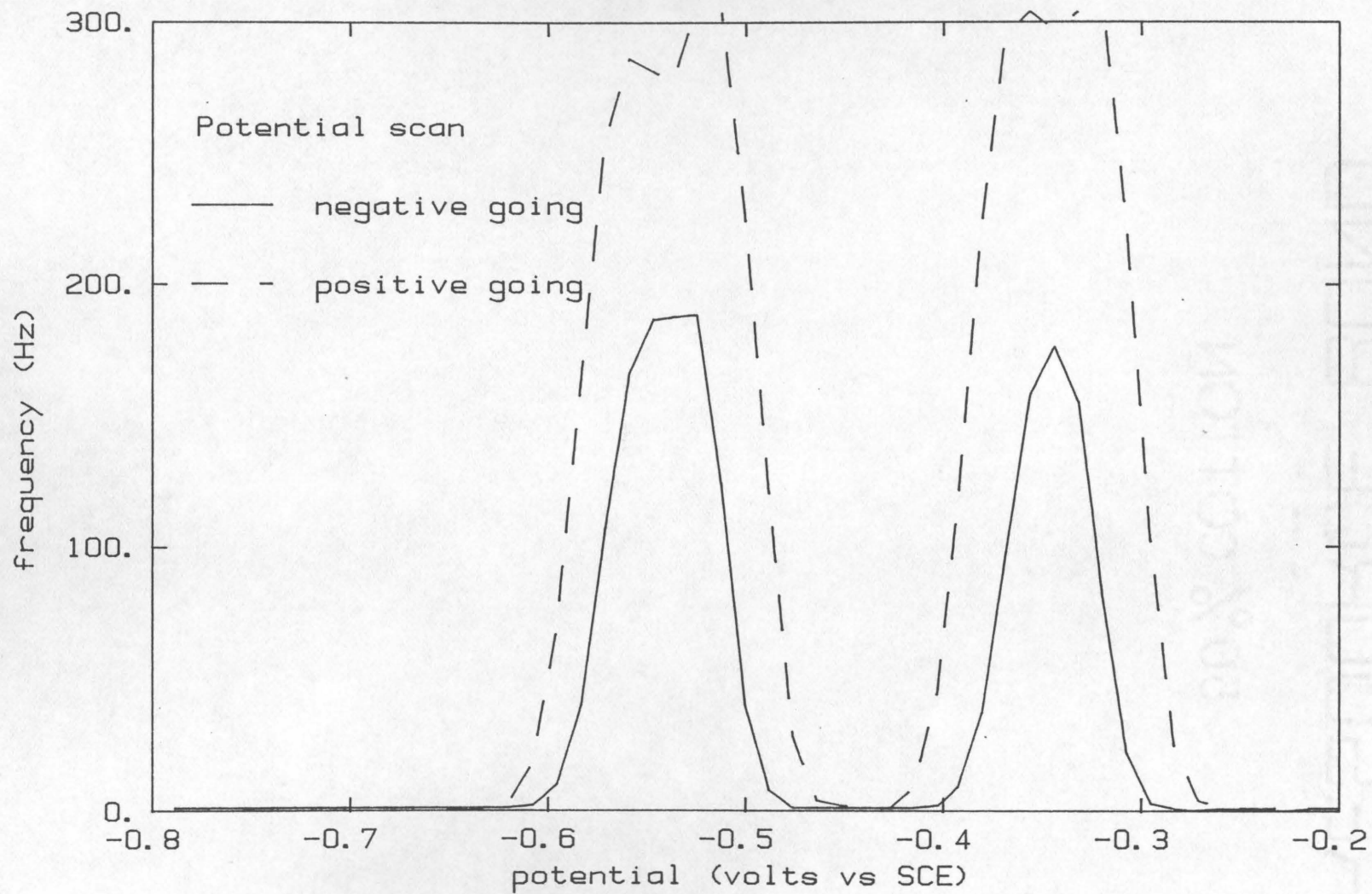
Figure 6.52 Oscillogenic scan, forward and reverse
(Pb 2 mM, Cd 4 mM)

One of the major differences between the Pb/Cd system and the $Fe(CN)_6^{-3}$ system is that the product of the Pb/Cd reaction is insoluble and plates out on the electrode, while the product of the $Fe(CN)_6^{-3}$ reaction is soluble and remains in solution. The derivative function describing the oscillogenic scan of the idealized rapid electrochemical reaction (reversible), equation 6.3, plotted in Figure 6.34 explains the similarity between the forward and reverse scans. If the reaction is fast, then equation 6.2 predicts the shape of the frequency versus potential graph. The direction of scanning does not affect the derivative of equation 6.1 with respect to potential. The oscillogenic experiment has literally hundreds of cycles at each potential value; the concentration profiles have time to reach a periodic state. The rapid switching of potential is superimposed on the much slower scan of the average potential value; thus a stable periodic state is reached at each potential. In scanning voltammetry, on the other hand, the concentration profiles do not reach any steady state, and the current/potential curve is always a transient curve. It is this transient concentration profile which causes the shifting peaks in scanning voltammetry between the positive and negative potential scans.

Since this chapter has been concerned with experimental tools to a great extent, this is the appropriate place to mention one tool used extensively in this work. All of the full page graphs were made on an HP-7470A plotter (Hewlett-Packard, 1820 Embarcadero Road, Palo Alto, CA 94303), driven by a BASIC program written by the author. This program is in Listing 26 of the Appendix.

In summary, this chapter has shown that the circuit suggested by the mathematical model of chapters 4 and 5 was a successful start to the design of an electrochemical oscillogenic instrument. The practical and non-ideal electronic devices and chemical systems used in the experiments required some modifications of the circuit to generate a stable oscillator. It was shown that the major source of instability in the final oscillogenic instrument was changes in the electrode surface. A novel "trapped mercury drop electrode" was proposed, but this work found it possible to get reasonable results with common electrodes. Scanning the average potential of the oscillogenic system through a range of potentials led to peaks in the frequency of the oscillogenic instrument which corresponded to different electrochemical species. Two different electrodes were used successfully. Electrochemical reaction with soluble and insoluble products were tested successfully. In cases of both single and multiple components, the peak frequencies were found to be directly related to the concentration of the electroactive species, and the results compared favorably with the standard electrochemical technique of scanning voltammetry. In multiple component mixtures, the peaks produced by the oscillogenic instrument were more separated than the peaks in scanning voltammetry. However, this work could show no significant improvement in selectivity by the oscillogenic technique over the scanning voltammetric technique.

The principle of an oscillogenic instrument has been demonstrated with real systems. This instrument can measure concentrations, and the advantages (simplicity and accuracy, to name two) of using a frequency

measurement would accrue to this device. The next chapter will serve to introduce a methodology that allows a computer to extract further information from the waveform of this oscillogenic instrument. Because frequency is only one measurement, the information content of a frequency measurement is less than the information content of a whole series of measurements of the waveform amplitude. The use of this additional information would increase the usefulness of the instrument.

Chapter 7. Parameter Estimation or How to Get More Than One Experimental Parameter from the Oscillogenic System.

"For we see now as through a glass darkly, but then we will see face to face"
1 Corinthians 13:12

Often instead of making a series of measurements, and then analyzing them as a set, one would like to have a more or less continuous measurement of some quantity. For example, if the species of interest absorbs light, one might use a spectrophotometer, and the absorbance at any time would be a measure of the concentration of the species. A meter is commonly used to display the absorbance and in this case one might think of the assemblage as a concentration meter. But experimental measurements are often only indirectly related to the quantity of interest, for example, a rate constant or diffusion coefficient. How, then does one make a 'meter' for such a quantity using the experimental measurements?

One technique, known as the Guggenheim method [Guggenheim (1926)], is still in use today [Bacon and Demas (1983), Hobey et al (1986)]. It is applicable to first order systems in one variable. Suppose we wish to measure the rate constant of a first order system:

$$\frac{dT}{dt} = -k\,(T - T_{ambient}) \qquad\qquad [7.1]$$

This is a system with a simple heat loss but the technique also applies to any other first order system such as reaction kinetics, nuclear decay, diffusion experiments, etc. The temperature, T, is measured at fixed intervals and from these measurements one wishes to evaluate the rate

constant, k, on a repetitive basis. The solution of the rate equation is:

$$(T - T_{ambient}) = (T_0 - T_{ambient})\, e^{-kt} \qquad\qquad [7.2]$$

where T_0 is the initial temperature

Figure 7.1 shows typical results for a first order system such as this; the data is synthetic to closely control the amount of noise in the measurement ($\sigma = 0.001\ °C$). At a given time, t, equation 7.2 becomes:

$$(T_t - T_{ambient}) = (T_0 - T_{ambient})\, e^{-kt} \qquad\qquad [7.3]$$

At a later time, $t+\Delta t$:

$$(T_{t+\Delta t} - T_{ambient}) = (T_0 - T_{ambient})\, e^{-k(t+\Delta t)} \qquad\qquad [7.4]$$

Taking the difference between equations 7.3 and 7.4:

$$(T_t - T_{t+\Delta t}) = (T_0 - T_{ambient})\, [\, e^{-kt} - e^{-k(t+\Delta t)} \,] \qquad\qquad [7.5]$$

or,

$$(T_t - T_{t+\Delta t}) = (T_0 - T_{ambient})\, e^{-kt}\, [1 - e^{-k(\Delta t)}] \qquad\qquad [7.6]$$

Taking the logarithm of equation 7.5:

$$\ln[T_t - T_{t+\Delta t}] = \ln[T_0 - T_{ambient}] - kt + \ln[1 - e^{-k(\Delta t)}] \qquad\qquad [7.6]$$
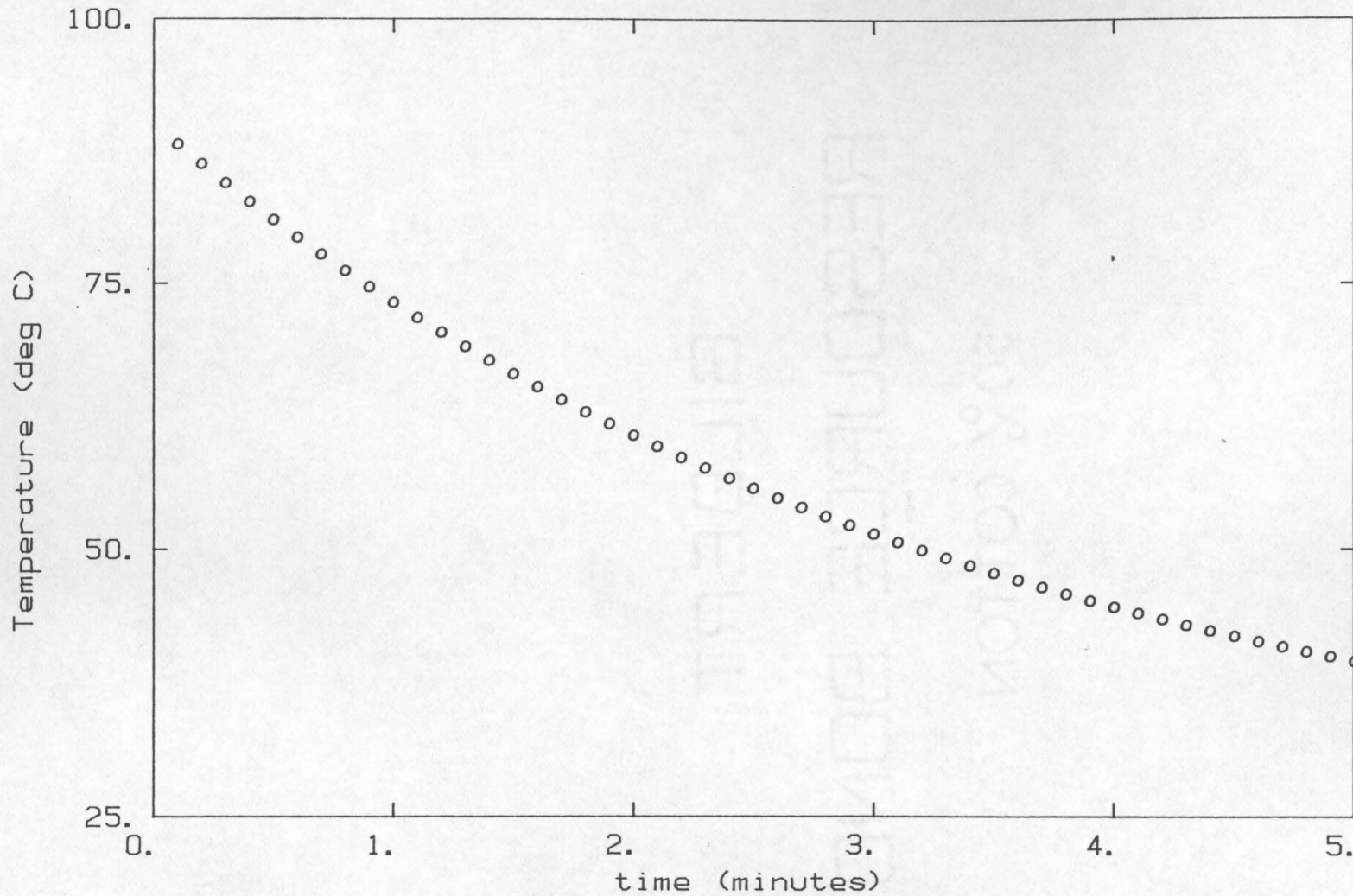
Figure 7.1   Temperature history of a cooling cup of coffee.

At the next sample time:

$$\ln[T_{t+\Delta t} - T_{t+2\Delta t}] = \ln[T_0 - T_{ambient}] - k(t+\Delta t) + \ln[1 - e^{-k(\Delta t)}] \quad [7.7]$$

Taking the difference between equations 7.6 and 7.7:

$$\ln[T_{t+\Delta t} - T_{t+2\Delta t}] - \ln[T_t - T_{t+\Delta t}] = -k(\Delta t) \quad [7.8]$$

Rearranging:

$$k = \{ \ln[T_t - T_{t+\Delta t}] - \ln[T_{t+\Delta t} - T_{t+2\Delta t}] \}/\Delta t \quad [7.9]$$

The two differencing operations compensate for the lack of knowledge of the exact equilibrium temperature and the exact initial temperature. Equation 7.9 is a Guggenheim method [equation 3, Guggenheim 1926] which after the first two points gives one estimate of the parameter, k, for each temperature measurement. The process can easily be automated and the resulting instrument might be referred to as a 'Guggenheim meter' [P. R. Rony 1981]. With the increasing availability and computational power of microcomputers, mathematical 'meters' such as equation 7.9 are becoming an attractive technique. Figure 7.2 illustrates the Guggenheim calculation by means of a block diagram. The general algorithm in Figure 7.2 may also be expressed as a series of equations.

Let the model of the dynamic system be:

$$\frac{dy}{dx} = -\theta \, (y - y_{ambient}) \quad [7.10]$$

where x is an independent variable (typically time)

x_k

$\ln(y_k - y_{k+1})$

$\ln(y_{k-1} - y_k)$
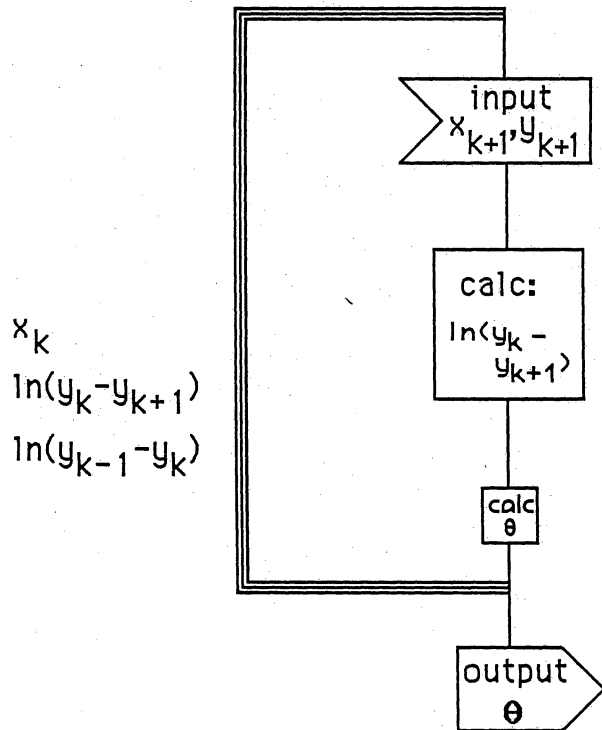
Figure 7.2 Calculation scheme for Guggenheim meter

y is a dependent variable (a measured quantity)

and θ is a parameter (a rate constant of some type)

At each step the following values are known:

x_k, y_k, $\ln(y_k - y_{k-1})$

(1)   measure x_{k+1}, and $y_{k+1}P$ [7.11]

(2)   calculate $\ln(y_k - y_{k+1})$ [7.12]

(3)   calculate $\theta = \{\ln(y_k - y_{k-1}) - \ln(y_k - y_{k+1})\}/(x_{k+1} - x_k)$ [7.13]

(4)   update the stored values:

$x_k := x_{k+1}$ [7.14]

$y_k := y_{k+1}$ [7.15]

$\ln(y_k - y_{k-1}) := \ln(y_{k+1} - y_k)$ [7.16]

and go back to step (1)

Note: The symbol ':=' is taken from PASCAL notation and is read
'is replaced by'.

The area of each block is a rough approximation to the amount of computer
time required to execute the calculation, and the three lines indicating
feedback show the number of values which must be stored between
measurements. The result of using equation 7.9 on the data of figure 7.1

is shown in figure 7.3. The figure illustrates the fact that the two differencing operations exaggerate the noise considerably. While this is an interesting and widely used technique, it leaves much to be desired. First of all, it applies only to systems with a single parameter. Second, the dynamic system (for the kinetics, transport phenomena, etc.) must be linear. Third, the logarithm function may bias the estimates of the parameters. And finally, as seen above, the technique unfortunately exaggerates the noise in the experimental measurements. This problem with noise is apt to be quite troublesome. So what might be done about it?

One way of taking a series of measurements and calculating a parameter in the presence of noise, is least-squares curve fitting. One assumes some model of the system, and calculates parameters for this model such that the sum of the square of the errors between the model and the experimental data is minimized. For illustration purposes let us take the simple linear equation below as a model of a system:

$$y = b_2 x + b_1 + \epsilon \qquad [7.17]$$

where ϵ is the error in the assumed model

Take a series of measurements:

$$\mathbf{x} = [x_1, x_2, \dots x_n]^T \qquad [7.18]$$

$$\mathbf{y} = [y_1, y_2, \dots y_n]^T \qquad [7.19]$$
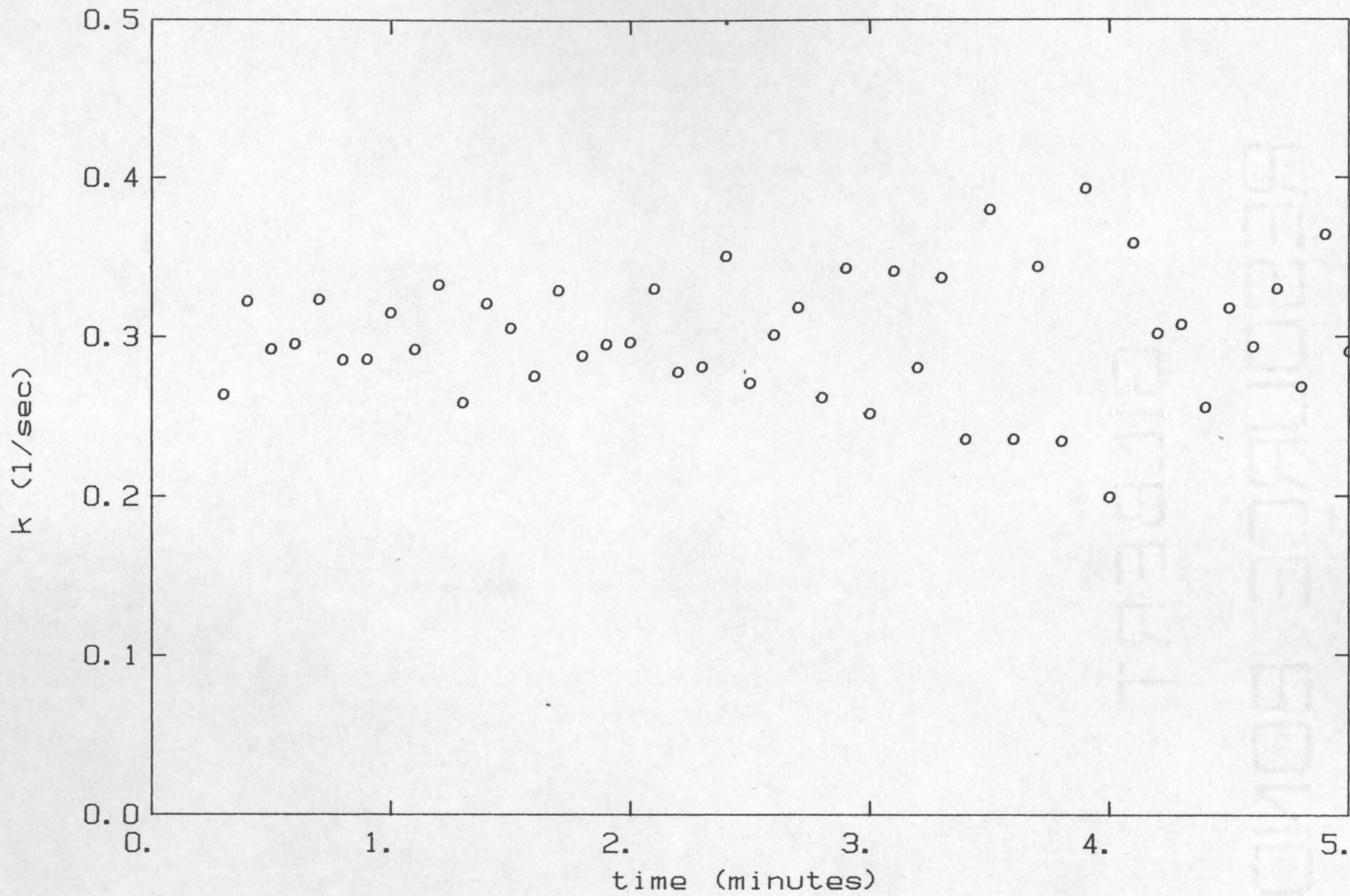
Figure 7.3   Guggenheim meter values for the rate of heat loss
from cooling coffee cup

Note: **bold** face is used to denote vector or matrix quantities

and the superscript (^{T}) indicates 'transpose'
(see any linear algebra text)

Define:

$$\boldsymbol{\varphi}_i = [\; x_i \;\; 1 \;]$$
[7.20]

and

$$\boldsymbol{\theta} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$
[7.21]

For any one experimental point,

$$y_i = \boldsymbol{\varphi}_i \, \boldsymbol{\theta} + \epsilon_i$$
[7.22]

Take an assemblage of all of the experimental points,

$$\mathbf{A} = \begin{bmatrix} \boldsymbol{\varphi}_1 \\ \boldsymbol{\varphi}_2 \\ . \\ . \\ . \\ \boldsymbol{\varphi}_n \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ . \\ . \\ . \\ \epsilon_n \end{bmatrix}$$
[7.23]

The linear relationship for all of the experimental points can be
expressed in matrix form:

$$\boldsymbol{\epsilon} = \mathbf{y} - \mathbf{A}\, \boldsymbol{\theta}$$
[7.24]

The goal is to find the 'best' fit to the linear equation, using the
principle of least squares. The sum of the errors squared is:

$$\sum_1 \text{err}^2 = \sum_1 [\, y_i - (\, b_2\, x_i + b_1 \,)\,]^2$$
[7.25]

$$= [y - A\theta]^T[y - A\theta] \tag{7.26}$$

$$= \epsilon^T\epsilon \tag{7.27}$$

The least squares condition is satisfied by minimizing the sum of the errors over all possible choices of the parameter vector, θ:

$$\frac{\partial \epsilon^T\epsilon}{\partial \theta} = 0 \tag{7.28}$$

The notation in equation 7.28 for the partial derivative of a scalar ($\epsilon^T\epsilon$) with respect to a vector (θ) may be unfamiliar. If the vector is a $1 \times n$ (column) vector the derivative may be defined as a $n \times 1$ (row) vector where each element is:

$$\frac{\partial s}{\partial c} \equiv r \qquad \text{where } r_i \equiv \frac{\partial s}{\partial c_i} \tag{7.29}$$

A similar definition can be given for the partial derivative of a column ($1 \times n$) vector with respect to another column ($1 \times m$) vector resulting in an $n \times m$ matrix:

$$\frac{\partial u}{\partial v} \equiv J \qquad \text{where } J_{ik} \equiv \frac{\partial u_i}{\partial v_k} \tag{7.30}$$

This matrix is called the Jacobian matrix.

Expand the left hand term of equation 7.28:

$$\frac{\partial \epsilon^T\epsilon}{\partial \theta} = \frac{\partial \{ [y - A\theta]^T[y - A\theta] \}}{\partial \theta} \tag{7.31}$$

Expand the numerator on the right hand side of 7.31

$$= \frac{\partial}{\partial \theta} \{ y^T y - y^T A \theta - \theta^T A^T y + \theta^T A^T A \theta \} \tag{7.32}$$

Take the derivative indicated in equation 7.32, making use of this property of the transpose:

$$(AB)^T = B^T A^T \tag{7.33}$$

and defining the partial derivative of a constant times a vector (a right multiplied vector) in the same way as a constant times a scalar:

$$\frac{\partial A\theta}{\partial \theta} = A \tag{7.34}$$

The result is (from equation 7.32):

$$\frac{\partial \epsilon^T \epsilon}{\partial \theta} = -2 y^T A + 2 A^T A \theta \tag{7.35}$$

One can see that if this expression is to equal zero then:

$$0 = -A^T y + A^T A \theta \tag{7.36}$$

Rearranging gives:

$$A^T A \theta = A^T y \tag{7.37}$$

and,

$$\theta = [A^T A]^{-1} A^T y \tag{7.38}$$

Equation 7.38 is the matrix form of the classic linear least squares regression formula. While equations 7.17, 7.20, 7.21, and 7.25, for

illustrative reasons, show only two parameters in the linear model (b_1 and b_2), the rest of the derivation (equations 7.18 to 7.38) applies to any number of parameters by merely making the dimension of the parameter vector, θ, larger.

Note: The use of the superscript (-1) notation for matrix inverse may be unfamiliar. The following definition of the notation will suffice:

$$\text{inverse}(\mathbf{A}) \equiv 1/\mathbf{A} \equiv \mathbf{A}^{-1} \qquad [7.39]$$

For further discussion of matrix inversion see any linear algebra text or Davis [1984].

For the two parameter case (equations 7.17, 7.20, 7.21, and 7.25):

$$\mathbf{A} = \begin{bmatrix} x_1 & 1 \\ x_2 & 1 \\ \cdot & \cdot \\ \cdot & \cdot \\ \cdot & \cdot \\ x_n & 1 \end{bmatrix} \quad \text{and} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \cdot \\ \cdot \\ \cdot \\ y_n \end{bmatrix} \qquad [7.40]$$

$$\mathbf{A}^T\mathbf{A} = \begin{bmatrix} \sum x^2 & \sum x \\ \\ \sum x & n \end{bmatrix} \qquad [7.41]$$

$$[A^TA]^{-1} = \begin{bmatrix} \dfrac{n}{n\,\Sigma\,x^2 - (\,\Sigma\,x\,)^2} & \dfrac{-\,\Sigma\,x}{n\,\Sigma\,x^2 - (\,\Sigma\,x\,)^2} \\[3ex] \dfrac{-\,\Sigma\,x}{n\,\Sigma\,x^2 - (\,\Sigma\,x\,)^2} & \dfrac{\Sigma\,x^2}{n\,\Sigma\,x^2 - (\,\Sigma\,x\,)^2} \end{bmatrix} \qquad [7.42]$$

$$A^Ty = \begin{bmatrix} \Sigma\,x\,y \\[2ex] \Sigma\,y \end{bmatrix} \qquad [7.43]$$

and,

$$\theta = [A^TA]^{-1} = \begin{bmatrix} \dfrac{n\,\Sigma\,x\,y - \Sigma\,x\;\Sigma\,y}{n\,\Sigma\,x^2 - (\,\Sigma\,x\,)^2} \\[4ex] \dfrac{\Sigma\,x^2\;\Sigma\,y - \Sigma\,x\;\Sigma\,x\,y}{n\,\Sigma\,x^2 - (\,\Sigma\,x\,)^2} \end{bmatrix} = \begin{bmatrix} b_1 \\[2ex] b_2 \end{bmatrix} \qquad [7.44]$$

The parameter values (b_1 and b_2) in equation 7.44 are the least squares regression parameters (slope and intercept) in the form usually seen in texts. The results of the matrix approach is exactly the same as other derivations, but the matrix formulation can easily be extended to greater numbers of parameters.

The least squares approach, by taking into account a number of experimental measurements, can reduce the effect of noise from individual measurements. The expression in equation 7.38 may be called *batch least squares,* because the calculation is performed on a whole batch of data. Figure 7.4 shows the results of using the equation 7.7 applied to the cooling curve data of figure 7.1 with a line from the batch least squares calculation of equation 7.38. The slope of the least squares line is the rate constant, k.

While we have improved the noise rejection characteristics, we have lost the elegant form of the Guggenheim formula, where each new measurement will yield one new estimate of the parameter. It is true that the entire batch calculation can be repeated for each measurement. However, in terms of storage and execution time requirements, this approach can quickly get out of hand. As the number of parameters increases, the storage area and execution time become prohibitive. Figure 7.5 shows a block diagram (which implements equation 7.38) for batch least squares. As with figure 7.2, the size of the blocks represent the computation time (relative to the Guggenheim technique of fig 7.2) and the number of lines feeding back to the start represents the storage requirements. These requirements are not too expensive for computation if there are only two parameters to estimate. But in the general scheme of equation 7.38, the amount of calculation and the storage increases with each point making this method unacceptable for continuous parameter 'meters'. When it is important to get a new estimate of the parameters for each experimental measurement, we turn to techniques that are known

slope = -k = -0.3

T(t) - T(t+1) (deg C)

time (minutes)
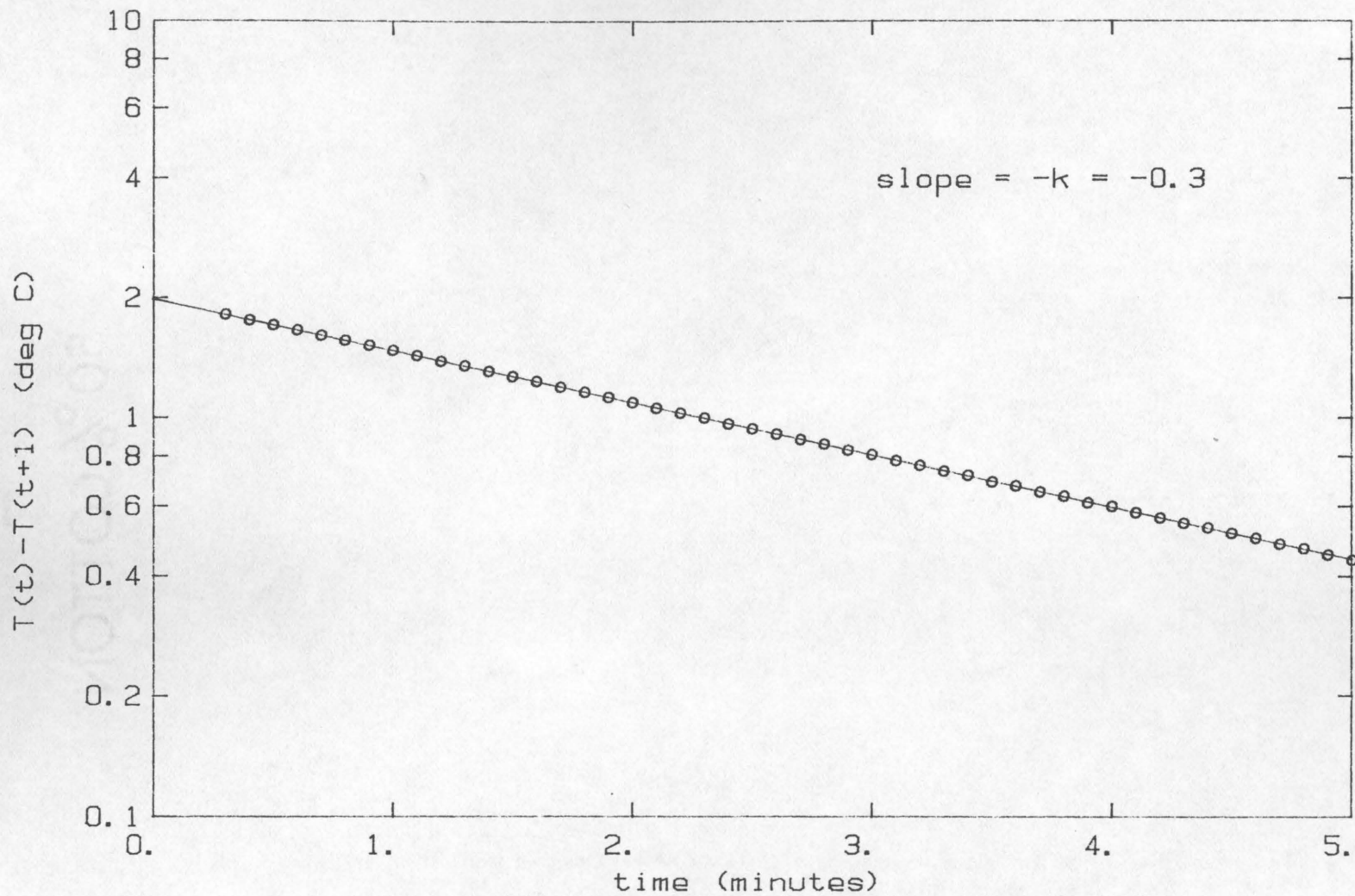
Figure 7.4 Batch least squares parameter estimation
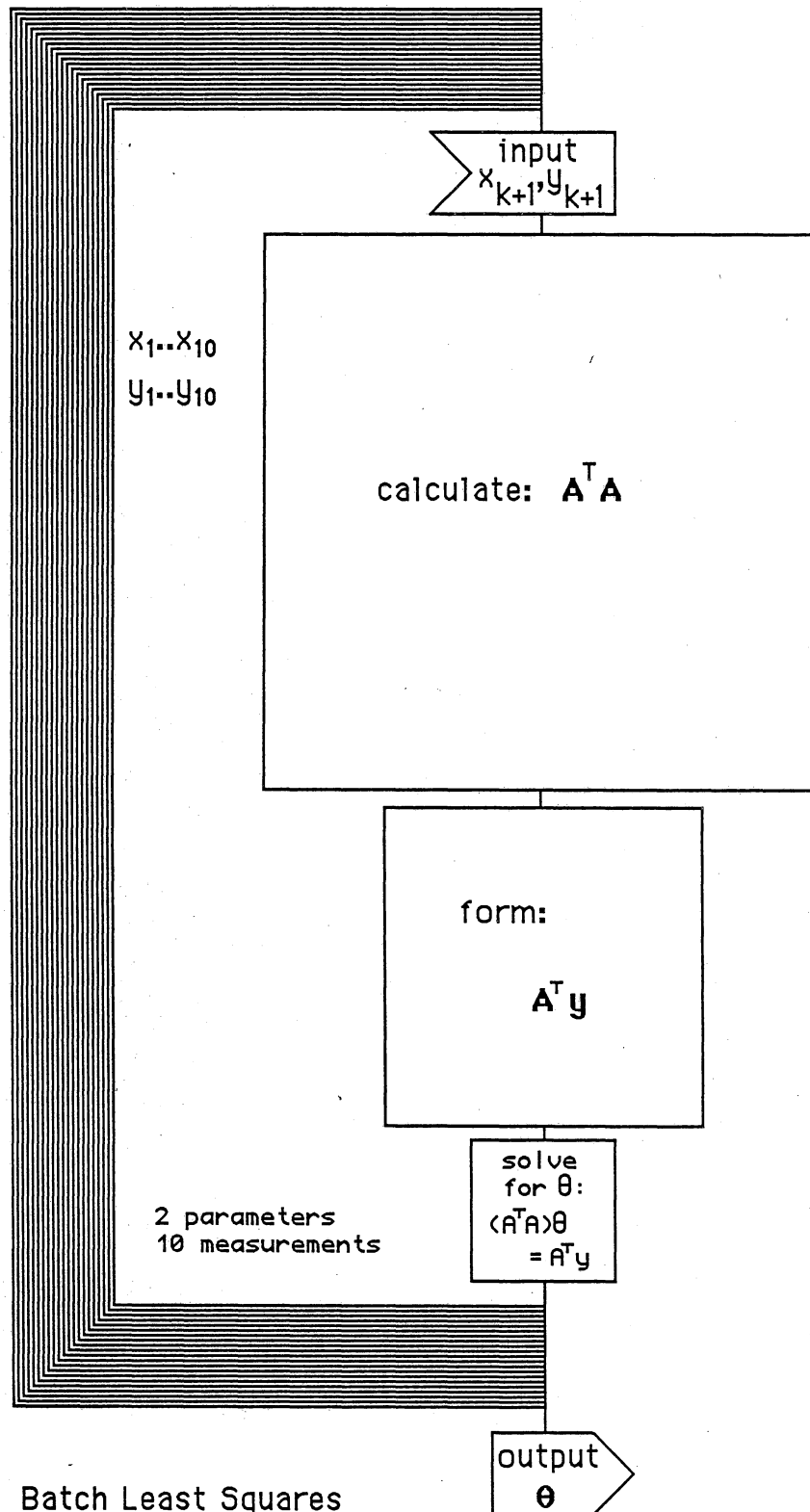
Figure 7.5 Batch Least Squares

as recursive methods. A *recursive algorithm* is a formulation such that for each new measurement point, one needs only certain parameters from the previous estimate and the value of the new measurement. That is, one does <u>not</u> store all the values of the previous measurements! The above least squares formulation (equation 7.38) can be rearranged to a recursive form.

To begin, let us first make an important modification to our least squares criterion. Let different points be weighted by different amounts. The motivation for this is that due to noise or changing experimental conditions, one might want certain measurements to assume greater importance than others. The individual weighting factors will be denoted by the symbol: λ. The weighted sum of the squares will then be [Ljung and Söderström 1983]:

$$\sum_i \lambda_i \, \epsilon_i^2 = \epsilon^T \Lambda \, \epsilon \qquad [7.45]$$

substituting this in the derivation above, one can see that the resulting least squares solution will be:

$$\theta = [A^T \Lambda \, A]^{-1} \, A^T \Lambda \, y \qquad [7.46]$$

Now let us consider the case when one has made some previous estimate of the parameters, and one wishes to update the parameters based on the results of a single measurement. Define a new matrix quantity [Ljung and Söderström 1983]:

$$R_{(t)} = A^T_{(t)} \, \Lambda_{(t)} \, A_{(t)} = \sum_{i=0}^{t} \lambda_i \, \varphi_i \, \varphi_i^T \qquad [7.47]$$

where the subscript $_{(t)}$ indicates that these are the results at time = t

Then the least squares formula becomes,

$$\theta_{(t)} = [R_{(t)}]^{-1} A^T_{(t)} \Lambda_{(t)} y_{(t)} \tag{7.48}$$

and,

$$R_{(t)} \theta_{(t)} = A^T_{(t)} \Lambda_{(t)} y_{(t)} \tag{7.49}$$

Since

$$R_{(t)} = \sum_{i=0}^{t} \lambda_i \varphi_i \varphi_i^T \tag{7.50}$$

then

$$R_{(t+1)} = \sum_{i=0}^{t} \lambda_i \varphi_i \varphi_i^T + \lambda_{(t+1)} \varphi_{(t+1)} \varphi^T_{(t+1)} \tag{7.51}$$

or

$$R_{(t+1)} = R_{(t)} + \lambda_{(t+1)} \varphi_{(t+1)} \varphi^T_{(t+1)} \tag{7.52}$$

Similarly,

$$[A^T \Lambda y]_{(t+1)} = [A^T \Lambda y]_{(t)} + \varphi_{(t+1)} \lambda_{(t+1)} y_{(t+1)} \tag{7.53}$$

Substituting these expressions into the least squares expression:

$$\theta_{(t+1)} = [R_{(t+1)}]^{-1} A^T_{(t+1)} \Lambda_{(t+1)} y_{(t+1)} \tag{7.54}$$

gives,

$$\theta_{(t+1)} = [R_{(t+1)}]^{-1} \{ [A^T \Lambda y]_{(t)} + \varphi_{(t+1)} \lambda_{(t+1)} y_{(t+1)} \} \tag{7.55}$$

$$= [R_{(t+1)}]^{-1} \{ R_{(t)} \theta_{(t)} + \varphi_{(t+1)} \lambda_{(t+1)} y_{(t+1)} \} \tag{7.56}$$

$$= [R_{(t+1)}]^{-1} \left\{ \; [\, R_{(t+1)} - \lambda_{(t+1)} \, \varphi_{(t+1)} \, \varphi^T_{(t+1)} \,] \; \Theta_{(t)} \right.$$
$$\left. + \; \varphi_{(t+1)} \, \lambda_{(t+1)} \, y_{(t+1)} \qquad \right\}$$
$$[7.57]$$

$$= [R_{(t+1)}]^{-1} \left\{ \quad R_{(t+1)} \, \Theta_{(t)} \right.$$
$$- \; \lambda_{(t+1)} \, \varphi_{(t+1)} \, \varphi^T_{(t+1)} \; \Theta_{(t)}$$
$$\left. + \; \varphi_{(t+1)} \, \lambda_{(t+1)} \, y_{(t+1)} \qquad \right\}$$
$$[7.58]$$

Because λ is a scalar,

$$\Theta_{(t+1)} = [R_{(t+1)}]^{-1} \left\{ \quad R_{(t+1)} \, \Theta_{(t)} \right.$$

$$- \; \lambda_{(t+1)} \, \varphi_{(t+1)} \, \varphi^T_{(t+1)} \; \Theta_{(t)}$$

$$\left. + \; \lambda_{(t+1)} \, \varphi_{(t+1)} \, y_{(t+1)} \qquad \right\}$$
$$[7.59]$$

and

$$\Theta_{(t+1)} = \Theta_{(t)} + [R_{(t+1)}]^{-1} \, \lambda_{(t+1)} \, \varphi_{(t+1)} \, [\, y_{(t+1)} - \varphi^T_{(t+1)} \, \Theta_{(t)} \,]$$
$$[7.60]$$

If one defines a gain, L, as:

$$L_{(t+1)} = [R_{(t+1)}]^{-1} \, \lambda_{(t+1)} \, \varphi_{(t+1)} \qquad\qquad [7.61]$$

Then

$$\Theta_{(t+1)} = \Theta_{(t)} + L_{(t+1)} \, \epsilon_{(t+1)} \qquad\qquad [7.62]$$

and, from a previous result,

$$R_{(t+1)} = R_{(t)} + \lambda_{(t+1)} \, \varphi_{(t+1)} \, \varphi_{(t+1)}^T \qquad\qquad [7.63]$$

The previous three equations (7.61 to 7.63) give a recursive algorithm for least squares analysis of sequential data. This algorithim can be expressed in the form of a block diagram as in figure 7.6. As in the previous block diagrams (figures 7.2 and 7.5), the area of the blocks represent computation time and the lines running from the bottom of the diagram back to the top represent the number of variables stored between measurements. There is one serious problem with this algorithm: the matrix, R, must be inverted at each step ($[R_{t+1}]^{-1}$). This inversion process can entail very large amounts of computer time and can also result in severe numerical problems due to round-off error in the computer. To circumvent these problems, one might define:

$$P_{(t)} = [R_{(t)}]^{-1} \tag{7.64}$$

And update the P matrix directly.

Also

$$[P_{(t)}]^{-1} = R_{(t)} \tag{7.65}$$

$$P_{(t+1)} = [R_{(t+1)}]^{-1} \tag{7.66}$$

and

$$[P_{(t+1)}]^{-1} = R_{(t+1)} \tag{7.67}$$

To accomplish the updating of the matrix, R, one can turn to a result from linear algebra known as the *matrix inversion lemma*:

Given matrices A, B, C, and D (of compatible dimensions):

$$[A+BCD]^{-1} = A^{-1} - A^{-1}B[BA^{-1}B + C^{-1}]^{-1}DA^{-1} \tag{7.68}$$

A proof of this result may be given simply [Ljung and Söderström 1983]:

Multiply both sides of the equation by $[A+BCD]$,
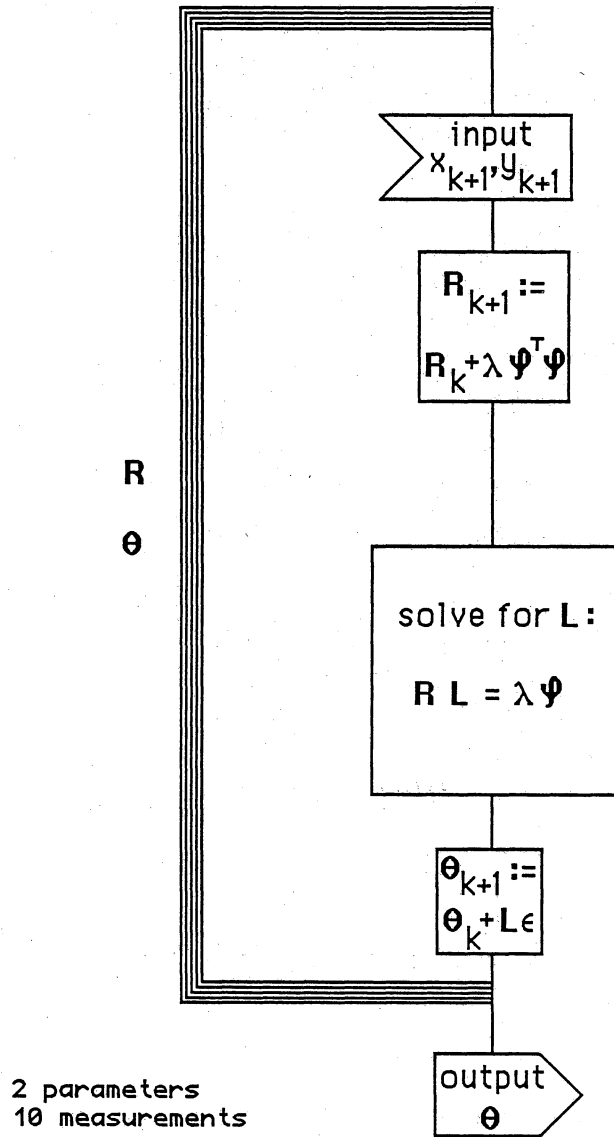
Figure 7.6 Recursive least squares algorithim

$$[A+BCD]^{-1}[A+BCD] = \{ A^{-1} - A^{-1}B [DA^{-1}B + C^{-1}]^{-1} DA^{-1}\} [A+BCD]$$

[7.69]

$$I = A^{-1}[A+BCD] - A^{-1}B [DA^{-1}B + C^{-1}]^{-1} DA^{-1}[A+BCD] \quad [7.70]$$

$$I = I + A^{-1}BCD - A^{-1}B [DA^{-1}B + C^{-1}]^{-1} [D+DA^{-1}BCD] \quad [7.71]$$

$$I = I + A^{-1}B \; CD$$
$$- A^{-1}B [DA^{-1}B + C^{-1}]^{-1} D$$
$$- A^{-1}B [DA^{-1}B + C^{-1}]^{-1} DA^{-1}BCD \quad [7.72]$$

$$I = I + A^{-1}B \; \{ \quad CD$$
$$- [DA^{-1}B + C^{-1}]^{-1} D$$
$$- [DA^{-1}B + C^{-1}]^{-1} DA^{-1}BCD \; \} \quad [7.73]$$

$$I = I + A^{-1}B [DA^{-1}B + C^{-1}]^{-1} \{ [DA^{-1}B + C^{-1}] CD$$
$$- D$$
$$- DA^{-1}BCD \quad \} \quad [7.74]$$

$$I = I + A^{-1}B [DA^{-1}B + C^{-1}]^{-1} \{ \quad DA^{-1}BCD + D$$
$$- D$$
$$- DA^{-1}BCD \quad \} \quad [7.75]$$

$$I = I + A^{-1}B [DA^{-1}B + C^{-1}]^{-1} \{ 0 \} \quad [7.76]$$

$$I = I \quad [7.77]$$

Q.E.D.

Make the following assignments:

$$A = P_{(t)} \qquad B = \varphi_{(t+1)} \qquad C = \lambda_{(t+1)} \quad \& \quad D = \varphi_{(t+1)}^T \qquad [7.78]$$

Comparing these relationships, the matrix inversion lemma, and the update formula above (remember that λ is a scalar),

$$R_{(t+1)} = R_{(t)} + \lambda_{(t+1)} \varphi_{(t+1)} \varphi^T_{(t+1)} \qquad\qquad [7.79]$$
$$\text{or} \quad = A + B \quad C \quad D \qquad\qquad [7.80]$$

and

$$P_{(t+1)} = [P_{(t)}^{-1} + \varphi_{(t+1)} \lambda_{(t+1)} \varphi^T_{(t+1)}]^{-1} \qquad [7.81]$$

So,

$$P_{(t+1)} = P_{(t)}$$
$$- P_{(t)} \varphi_{(t+1)} [\varphi_{(t+1)} P_{(t)} \varphi_{(t+1)} + 1/\lambda_{(t+1)}]^{-1} \varphi^T_{(t+1)} P_{(t)}$$
$$[7.82]$$

$$\text{or} \quad = A^{-1}$$
$$- A^{-1} B [D \quad A^{-1} \quad B + C^{-1}]^{-1} D \quad A^{-1}$$
$$[7.83]$$

Rearranging,

$$P_{(t+1)} = P_{(t)} - \frac{P_{(t)} \varphi_{(t+1)} \varphi^T_{(t+1)} P_{(t)}}{[\varphi^T_{(t+1)} P_{(t)} \varphi_{(t+1)} + 1/\lambda_{(t+1)}]} \qquad [7.84]$$

The advantage of this expression is that the quantity in brackets in the denominator is a scalar (for scalar measurements); finding the inverse of this quantity is trivial compared to inverting the previous

matrix (R) which has a dimension equal to the number of variables being estimated. The resulting algorithm is:

$$L_{(t+1)} = \frac{P_{(t)} \varphi_{(t+1)}}{[\ \varphi^T_{(t+1)} P_{(t)} \varphi_{(t+1)} + 1/\lambda_{(t+1)}\ ]} \qquad [7.85]$$

$$\Theta_{(t+1)} = \Theta_{(t)} + L_{(t+1)} \epsilon_{(t+1)} \qquad [7.86]$$

$$P_{(t+1)} = P_{(t)} - \frac{P_{(t)} \varphi_{(t+1)} \varphi^T_{(t+1)} P_{(t)}}{[\ \varphi^T_{(t+1)} P_{(t)} \varphi_{(t+1)} + 1/\lambda_{(t+1)}\ ]} \qquad [7.87]$$

$$= P_{(t)} - L_{(t+1)} \varphi^T_{(t+1)} P_{(t)} \qquad [7.88]$$

Equations 7.86 to 7.88 constitute a practical recursive least squares algorithim. The structure of this algorithim is illustrated in figure 7.7, which shows the computation time and the storage requirements in the same way as figures 7.2, 7.5, and 7.6.

Another way to consider the recursive least squares algorithim is in the form of a block diagram, as in figure 7.8
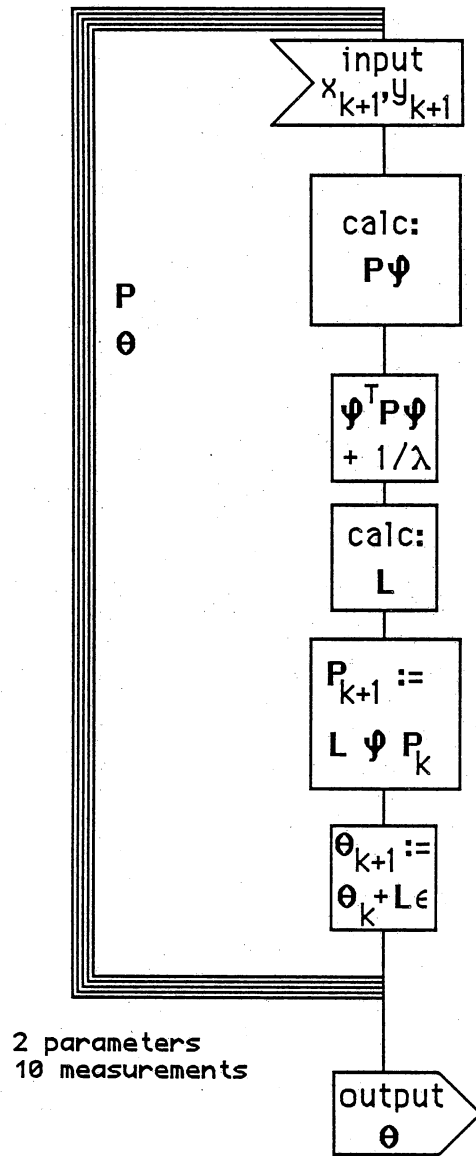
Figure 7.7 Recursive least squares making use of the
matrix inversion lemma

Figure 7.8 Block diagram for recursive parameter estimation
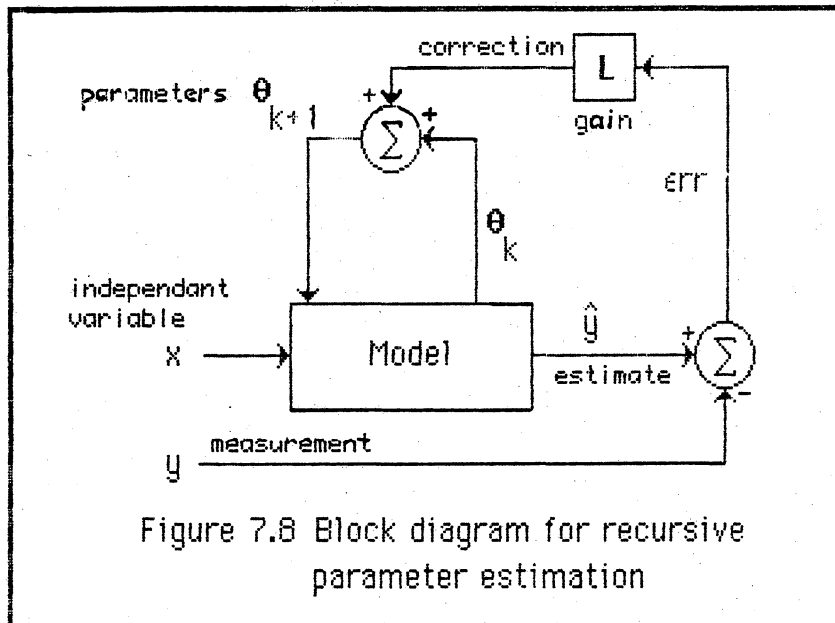
This figure shows the measurements entering a mathematical model on the left. The model calculates an estimate of the measurement, $\hat{y}$, and takes the difference between the estimate and the measurement, $\bar{y}$. This difference is an ϵrr signal, which multiplied by a gain, L, gives a correction term for the parameters, Θ. This correction term is added to the old values of the parameters to give a new estimate for the parameters, which will be used by the model the next time around.

Consider this algorithm from an intuitive point of view. The variables may be summarized as follows. The vector φ consists of the independent variables in the model of the process. The model has a parameter vector Θ, the elements of which are the actual quantities of interest: these are the reasons for performing the experiment and taking the measurements. The quantity ϵ is the error term derived from comparing the model- predicted dependent variable to the measured variable in the experiment. The matrix P is, in effect, a summary of all

of the past independent variable data. Since P is related to the term $(A^T A)^{-1}$ in the batch least squares formula, P may be referred to as a 'covariance matrix' [Bierman 1977, pg 16]. And since P is related to an estimate of the random error in the experiment (covariance), it seems reasonable that any correction term to the parameters of the model will depend on the covariance term (P), the dependent variables (φ), and the error (ϵ) of the model. The vector L is a gain term calculated from the covariance estimate, the value of the independant variables, and the weighting factor (λ). The error times this gain factor gives a correction term for the model parameters (θ). Finally, the covariance P is corrected to include the last experimental point. The recursive nature of the algorithm allows a computer to only store and operate on the matrix P rather than the whole series of measurements.

A FORTRAN program which incorporates a recursive least squares algorithm is shown in Listing 27. Figure 7.9 shows the results of applying this program to the cooling curve of Figure 7.1. Note how the initial guess of the parameter (θ_1, which is related to k) converges on a statistical estimate of the parameter. There is still some effect of noise in the measurements (and also truncation error in the calculation), but the result is much improved over the Gugenheim meter of Figure 7.3.

To reiterate, the structure of the recursive algorithm (equations 7.77, 7.78, and 7.80) is in three parts. First, a gain factor, L, is calculated. Second, this gain and the model error is used to correct the estimate of the model parameters. And third, the stored information is adjusted to take into account the additional experimental point.
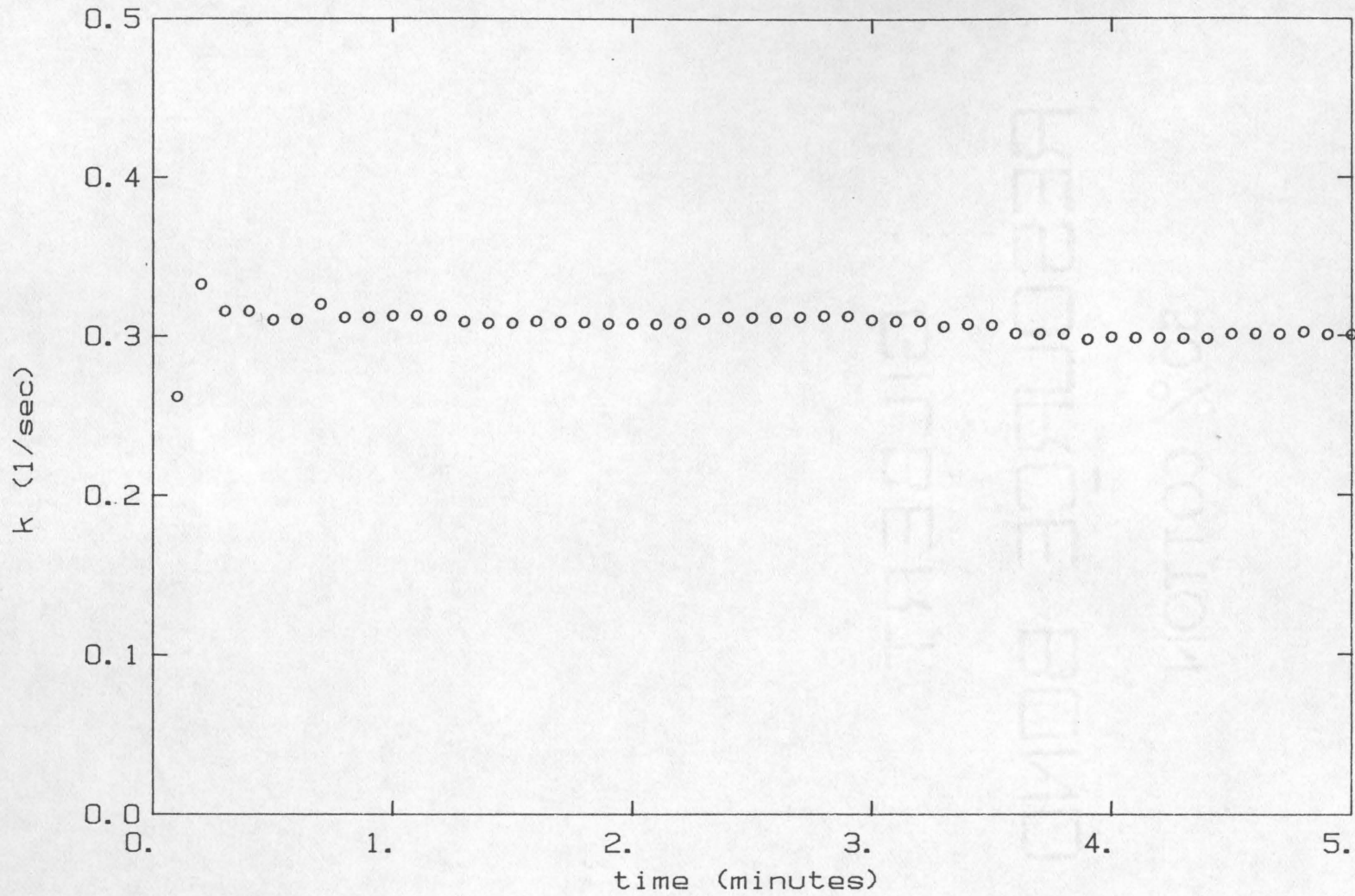
Figure 7.9 Recursive parameter estimation meter values
for the rate of heat loss from a cooling coffee cup

The fitted model so far has not explicitly incorporated a dynamic model of the system; that is, a model described by a time dependent differential equation. An electrochemical system obviously requires an estimation scheme which includes such a dynamic model. A common way of representing the time dependence of a model is the so-called 'state space' approach. In this technique of representing a dynamic system, the rate of change of the system is a function of a set of variables describing the condition of the system.

Consider a simple example, a heated object of high thermal conductivity is dropped into a well-stirred container of liquid. This system is dynamic because one expects the temperatures to change with time, the solid will lose heat to the liquid and the liquid will in turn lose heat to the surroundings. In this simple case, one might consider the temperature of the solid and the temperature of the liquid to describe the condition of the system. The 'state space' approach would call these 'state variables'; these variables can be considered sufficient to describe the actions of this simple system. Recall a definition from a previous chapter: the state variables of a system are those variables sufficient to describe the system at any time such that the time behavior of the system can be calculated. If the state variables are known, then the present condition of the system is fixed. This is why they are known as 'state' variables; they fix the state of the system. The expression 'state space' refers to the full possible range of the variables, that describe the system.

Are the particular variables of solid and liquid temperature necessary variables to describe the simple system? No. Instead of the temperature of the solid, consider the rate of change of the temperature of the solid. Using this rate and the temperature of the liquid, one could just as well describe the entire time history of the system. Thus the choice of state variables is not unique. One picks the describing variables based on some other criterion: perhaps ease of measurement (heat flux is not directly measurable), perhaps ease of calculation, perhaps to minimize the sensitivity of the calculations to errors in the assumed parameters, or perhaps to minimize the effects of truncation error in the computer solution of the model, etc.

In the simple temperature system, or for any combination of the two variables, that is for any condition or state of the system, the rate of change of the system with time can be described. The system model is represented as a set of simultaneous first-order differential equations:

$$\dot{x} = f(x, \theta)$$ [7.89]

where x is the vector of the state variables

$\dot{x}$ is the vector of the rates of change of those variables with respect to time

θ is a vector of the parameters of the model

$f(\ ,\ )$ is a vector function of the states and the parameters

One can see that the mathematical model of the electrochemical system in Chapter 4 is posed in just such a state space fashion.

In a mathematical model, the state variables are known, but in a system with only limited measurements available, the state variables may not be directly known. The ideas of estimation may be used for calculating some approximation of the state of the system.

The work of Kalman [1958, 1963] is directed towards exactly this situation. He reasoned from a statistical basis for the calculation of optimal estimates of the values of the state variables from the available measurements. The result is known as the 'Kalman filter', because it filters out the estimates of the states from measurements which have noise mixed in. In keeping with the comments at the beginning of this chapter, one could refer to a 'Kalman meter', that is a superset of the previously mentioned 'Guggenheim meter'. The Kalman meter preserves the idea of repetitively updated estimates of the desired parameters (recursion), but extends the concept to more realistic (and more complicated) system models and shows how to make optimal estimates in the presence of noise.

To give an example of Kalman's results, the simplest example uses a discrete model (that is, piecewise constant) of the system [Franklin and Powell 1980]. Assume the following model of the system:

$$x(t+1) = \Phi\, x(t) \;+\; \Gamma\, u(t) \;+\; w(t) \qquad\qquad [7.90]$$

$$y(t+1) = C\, x \;+\; e(t) \qquad\qquad [7.91]$$

where x is the vector of states

        Φ is a matrix known as the state transition matrix

        Γ is a matrix known as the control transition matrix

u is the vector of inputs to the system

w(t) is a random variable representing noise in the system

y is the measurement

e(t) is a random variable representing noise in the measurement

C is the measurement matrix

If x is an estimate of the state vector then the Kalman filter for this system is:

$$K_{(t+1)} = \frac{\Phi \, P_{(t)} \, C^T}{[\; C \, P_{(t)} \, C^T + r_2 \;]} \tag{7.92}$$

$$x_{(t+1)} = \Phi \, x_{(t)} + K_{(t)} [\; y_{(t)} - C \, x_{(t)} \;] \tag{7.93}$$

$$P_{(t+1)} = \Phi \, P_{(t)} \, \Phi^T + R_1 - \frac{\Phi \, P_{(t)} \, C^T \, C \, P_{(t)} \, \Phi^T}{[\; C \, P_{(t)} \, C^T + r_2 \;]} \tag{7.94}$$

where R_1 is the covariance matrix of the system noise, w(t)

and r_2 is the covariance of the measurement noise, e(t)

Observe the structural similarity of this algorithm to that of the previous least squares algorithm (equations 7.77, 7.78, and 7.79). If the following replacements are made: $\Phi = I$, $R_1 = 0$, $r_2 = 1/\lambda$, $K = L$, $x = \theta$, and $C = \varphi^T$, then the formulae are exactly the same! This similarity is, in fact, the central thesis of the book by Ljung and Söderström [1983]: "There is only one recursive identification method. It contains some design variables to be chosen by the user."

The Kalman filter provides an estimate of the state and requires knowledge of the noise structure of the system. The standard Kalman

filter requires knowledge of a system model, just as the other estimation algorithms discussed in this chapter, and it requires that the model be linear in the state variables. By augmenting the state vector with a vector for parameters, one can incorporate both parameter <u>and</u> state estimation. And by linearizing the model about the current estimate of the state and parameters, one can accommodate nonlinear system models. The result is known as the 'Extended Kalman Filter' [Ljung and Söderström 1983].

Let:

$$X = \begin{bmatrix} x \\ \theta \end{bmatrix}$$, the augmented state and parameter vector [7.95]

$$P = \begin{bmatrix} P_1 & P_2 \\ P_2 & P_3 \end{bmatrix}$$, the augmented 'covariance' matrix [7.96]

Note that the matrix **P** cannot be referred to exactly as a covariance matrix because it is the result of a non-linear system. Thus it is an approximation to the covariance of the linearized system. But even in linear systems, the calculated covariance matrix is only an estimate of the true covariance.

Among the few examples of the use of the Kalman filter in Chemistry, the most extensive (and the only use in the context of dynamic chemical systems) are papers with S. D. Brown [T. Brown et al 1984, S. C. Rutan and S. D. Brown 1985a,b]. In the 1984 paper Brown et al apply the Extended Kalman Filter (EKF) to cyclic scanning voltammetry. A discrete finite difference model of the electrochemical system is used, and formulation of the EKF assumes that model predicts the state perfectly

(thus no need for state estimation). He then uses the EKF to estimate the electrochemical parameters of the system. The present work differs from Brown et al in that the state variables are not assumed from the model, and a more sophisticated mathematical model of the electrochemical system is used.

To apply the EKF to an oscillogenic system, one first must realize that there is no *a priori* knowledge of the values of the state variables of the system. In an effort to provide a level of computational precision comparable to that customary in electrochemical simulation, while decreasing the amount of computer storage and execution time, this work uses the spline collocation model described in chapter 4. As previously mentioned, this computational procedure used a spline approximation of the spatial concentration profile, while using a continuous representation of the temporal variable.

The state variables are the spline coefficients describing the concentration profile; the state of the system at any time is fully described by the concentration profile. To predict changes with time, the state is used with the input (the applied voltage), and the parameters (rate constant, k_0, the charge transfer coefficient, β, and the standard half-cell potential, $E_{1/2}$) to predict the current output of the system. A comparison with the measured current is used to correct the estimates of the state and parameter variables.

The set of coupled ordinary differential equations describing the electrochemical system is continuous (not discrete) in time. The initial value solver (in the present case, Runge-Kutta-Fehlberg) then solves for

the concentration profile as a function of time. While the Runge-Kutta-Fehlberg initial value solver is discrete in time, the temporal step-size is not constant within a given call to the routine and the routine can adjust the overall temporal step-size. The temporal step-size is adjusted to meet user defined truncation error tolerances, thus the initial value solver treats the problem as continuous in time. The measurements on the system, on the other hand, are at a fixed time intervals. In general, these intervals are a function of the analog-to-digital circuits (hardware) employed, and not continuously variable by the estimation routine (software).

Because the spline collocation model is essentially continuous in time, a continuous version of the Kalman filter is indicated. But the measurements of the current flowing to the electrochemical cell are discrete in nature (from an analog-to-digital converter via a computer). This in turn would indicate the use of a discrete Kalman filter. The solution to this dilemma lies in noting that the covariance (**P**) matrix update in the Kalman filter can be separated into two steps: a time propagation step, and a measurement update. For the discrete model mentioned above, the **P** matrix update can be expressed as:

1. A time update or error covariance prediction:

$$P_{(t+1 \mid y(t))} = \Phi \, P_{(t)} \, \Phi^T + R_1 \qquad\qquad [7.97]$$

and 2. A measurement update or error covariance correction:

$$P_{(t+1|y(t+1))} = [\; I \; - \; K\;C\;]\;\; P_{(t+1|y(t))} \qquad [7.98]$$

$$\text{or}\;\; P_{(t+1|y(t+1))} = \left[\; I \; - \; \frac{P_{(t+1|\;y(t))}\;C^T\;C}{[\;C\;P_{(t)}\;C^T\;+\;r_2\;]}\;\right]\;P_{(t+1|y(t))} \qquad [7.99]$$

Separating the two aspects of updating the covariance matrix into a predictor/corrector pair, opens up the possibility of using different methods for each of the two steps. Gelb [1974] refers to this as the 'continuous/discrete extended Kalman filter'. First, to conform with the continuous time expression for the state model, a continuous covariance time update expression is used:

$$\dot{P}_{(t)} = F\;P\;+\;P\;F^T\;+\;R_1 \qquad [7.100]$$

so P can be integrated over the time step

where

$$F\;=\;\frac{\partial f}{\partial X}\;, \text{ the partial of the nonlinear system model} \qquad [7.101]$$

with respect to the augmented state/parameter vector

Then, for the discrete measurement of the output of the system, a discrete covariance measurement update is used:

$$P_{(t+1|y(t+1))} = [\; I \; - \; K\;C\;]\;\; P_{(t+1|y(t))} \qquad [7.102]$$

measurement correction

As many workers have pointed out [Bierman 1973, 1976, 1977, T. Brown et al 1984, Carlson 1973, Ljung and Söderström 1983], the straight

forward implementation of the Kalman filter is subject to numerical difficulties. The covariance estimate, **P**, is positive definite by definition. However, the truncation errors in the computer implementation of the Kalman algorithm will often allow the computer representation of **P** to become nearly singular or indefinite causing lack of convergence of the filter to the 'true' values. Various methods have been proposed to circumvent this problem. Brown et al [1984] use an algorithm known as the Potter-Schmidt square root covariance algorithm. Bierman [1977] discusses this algorithm in detail and proposes a more efficient and numerically slightly more stable square-root-free Cholesky decomposition method. But these methods do not apply to continuous formulations of the Kalman filter, and are not efficiently implemented for nonlinear models. Bierman [1977] points out a more stable mechanisation of the Kalman algorithm:

$$\mathbf{P}_{(t+1|y(t+1))} = [\mathbf{I} - \mathbf{K}\,\mathbf{C}]\,\mathbf{P}_{(t+1|y(t))}\,[\mathbf{I} - \mathbf{K}\,\mathbf{C}]^T + \mathbf{K}\,r\,\mathbf{K}^T$$

[7.103]

For ease of exposition, the subscripts will be dropped for the following demonstration. Since the **P** terms are not exchanged across the equals sign, the results are unambiguous. The equivalence of this expression to the standard expression can be demonstrated:

$$\mathbf{P} = [\mathbf{P} - \mathbf{K}\,\mathbf{C}\,\mathbf{P}]\,[\mathbf{I} - \mathbf{C}\,\mathbf{K}^T] + \mathbf{K}\,r\,\mathbf{K}^T$$

[7.104]

$$= [\mathbf{P} - \mathbf{K}\,\mathbf{C}\,\mathbf{P}]\,[\mathbf{I} - \mathbf{C}^T\,\mathbf{K}^T] + \mathbf{K}\,r\,\mathbf{K}^T$$

[7.105]

$$= \mathbf{P} - \mathbf{K}\,\mathbf{C}\,\mathbf{P} - \mathbf{P}\,\mathbf{C}^T\,\mathbf{K}^T + \mathbf{K}\,\mathbf{C}\,\mathbf{P}\,\mathbf{C}^T\,\mathbf{K}^T + \mathbf{K}\,r\,\mathbf{K}^T$$

[7.106]

$$= P - K C P + [- P C^T + K C P C^T + K r] K^T \qquad [7.107]$$

$$= P - K C P + [- P C^T + K C P C^T + K r] K^T \qquad [7.108]$$

For the 'continuous-discrete extended Kalman filter' [Gelb1974], the Kalman gain expression for the measurement update is:

$$K = \frac{P C^T}{[C P C^T + r_2]} \qquad [7.109]$$

Then,

$$P = P - K C P + \left[- P C^T + \frac{P C^T C P C^T}{[C P C^T + r_2]} + \frac{P C^T r}{[C P C^T + r_2]} \right] \qquad [7.110]$$

$$= P - K C P + \left[- P C^T + \frac{P C^T [C P C^T + r]}{[C P C^T + r_2]} \right] K^T \qquad [7.111]$$

$$= P - K C P + [0] K^T \qquad [7.112]$$

$$P = P - K C P \qquad [7.113]$$

Q.E.D.

Thus an acceptable algorithm is achieved. The state estimate can be extrapolated by the previously discussed spline collocation model [Chapter 4]. The parameter and state estimates can be corrected by a Kalman gain times the difference between the predicted and observed

system output. The 'covariance' matrix can be extrapolated by a continuous time update [Gelb 1974], and corrected by a discrete measurement update using a numerically improved algorithm [Bierman 1977].

Integrate from (t) to (t+1):

$$\dot{x} = f(x, \theta) \qquad [7.114]$$

$$\dot{P}_{(t)} = F P + P F^T + R_1 \qquad [7.115]$$

Correct the parameter and state estimates,

$$K = \frac{P_{(t+1|y(t))} C^T}{[C P_{(t+1|y(t))} C^T + r_2]} \qquad [7.116]$$

$$\epsilon = y(t+1) - C x_{(t+1|y(t))} \qquad [7.117]$$

$$X_{(t+1|y(t+1))} = X_{(t+1|y(t))} + K \epsilon \qquad [7.118]$$

Update the 'covariance' estimate,

$$P_{(t+1|y(t+1))} = [I - K C] P_{(t+1|y(t))} [I - K C]^T + K r K^T$$
$$[7.119]$$

In summary, this chapter covered the development of a generalized parameter estimation 'meter'. Starting with the Guggenheim idea of a recursive parameter estimation scheme, a way to put the scheme on a statistically justified basis was demonstrated. Kalman's statistically

optimal procedure was shown. The concept was extended to multivariable and nonlinear systems. The numerical stability and the efficiency of the algorithm were considered, and the final algorithm was expressed as a series of formulae.

Chapter 8. Generalizing Oscillogenic Techniques

"The only thing we can state with certainty is that time plays tricks on anyone who claims, 'There will never be any practical use for ____.'"

Arthur Jaffe

Chapters 1, 2, and 3, of this work, examined the relationship of chemical oscillators to physical oscillators. Chapter 2 showed that any physically realizable oscillator had to have a non-linear element to control the stability of the oscillator. Chapter 3 reached the conclusion that the major difference between chemical oscillators and physical oscillators was that nonlinearities <u>dominated</u> the presently known chemical oscillators, while the common physical oscillators are nearly linear. Because nonlinear analysis is not as well developed as linear analysis, the mathematical study of chemical oscillators is much more difficult. Taking engineering control theory as a starting point (Chapter 3), a series of ideas allowed P. R. Rony and P. K. Mercure to propose a method for creating a general chemical oscillator by using a nonlinear external feedback system.

Taking electrochemistry as a model chemical system, a mathematical model was proposed (chapter 4). The solution of this model (chapter 5) allowed the system to be studied in detail and a specific switching procedure to be proposed for oscillogenic feedback. The switching procedure could be expressed as an algorithim or a circuit. This switching procedure was verified by using an experimental system (chapter 6). The resulting instrument did indeed oscillate, and was shown to be sensitive to the concentration of chemical species. The instrument was also selective in its response; it could distinguish between different chemical species. The waveform of such an oscillator can be used with parameter estimation methods as a source of information

(chapter7) allowing chemical parameters to be measured. This final chapter considers what additional types or families of oscillogenic systems can be devised based on the general idea proposed in chapter 3, and the working prototype discussed in chapter 6.

The purpose of this chapter is to suggest the extension of oscillogenic instrumentation to a diverse range of applications. As an analogy consider the case of chromatography. The first instance, and the first exposure of most people to chromatography, is the technique of paper chromatography. Yet consider the vast array of techniques available today: gas/solid, gas/liquid, high performance liquid, ion, hydrodynamic, thin layer, gel permeation, ion exchange, two dimensional, etc. Also, there is little reason to think that the full range of methods has been exposed. The humble paper chromatograph is but a precursor to more sophisticated techniques, but these other techniques share the basic principles of the original method. Specifically, they share the idea of a differential motion between two carriers and the transfer of species between the carriers. Similarly, the electrochemical oscillogenic method discussed in this work has some basic principles that can be shared by other, derivative, methods.

Foremost in these principles is the idea of an *input* and an *output* for a chemical system. The oscillogenic device, by purpose, *stimulates* a chemical system in a certain way and uses the *response* of the system to allow an oscillation to be produced. The *response* or *output* of the system is used in an algorithim <u>outside</u> the chemical system to calculate a changing *stimulus* or *input* to the chemical system.

This outside algorithim is called a *feedback algorithm,* because it feeds the input of the chemical system with processed information taken backwards from the output of the chemical system. The normal flow of

information in the chemical system is from input to output; the external feedback allows a flow of information in the other direction. The feedback mechanism includes a *non-linear function*, which serves to stabilize the oscillations produced by the feedback loop/chemistry combination.

The feedback algorithm suggested (Chapter 3) and used experimentally (Chapter 6) in this work has two specific characteristics which make it a subset of the much larger set of all possible feedback mechanisms. First, the result of the feedback algorithm, the information which is fed into the chemical system, has only two levels. Certain properties of this result can be changed, such as the average electrical potential which was varied for the oscillogenic scans of Chapter 6. But, the actual feedback which allows the oscillation has only a high and a low value at any instant. Second, some type of *hysteresis* is used in the feedback algorithm. The hysteresis in the device used in Chapter 6 is a Schmitt trigger, but other devices were suggested in Chapters 3 and 5.

Certainly, the feedback algorithm suggested by the present work is not the only possible one. Chapter 2 of this work has already discussed the use of predominately linear feedback with a non-linear amplitude control system for stability. This nearly linear feedback was rejected for an electrochemical oscillogenic system for reasons explained in Chapter 3, but other chemical systems might use this approach. There is currently some interest in just such linear feedback, in the form of a proportional controller, to produce oscillations in CSTRs with non-linear thermokinetic behavior [Chang and Chen 1984, Lyberatos et al 1985a, McDermott (in press), Ray and Jensen 1980].

The feedback circuit in an oscillogenic instrument will most likely supply the nonlinearity required for oscillator stabilization. Thus, the

design of the feedback circuit falls within the domain of nonlinear controls. There is a rich literature in the field of nonlinear systems and oscillators. Some of the works found useful in the present work are: Brown 1980, Chang and Chen 1984, Derusso 1965, Epstein 1981, Epstein 1983a, Gilsin 1982, Graham and McRuer 1961, Hale and LaSalle 1963, Lybertos et al 1984, 1985a,b, Minorsky 1962, Nayfeh and Mook 1979, Nicholis and Prigogine 1977, Padmanabhan and Lapidus 1977, Pisman 1984, Raschman 1980, Ray 1980, Sheintuch and Luss 1985, Taylor 1980, Tyson 1973]. In particular the work by Minorsky [1962], and that of Nayfeh and Mook [1979] are considered classics in the analysis of nonlinear oscillators. The vast majority of the published material is concerned with the analysis of nonlinear systems and nonlinear oscillators; there is essentially no work in the synthesis of nonlinear oscillators. Today, one cannot state a priori what the 'best' nonlinear feedback algorithm is for any given system. This is due in part to the lack of mathematical tools for this synthesis: "Synthesis, that is, defining the best system if the inputs and desired outputs are specified, is more difficult than analysis. Only one mathematical synthesis problem in connection with nonlinear control systems has as yet been solved" [Graham and McRuer 1961]. For example, the work of the Brandeis group on the synthesis of chemical oscillators has shown some general procedures to follow if a bistable chemical system is known. But if the goal is discovering new bistable systems, there is no general way to "synthesize" them. Or, if the goal is to improve the yield or selectivity of a chemical reaction, there is no way to predict what kind of a chemical oscillation system (if any) might optimize these quantities. Given a set of nonlinear kinetic equations, there is no general way to produce or synthesize nonlinear oscillators. Some progress has been made with work on bifurcation theory, which

predicts when a system can become bistable and break into oscillation [Chang and Chen 1984, Lyberatos et al 1985a,b, McDermott (in press), Nicolis and Prigogine 1977, Pisman 1984, Ray and Jensen 1980, Sheintuch and Luss 1985]. But a general method of synthesizing nonlinear oscillators is not available.

It would be an oversimplification to try and summarize all of the results of nonlinear analysis and nonlinear control theory in a work of the present scope, so the generalization of the oscillogenic instrument will be restricted to the feedback algorthms described and used in Chapters 3, 5 and 6. This is not a severe restriction because this feedback algorithm has a great deal of potential. The oscillogenic feedback mechanism used (see Chapters 3 and 5):

    (1) has two output levels

    (2) may have one of two hysteresis elements

        a. a Schmitt trigger

        b. a "pre-action" trigger

The first step in generalizing is to enumerate possible inputs and outputs for chemical systems. Inputs are the means by which one can manipulate the behavior of a chemical system. Inputs can be the addition of energy or mass. Types of energy commonly encountered in chemistry are electromagnetic radiation (light), electromotive force (electricity), and thermal. The field of photochemistry concerns the effects of light on chemistry. In part, these include intramolecular effects from fluorescence to phosphorescence and intermolecular effects involving the formation of reactive species. The chemical effect is initiated by the absorption of light by the chemical species. As seen in the previous chapters, imposing a voltage on an electrode can cause chemical reactions

to take place. Changes in temperature from the application of heat, although perhaps a mundane example, can cause the rates of reactions to change and thus affect the chemistry of a system. Types of mass addition to a chemical system may be divided into two types: reagents and catalysts. Reagents are consumed by a reaction directly; while catalysts, which increase the rate of certain reactions, are not directly consumed by the reaction. However, the catalyst may be consumed or changed by other reactions. The immediate effect of a mass addition is to change the concentration of some component in the chemical system. Thus the effect on the chemistry may be modeled by a change in concentration.

The outputs of a chemical system are observable quantities which in some way reflect the state of the system. A useful distinction among these quantities is whether the measurement reflects a level or a flux. As has been seen, the current flowing in an electrochemical system is representative of the flux of ions reacting at the surface. Similarly, the emission of photons in the course of a chemical reaction is a measure of the flux or rate of reaction. The temperature of a reaction mixture is also indicative of the state of a reacting chemical system. Other measures indicate the level or amount of some species. For example, a concentration sensor such as an ion selective electrode might have an output voltage related to the concentration of some species. Possible outputs are as varied as all of the methods of analytical chemistry, but one can enumerate a few of them: the current in an electrochemical reaction, the emission of light, the absorbance of light (which indicates the concentration of some species), the temperature of an exo- or endo-thermic reaction, thermal conductivity, refractive index, density (or pressure), and others especially including any measure of chemical concentration.

One might make a large matrix representing the various combinations of these input/output combinations, and devise different oscillogenic methods by taking entries out of the table and applying them to appropriate chemical systems. Figure 8.1 on the following page shows just such a matrix of input/output pairs. The output types are given codes from 1 to 12 and the input types are given codes from A to F. The figure shows 72 classes of oscillogenic systems.

Outputs:

Matrix of Oscillogenic Systems	Energy:				Mass:			Modulation:				
Inputs:	Electromagnetic radiation emission	Temperature	Electrical potential	Pressure	Concentration of a species	Gravimetric	Electrical current	Electromagnetic radiation absorbance	Optical properties	Electrical conductivity	Thermal conductivity	Oscillogenic system
Energy: Electromagnetic radiation	A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11	A12
Heat	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	B12
Electrical Potential	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	C11	C12
Mass: Reactant	D1	D2	D3	D4	D5	D6	D7	D8	D9	D10	D11	D12
Catalyst	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10	E11	E12
Regulator	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12

Figure 8.1 A range of possible oscillogenic instruments

The inputs and outputs are meant to be broad classes and not specific mechanisms. Electromagnetic radiation (A) is light of any wavelength which interacts chemically with a system; that is, the light interacts photochemically in the sense of generating excited states or reactive species. Using heat (B) as an input does not specify how that heat is added; it may be added from electrical resistance heating, induction heating, opening a steam valve, etc. Electrical potential (C) can be in the traditional electrochemical sense, which forms the major part of this work, or it could also refer to situations more akin to electrophoresis. Reactant addition (D) refers to the transfer of mass to the chemical system; this addition may be as simple as the opening of a valve, or could be the generation of chemical species at an electrochemical electrode or any other coupling of reactions. The term, reactant, indicates that the species added is consumed by a reaction in the oscillogenic instrument. On the other hand, the addition of a catalyst (E) might be useful in some oscillogenic systems. Some types of catalysts might be adapted to addition and removal by a single mechanism (electrochemical?); if not, then some means of removing the catalyst will probably be necessary at some point in the oscillation to reverse the direction of the oscillating quantity. The turning on and off of a catalyst is the action refered to in the item called "regulator" (F). For example, the activity of *regulatory* or *allosteric* enzymes can be changed by a *modulator* or *effector* species. If this modulator species can be added and removed from the reactant mixture, then this could form the input side of an oscillogenic system.

The output classes are also quite broad. Emission of electromagnetic radiation (1) includes inter-molecular reactions which produce excited species which then decay (i.e. the chemiluminescent

reaction of O_3 and NO_2) and intra-molecular effects such as fluorescence and phosphorescence. Temperature (2) as an output in an oscillogenic system with exo- or endo-thermic reactions or in systems which one wants to measure quantities such as thermal conductivity or heat capacity. Using electrical potential (3) as an output can reflect the relative quantities of two different oxidation states of a redox couple [Field 1972b, Nagy 1986]. Pressure (4) can be used with either thermal excitation (as in non-dispersive infrared specrophotometers) or with gas phase reactions where the total number of molecules changes during the course of the reaction. The use of the concentration of a chemical species (5) as an output directly is dependent on sensors to convert that concentration into an electrical quantity: pH electrodes, specific ion electrodes (including chemfets), probes using flourescence techniques (optrodes), etc. Electrical current (6) in an electrochemical system has been demonstrated previously in this work. A gravimetric (7) device would measure the deposition of some species by measurements of weight.

The next series of outputs is labeled "modulation" techniques. These are measurement methods which depend on their own input/output pairs. A common technique is measuring a species' absorbance of electromagnetic radiation (8). For oscillogenic techniques, this light could be from infrared to visible to ultraviolet or it could be techniques using radiofrequencies such as NMR. Optical properties outputs (9) include refractive index, rotation of polarized light, and circular dichroism. While electrical conductivity (10) has been mentioned previously in this work as non-selective, if the input is a selective method (affects only one species or reaction), then a non-selective output such as this might be useful. The meaning of electrical conductivity should be construed to include such detectors as flame ionization and electron capture which are forms of

conductivity. Similarly, although thermal conductivity (11) is non-selective, it could be used with a selective input. Finally, these measurements have been called modulation, and oscillogenic instruments have been seen earlier in this work as frequency modulators (frequency is modulated by concentration). So, a second oscillogenic system (12) might be imbedded in the first oscillogenic system to form a *recursive* or *cascaded* oscillogenic instrument. The frequency of the first oscillogenic system could be a function for concentration, for example, and that concentration measure could be used via a second external feedback path to set the output level of a second output signal. This second output signal could indirectly affect the first input. There appears to be no limit to the numbe

Given that each cell in figure 8.1 represents a class of oscillogenic instrument, one can speculate on the usefulness of assorted oscillogenic systems. These may also be taken as suggestions for future work.

A1. The most obvious use for a light in/light out oscillogenic instrument is to measure fluorescence. One of the problems with the usual fluorometers is a sensitivity to light scattered from particles in the sample. The two differential levels of the oscillogenic feedback output as seen in this work, may allow a distinction between the fluorescence and the scattering, thus reducing the error.

A3 and A6. Photoelectrochemistry holds promise for use both as a synthetic route for chemical compounds as well as in energy conversion (photovoltaic cells). By using an oscillogenic instrument, light striking a part of an electrode would be switched and the electrical potential or current of that electrode measured. The relationship between the light and the electrical activity would result in a frequency in the oscillogenic instrument. This frequency could measure some aspect of the kinetics of

the electrode (perhaps detecting aging) or the concentration of the electrolyte. The frequency signal could then be used either as a monitor or perhaps in a control system to adjust some other operating parameter.

A4. In the currently employed non-dispersive infrared spectrophotometer, a chopper mirror is employed to switch a light beam from a sample to a reference beam. This beam then falls on a chamber containing a fixed amount of the chemical compound which one wishes to measure. The light is absorbed by the gas in the cell and subsequently converted into heat and thus pressure. A pressure sensor (usually a capacitive sensor) is employed and the measurement signal is the ratio of the pressures resulting from the sample to the reference beams. If the light were modulated instead by an oscillogenic system, the frequency output of the system could be related to the measurement signal, avoiding some of the electronic conversions and perhaps yielding a more precise measurement.

A8. There are photochemical bleaching reactions where exposure to light causes a chemical change resulting in the decrease of absorbance of light. The measurement of absorbance and returning this signal to the input of the chemical system via a modulation of the incident light could result in an oscillogenic system. It is interesting to note that the absorbance measurement could depend on the same light used for the input to the system; this commonality of input and output devices simplifies the design of the instrument.

A9. There is the possiblity that the products of a photochemical reaction might have different optical properties than the reactants. The measurement of these properties and feeding them back to the input could produce an oscillogenic instrument.

B1. There might be thermoluminescent materials which could

participate in an oscillogenic instrument.

B2. The effect of heat on the temperature is the definition of heat capacity. An oscillogenic heat capacity meter is possible.

C1. Electroluminescent materials are known. The two signals could be coupled electronically to produce an oscillogenic system.

C5. Similar to the electrochemical system discussed in chapters 4 to 6, an electrode could be used to generate a chemical species, and the concentration of that species (or one coupled to it by another reaction) could be measured to provide the oscillogenic feedback.

C8. Optically transparent thin layer electrodes [DeAngelis and Heineman 1976] are used to study electrochemical reactions which produce or consume a light absorbing species. These could be converted to oscillogenic instruments.

D1. Chemiluminescence reactions are known and used analytically. These reactions could be coupled to an oscillogenic feedback, resulting in a frequency output proportional to concentration or reaction kinetics, possibly more sensitive to the species being analysed and/or perhaps less sensitive to interferences in the medium.

D2. Exo- or endo-thermic reactions could be made a part of oscillogenic systems.

D3. If the reactant added is one of a redox couple, then the potential of a platinum electrode could be measured for oscillogenic feedback.

D4. Possible use of reactant addition with pressure feedback in gas flames or reactions to produce oscillations.

D5. Reactant in and concentration out is perhaps the most general statement of a chemical oscillogenic instrument.

D8. There are many colorimetric methods of analysis. Using a

flowing reagent with a varying flow rate and measuring the absorbance would produce an oscillogenic instrument capable of many analyses.

E5. Adding a catalyst and monitoring the concentration of a species forms the kernel of techniques that are known as kinetic methods of analysis. If the catalyst addition can be controlled, then these techniques can become oscillogenic techniques.

F5. Enzymes are the best known catalysts which can change activity as the result of a non-reacting molecule. The vast majority of biochemical reactions are enzymatic, and a common method of internally regulating cell metabolism is via these types of enzymes. Oscillogenic techniques could be applied in such cases if the concentration of the regulator molecules could be controlled. The increasingly important study of biochemical systems might be a promising area for oscillogenic instruments.

F8. Many biologically active molecules have an optical absorbance. Coupling this absorbance to the generation of an enzymatic modulator could produce an oscillogenic instrument.

Since the above list ended with biological systems, it is interesting to note a significant compound which could be coupled in an oscillogenic instrument of one type or another. Nicotinamide adenine dinucleotide (NAD^+) is an extremely important enzyme cofactor. It is electroactive (E_0' = -0.32 volt), light absorbent (260 and 340nm), and fluorescent (440nm). Any of these properties could be used to make an oscillogenic instrument capable of measuring NAD^+. While the measurement of NAD^+ itself is important in monitoring biochemical reactions, it also is used in a wide variety of biochemical assays. Table 8.2 shows a list of common assays which use enzymes to couple NAD^+ consumption to a reaction with a

Table 8.2 Diagnostic kits for clinical laboratories using NAD$^+$

SIGMA Chemical Company
St. Louis, MO 63178

adenosine 5'-triphosphate
alanineaminotransferase
aspartateaminotransferase
ammonia
amylase
carbon dioxide
creatine kinase
2,3-diphosphoglyceric acid
ethyl alcohol
galactose-1-phosphate uridlyl transferase
glucose
glucose-6-phosphate dehydrogenase
inorganic phosporus
isocitrate dehydragenase
lactate dehdrogenase
lactic acid
5' nucleotidase (5'-ND)
phosphohexose isomerase
pyrophosphate
sorbitol dehdrogenase
triglycerides
urea
uric acid

species of interest. This compound would make a good candidate for the next oscillogenic instrument.

In summary, this work has shown that by comparing physical and chemical oscillators, insights can be found which allow the design of a general chemical oscillator. This generalized oscillator consists of a chemical (or physical) system coupled with an external (electronic) feedback. The combination can be designed to oscillate, such that the output frequency is a measure of some chemical parameter. The term *oscillogenic* is suggested for this combination, because neither element oscillates by itself but the combination gives rise to oscillations. The theory was reduced to practice with an electrochemical oscillogenic instrument, which was demonstrated to give a frequency proportional to concentration. One of the tools developed for analysing the system and reported here was a series of FORTRAN programs for solving the electrochemical reaction-diffusion problem. A general method was shown for extracting multiple chemical measurements from a time varying signal, and this method is not restricted to oscillogenic instruments. Finally, a wide range of applications for the oscillogenic technique were suggested for future work.

Bibliography

Akima, Hiroshi, "A New method of interpolation and Smooth curve fitting based on local procedures", J ACM, 17(4), 589 (Oct 1970)

Alamgir, Mohamed et al, "A New Type of Bromate Oscillator: The Bromate-Iodide Reaction in a Stirred-Flow Reactor", JACS, 105, 2641 (1983)

Andricacos, P.C. and P.N. Ross, "Diffusion-Controlled Multisweep Cyclic Voltammetry I. Reversible Deposition on a Rotating Disk Electrode", J Electrochem Soc, 130(6), 1340 (Jun 1983)

Andricacos, P.C. and P.N. Ross, "Diffusion-Controlled Multisweep Cyclic Voltammetry II. Reversible Deposition on a Stationary Planar Electrode", J Electrochem Soc, 130(6), 1354 (Jun 1983)

Anson, Fred C., "Chronocoulometry", J Chem Ed, 60(4), 293 (Apr 1983)

Aris, Rutherford and Neal R. Amundson, "An analysis of chemical reactor stability and control - I", Chem Eng Sci, 7(3), 121 (1958)

Ascher, U. et al, "COLSYS -- A collocation code for Boundary-Value", Problems Lecture Notes in Computer Science, 76, Codes for B.V.P. in O.D.E., ed B Childs et al, 164 (1978)

Ascher, U. et al, "On spline basis selection for solving differential Equations", SIAM J Numer Anal, 20(1), 121 (Feb 1983)

Baccaro, G. P., et al, "An Experimental Study of Oscillating Reactors", AIChE J, 16(2), 249 (Mar 1970)

Bacon, J. Roger and J. N. Demas, "Phase-Plane and Guggenheim Methods for Treatment of Kinetic Data", Anal. Chem., 55(4), 653 (April 1983)

Baden, Niels and John Villadsen, "A family of Collocation based Methods for Parameter Estimation in Differential Equations", to appear in Chem Eng J.

Bailey, J.E., "Periodic Operation of Chemical Reactors: A Review", Chem Eng Commun., 1, 111 (1973)

Bailey, James E., "Periodic Phenomena", Chemical Reactor Theory: A Review, ed L. Lapidus and N. R. Amundson, Prentice Hall, (1977)

Balzani, Vincenzo, et al, "Electron Transfer Reactions Involving Light", J Chem Ed, 60(6), 447 (Jun 1983)

Bar-Eli, K., "Coupling of Chemical Oscillators", J Phys Chem, 88, 3616 (1984)

Bard, Allen J., "Chemical Modification of Electrodes", J Chem Ed, 60(4), 302 (Apr 1983)

Bauerlen, Hans, et al, "Distance-frequency transducer", (to Robert Bosch GmbH), U. S. Patent 4,392,383, (12 Jul 1983)

Bellman, R., Stability Theory of Differential Equations, Dover NY, (1969)

Belousov, B. P., "A Periodic Reaction and Its Mechanism", Sbornik Referatov po Radiatsionni Meditsine, Medgiz, Moscow, 143 (1959)

Berzins, Talivaldis, and Paul Delahay, "Kinetics of Fast Electrode Reactions", JACS, 77, 6448 (20 Dec 1955)

Bierman, G.J., "A comparison of Discrete linear filtering algorithms", IEEE Trans on Aerospace and Electronic Systems, AES-9(1), 28 (Jan 1973)

Bierman, Gerald J., "Measurement Updating using the U-D Factorization", Automatica, 12, 375 (1976)

Bierman, Gerald J., Factorization Methods for Discrete Sequential Estimation, Academic Press, (1977)

Bird, R. B., W. E. Stewart, & E. N. Lightfoot, Transport Phenomena, John Wiley & Sons, (1960)

Bittanti, Sergio, "Local Identifiability of Time-invariant linear systems by periodic test signals", Automatica, 18(2), 215 (1983)

Bockris, J. O'M., "Teaching the Double Layer", J Chem Ed, 60(4), 265 (Apr 1983)

Boissonade, J. and P. de Kepper, "Transitions from Bistability to Limit Cycle Oscillations. Theoretical Analysis and Experimental Evidence in an Open Chemical System", J Phys Chem, 84, 501 (1980)

Bond, A. M., Modern Polaragraphic Methods in Analytical Chemistry, Marcel Dekker, Inc. New York, 23 ()

Bos, Martin, "Resonent Frequency Measurements as an Alternative to Phase Selective AC-Polarograpy", 1983 ACS Annual meeting Washington D.C.

Botre, Claudio et al, "On the entropy production in oscillating chemical systems", Bioelectrochemistry and Bioenergetics, 8, 201 (1981)

Bray, W. C., "A Periodic Reaction in Homogeneous Solution and Its Relation to Catalysis", <u>JACS</u>, <u>43</u>, 1262 (1921)

Brisset, J. L., "Ocillating chemical systems", <u>Bull Union Physiciens</u>, <u>75</u>(629), 371-82 (1980)

Brown, N. L., U. S. Patent 3,549,989, (22 Dec 70)

Brown, Robert A. et al, "Computer-Aided Analysis of Nonlinear Problems in Transport Phenomena", <u>New approaches to Nonlinear Problems in Dynamics</u>, ed. Philip J. Holmes, SIAM Philadelphia, (1980)

Brown, Teri F. and Steven D. Brown, "Resolution of Overlapped Electrochemical Peaks with the Use of the Kalman Filter", <u>Anal Chem</u>, <u>53</u>, 1410 (1981)

Brown, Teri F., et al, "Estimation of Electrochemical Charge Transfer Parameters with the Kalman Filter", <u>Anal Chem</u>, <u>56</u>(8), 1214 (Jul 1984)

Bruice, Thomas C. and George J. Kasperek, "Oscillating reaction. Reaction of potassuim bromate, ceric sulfate and a dicarboxilic acid", <u>Inorg Chem</u>, <u>10</u>(2), 382-6 (1971)

Burger, M. and E. Bujdoso, "Oscillating Chemical Reactions as an Example of the Development of a Subfield of Science", <u>Oscillations and Traveling Waves in Chemical Systems</u>, ed. Richard J. Field and Maria Burger, John Wiley & Sons, 565 (1985)

Burger, M., and R. J. Field, "A New Chemical Oscillator Containing Neither Metal Nor Oxyhalogen Ions", <u>Nature</u>, <u>307</u>, 720 (1984)

Burger, M. et al, <u>Homogeneous Osc. Chem. Reactions: A Bibliography</u>, Dept of Chem, University of Montana, Missoula (1983)

Bush, S.F., "The measurement and prediction of sustained temperature oscillations in a chemical reactor", <u>Proc. Roy. Soc.</u>, <u>A 309</u>, 1 (1969)

Caillaud, J.B. and L. Padmanabhan, "An improved semi-implicit Runge-Kutta Method for Stiff Systems", <u>Chem Eng J</u>, <u>2</u>, 227 (1971)

Carlson, Neal A., "Fast Triangular Formulation of the Square Root Filter", <u>AIAA Journal</u>, <u>11</u>(9), 1259 (Sep 1973)

Carnahan, B. et al, <u>Applied Numerical Methods</u>, John Wiley & Sons, (1969)

Castellan, Gilbert W., <u>Physical Chemistry</u>, Addison-Wesley Publishing Company, Reading Mass. (1964)

Chang, Hsueh-Chia, and Liang-Heng Chen, "Bifurcation characteristics of nonlinear systems under conventional PID control", Chem Eng Sci, 39(7/8), 1127 (1984)

Childers, T.C. et al, "Polaragraphic analysis of precious metals in plating baths", American Laboratory, 37 (Oct 1984)

Clement, Karsten et al, "Fixed-Bed Reactor Kalman Filtering and Optimal Control-II Experimental Investigation of Discrete Time Case with Stochasitc Disturbances", Chem Eng Sci, 35, 1231 (1980)

Cordero, A. Osorio and D.Q. Mayne, "Deterministic convergence of a self-tuning regulator with variable forgetting factor", IEE Proc pt D, 128(1), 19 (Jan 1981)

Crowley, Michael F. and Richard J. Field, "Electrically Coupled Belousov-Zhabotinskii Oscillators. 1. Experiments and Simulations", J Phys Chem, 90(9), 1907 (1986)

D'Azzo, J. J., & C. H. Houpis, Linear Control System Analysis and Design, McGraw-Hill New York, 23 (1975)

Dateo, Christopher E. et al, "Bistability and Oscillations in the Autocatalytic Chlorite-Iodide Reaction in a Stirred-Flow Reactor", JACS, 104(2), 504 (1982)

Davis, Mark E., Numerical Methods and Modeling for Chemical Engineering, John Wiley & Sons, (1984)

de Boor, Carl, A Practical Guide to Splines, Springer-Verlag, N.Y., (1978)

De Kepper P. and J. Boissande, "From Bistability to Sustained Oscillation in Homogeneous Chemical Systems in Flow Reactor Mode", Oscillations and Traveling Waves in Chemical Systems, ed. Richard J. Field and Maria Burger, John Wiley & Sons, 223 (1985)

De Kepper, P. et al, "Systematic Design of Chemical Oscillators 3. Bistability in the Oxidation of Arsenite by Iodate in a Stirred Flow Reactor", JACS, 103, 6121 (1981b)

De Kepper, P. et al, Non-equilibrium Dynamics in Chemical Systems, ed. Vidal, C. and A. Pacault, Springer, Berlin, 44-49 (1984)

De Kepper, Patrick et al, "Systematic Design of Chemical Oscillators. 2. A Systematically Designed Homogeneous Oscillating Reaction: The Arsenite-Iodate-Chlorite System", JACS, 103(8), 2133 (1981a)

DeAngelis, Thomas P. and William R. Heineman, "An Electrochemical Experiment Using an Optically Transparent Thin Layer Electrode", J Chem Ed, 53(9), 595 (Sep 1976)

Degn, H. and J. Higgins, "Existence of homogeneous oscillating reactions",
 J. Phys Chem, 72(7), 2692 (1968)

Degn, Hans, "Effect of Bromine Derivatives of Malonic Acid on the
 Oscillating Reaction of Malonic Acid, Cerium Ions and Bromate",
 Nature, 589 (11 Feb 1967)

Degn, Hans, "Oscillating Chemical Reactions in Homogeneous Phase", J
 Chem Ed, 49(5), 302 (May 1972)

Delaney, Michael F., "Chemometrics", Anal. Chem., 56, 261R (1984)

Demas, J. N. and D. Diemente, "An Oscillating Chemical Reaction with a
 Luminescent Indicator", J Chem Ed, 50(4), 357 (May 1973)

Denbigh, K. G. and J. C. R. Turner, Chemical Reactor Theory, Cambridge
 University Press, 198 (1971)

Derusso, Paul M. et al, "Introducton to Stability Theory and Lyapunov's
 Second Method", State Variables for Engineers, John Wiley and Sons,
 (1965)

Didden, C.B.M., and H.N.J. Poulisse, "On the determination of the number of
 components from simulated spectra using Kalman filtering.",
 Analytical Letters, 13(A11), 921 (1980)

Dixon, A. G., "Solution of Packed-Bed Heat-Exchanger Models by
 Orthogonal Collocation using Piecewise Cubic Hermite Functions",
 MRC Technical Summary Report #2116, University of Wisconsin -
 Madison, (Sep 1980)

Douglas J.M., Process Dynamics and Control, Prentice-Hall, 392 (1972)

Douglas, James M., "Analytical estimates of the performance of chemical
 oscillators", Ind Eng Chem Fundam, 6(2), 265-76 (1967)

Edelson, David, "Mechanistic Details of the Belousov-Zhabotinsky
 Oscillations IV. Sensitivity Analysis", Int J of Chem Kin, 13, 1175
 (1981)

Edelson, David, "Computer Simulation in Chemical Kinetics", Science,
 214(4524), 981 (27 Nov 1981)

Edelson, David and Valerie M. Thomas, "Sensitivity Analysis of Oscillating
 Reactions. 1. The Period of the Oregonator", J Phys Chem, 85(11),
 1556 (1981)

Edwards, Pamela and Karen K. Haak, "A pulsed amperometric detector for
 Ion Chromatography", American Laboratory, 78 (Apr 1983)

Eisenbaumer, R. L., "Perchlorates in doping polymers can pose hazards",
 Chemical and Engineering News, 4 (24 Jun 1985)

Ellis, Christopher, and Edward Naylor, U. S. Patent 4,091,833, (30 May 1978)

Epstein, I.R., "Oscillations and chaos in chemical systems", Physica D Nonlinear Phenomena, 7D(1-3), 57 (1983a)

Epstein, I. R. et al, Nonlinear Phenomena in Chemical Systems, ed. Vidal, C. and A. Pacault, Springer, Berlin, 188-191 (1981)

Epstein, I. R. et al, "Oscillating Chemical Reactions", Scientific American, 248(3), 112 (March 1983b)

Epstein, Irving R., "Complex Dynamical Behavior in 'Simple' Chemical Systems", J. Phys. Chem., 88(2), 187-198 (1984a)

Epstein, Irving R., "The Search for New Chemical Oscillators", Chemical Instabilities, ed G. Nicolis and F. Baras, D. Reidel Publishing Company, Dordrecht, 3 (1984b)

Epstein, Irving R. and Miklos Orban, "Halogen-Based Oscillators in a Flow Reactor", Oscillations and Traveling Waves in Chemical Systems, ed. Richard J. Field and Maria Burger, John Wiley & Sons, 257 (1985)

Evans, Dennis H. et al, "Cyclic Voltammetry", J Chem Ed, 60(4), 290 (Apr 1983)

Eykhoff, Pieter, "Kalman-Bucy Filter", System Identification Parameter and State Estimation, John Wiley NY (1974)

Feldberg, S. W., "Digital Simulation", Electroanalytical Chemistry, ed A.J.Bard, Marcel-Dekker Inc, N.Y., 199 (1969)

Feldberg, Stephen W., "Optimization of explicit finite-difference simulation of electrochemical phenomena utilizing an Exponentially Expanded space grid.", J Electroanal Chem, 127, 1 (1981)

Field, R.J., "A reaction periodic in time and Space", J Chem Ed, 49(5), 308 (May 1972a)

Field, R. J., "Chemical Organization in Time and Space", American Scientist, 73, 142 (March-April 1985)

Field, Richard J., "Experimental and Mechanistic Characterization of Bromate-Ion-Driven Chemical Oscillations and Traveling Waves in Closed Systems", Oscillations and Traveling Waves in Chemical Systems, ed. Richard J. Field and Maria Burger, John Wiley & Sons, 55 (1985)

Field, Richard J. and Richard M. Noyes, "Mechanisms of Chemical Oscillators: Conceptual Basis", <u>Accounts of Chemical Research</u>, <u>10</u>(6), 214 (1977)

Field, Richard J. et al, "Oscillations in Chemical Systems. II. Through Analysis of Temporal Oscillation in the Bromate-Cerium-Malonic Acid System", <u>JACS</u>, <u>94</u>(25), 8649 (13 Dec 1972b)

Fife, Paul C., "Comments on a paper by Edelson and Thomas on Sensitivity Analysis of Oscillating Reactions", <u>J Phys Chem</u>, <u>85</u>(19), 2661 (1981)

Finklea, Harry O., "The Preparation of TiO2 Electrodes with Minimum Mott-Schottky Frequency Dispersion", <u>J Electrochemical Soc</u>, <u>129</u>(9), 2003 (Sep 1982)

Finklea, Harry O., "Photoelectrochemistry: Introductory Concepts", <u>J Chem Ed</u>, 60(4), 325 (Apr 1983)

Finlayson, B., <u>Nonlinear Analysis in Chemical Engineering</u>, McGraw-Hill, (1980)

Forsythe, George and Cleve B. Moler, <u>Computer Solutions of Linear Algebraic Systems</u>, Prentice-Hall, Englewood Cliffs, N.J., (1967)

Franck, U.F., "Feedback Kinetics in Physicochemical Oscillators", <u>Ber. Busenges Phys. Chem</u>, <u>84</u>, 334 (1980)

Frank, Ildiko K., and Kowalski, Bruce R., "Chemometrics", <u>Anal. Chem.</u>, <u>54</u>, 232R (1982)

Frank-Kamenetskii, David A., <u>Diffusion and Heat Transfer in Chemical Kinetics</u>, trans ed John P. Appleton, Plenum Press, 517 (1969)

Franklin, Gene F., and J. David Powell, <u>Digital Control of Dynamic Systems</u>, Addison-Wesley Publishing Co., (1980)

Fraser, A. B., "Frequency Encoding Closed Loop Circuit with Transducer", (to The John Hopkins University), U. S. Patent 4,408,169, (4 Oct 1983)

Frigato, Giovanni, "Device for measuring conductivity of a solution", (to Sandoz Ltd), U. S. Patent 4,160,946, (10 Jul 1979)

Froment, Gilbert F. and Kenneth B. Bischoff, <u>Chemical Reactor Analysis and Design</u>, John Wiley & Sons, New York, 116 (1979)

Fujita, I. and C.K. Chang, "Microcell for Cyclic voltammetry", <u>J Chem Ed</u>, <u>61</u>(10), 813 (Oct 1984)

Gear, C. William, <u>Numerical Initial Value Problems in O. D. E.'s</u>, Prentice-Hall, Englewood Cliffs, N.J., (1971)

Gear, Charles W., "Simultaneous Numerical Solution of Differential-Algebraic Equations", IEEE Trans on Circuit Theory, CT-18(1), 89 (Jan 1971)

Gilsinn, David E., "The method of averaging and domains of stability for integral manifolds", SIAM J Appl Math, 29(4), 628 (Dec 1975)

Gilsinn, David E., "A High Order Generalized method of Averaging", SIAM J Appl Math, 42(1), 113 (Feb 1982)

Golub, G., "Numerical Methods for solving Linear Least Squares Problems", Numerische Mathematik, 7, 206 (1965)

Goodall, David M., and David R. Roberts, "Energy Transfer between Dyes", J Chem Ed, 62(8), 711 (Aug 1985)

Graham, Dunstan and Duane McRuer, Analysis of Nonlinear Control Systems, Dover Publications Inc., New York, (1961)

Grassmann, P., "'Kipp' Oscillations", Physical Principles of Chemical Engineering, Pergamon Press, 793 (1971)

Griffiths, John F., "Thermokinetic Oscillations in Homogeneous Gas-Phase Oxidations", Oscillations and Traveling Waves in Chemical Systems, ed. Richard J. Field and Maria Burger, John Wiley & Sons, 529-564 (1985)

Grindheim, Earl A., "Parallel T impedance measurement circuit for use with variable impedance sensor", (to Rosemount Inc.), U. S. Patent 4,391,146, (5 Jul 1983)

Grossblatt, Robert, "Designing with the Schmitt Trigger", Radio-Electronics, 80 (Sep 1984)

Guajardo, Ciro, "Dewpoint monitor", (to Robbins Aviation, U. S. Patent 3,689,907, (5 Sep 1972)

Guggenheim, E. A., "On the determination of the velocity constant of a unimolecular reaction", Philosophical Magazine, 2(9) Seventh Series, 538 (Sep 1926)

Guilbault, George G., "Uses of the piezoelectric cyrstal detector in analytical chemstry", Ion-Sel. Electrode Rev., 2(1), 3-16 (1980)

Hale, Jack K. and Joseph P. LaSalle, "Differential Equation: Linearity vs Nonlinearity", SIAM Review, 5(2), 249 (July 1963)

Hansen, Knud Waede and Sten Bay Jørgensen, "Dynamic Modelling of a gas Phase Catalytic Fixed-bed reactor-1, Experimental Apparatus and Determination of Reaction Kinetics", Chem Eng Sci, 31, 579 (1976)

Hansen, Knud Waede and Sten Bay Jørgensen, "Dynamic Modelling of a gas Phase Catalytic Fixed-bed reactor-II, Results from Dynamic Experiments", Chem Eng Sci, 31, 587 (1976)

Harada, K et al, "Fluctuation near the Instability Point in the Linear Feedback Oscillator", J Phys Soc of Japan, 50(7), 2450 (July 1981)

Hearon, J.Z., Bull. Math. Biophys., 15, 121 (1953)

Heineman, William R., "Spectroelectrochemistry", J Chem Ed, 60(4), 305 (Apr 1983)

Hetrick, Robert E., "Transient mode oxygen sensor and method", (to Ford Motor Company), U. S. Patent 4,272,330, (1981a)

Hetrick, Robert E., "Oscillatory mode oxygen sensor and method", (to Ford Motor Company), U. S. Patent 4,272,331, (1981b)

Hetrick, Robert E. and William A. Fate, "Absolute Pressure Sensor", (to Ford Motor Company), U. S. Patent 4,272,329, (1981)

Hetrick, Robert E., and William A. Fate, "Absolute Pressure Sensor", (to Ford Motor Company), U. S. Patent 4,396,466, (2 Aug 1983)

Higgins, Joseph, "The Theory of Oscillating Reactions", Applied Kinetics, ACS, 143 (1967)

Hobey, W. D. et al, "The Guggenheim method in computerized kinecitc analysis.", American Laboratory, 18(3), 97 (March 1986)

Hofer, B. E., "A Comparison of Low frequency RC Oscillator Topologies", 64th Convention of the Audio Engineering Society, (2 Nov 1979)

Hosten, Lucien H., and Gilbert F. Froment, "Isomerization of n-Pentane", Ind. Eng. Chem. Process Des. Develop., 10(2), 280 (1971)

Hu, S.S. and W.E.Schiesser, "An adaptive grid method in the numerical method of lines", Adv in Comp. Methods for Parital Diff Eqns IV, ed. R. Vichnevetsky and R.S. Stepleman, 305 (1981)

Jørgensen, Sten Bay and Knud Waede Hansen, "Dynamic Modelling of a gas Phase Catalytic Fixed-bed reactor-III, Identification using binary pulse sequence test signals", Chem Eng Sci, 31, 473 (1976)

Jackson, W., U. S. Patent 2,886,776, (12 May 1959)

Jaffe, Arthur, "Order the Universe: The Role of Mathematics", SIAM Review, 26(4), 473 (Oct 1984)

Jensen, Hames H., "A New Type of Oscillating Reaction: Air Oxidation of Benzaldehyde", JACS, 105, 2639 (1983)

Jha, P. N. and R. K. Prasod, "Stability and phase plane analysis of 'Brusselator'", J Indian Chem Soc, 58(4), 377-81 (1981)

Jutan, Arthur et al, "Multivariable Computer Control of a Butane Hydrogenolysis Reactor: Part I. State Space Reactor Modeling", AIChE J, 23(5), 732 (Sep 1977)

Kalman, R. E., "Mathematical Description of Linear Dynamical Systems", J. Soc. Ind. Appl. Math., ser. A. Control, 1(2), (1963)

Kalman, R. E., and R. S. Bucy, "New results in linear filtering and prediction theory.", Transactions ASME, 80, 468 (1958)

Kelzer, Joel and Daniel Scherson, "A Theoretical Investigation of Electrode Oscillations", J Phys Chem, 84, 2025 (1980)

Kirk, D. E., Optimal Control Theory, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 109 (1970)

Kissinger, Peter T. and William R. Heineman, "Cyclic Voltammetry", J Chem Ed, 60(9), 702 (September 1983)

Knowles, Greg, An Introduction to Applied Optimal Control, vol 159 Mathematics in Science and Engineering, Academic Press (1981)

Kopell, N., "Time-periodic but spatially irregular solution to a model reaction-diffusion equation", Ann. N.Y. Acad of Sci, 357(nonlin dyn), 397-409 (1980)

Kowalski, Bruce R., "Chemometrics", Anal. Chem., 52, 112R (1980)

Krüger, F., "Oszillatorische Entladung polarisierter Zellen", Annalen der Physik, Folge 4 Band 21, 701 (1906)

Lancaster, Don, TTL Cookbook, Howard W. Sams & Co., Inc., 171 (1974)

Lancaster, Don, CMOS Cookbook, Howard W. Sams & Co., Inc., 88 (1977)

Larter, Raima, "Sensitivity analysis: A numerical tool for the study of parameter variations in oscillating reaction models", Chemical Instabilities, ed G. Nicolis and F. Baras, D. Reidel Publishing Company, Dordrecht, 59 (1984)

Lefelhocz, John F., "The Color Blind Traffic Light", J Chem Ed, 49(5), 312 (May 1972)

Lehninger, Albert L., Biochemistry, Worth Publishers, Inc. New York, 1970.

Levenspiel, Octave, The Chemical Reactor Omnibook, Oregon St Univ Bookstores, Inc. Corvallis, OR, (1979)

Levie, R de, "On the Electrochemical Oscillator", Electroanal Chemistry Interfacial Electrochem., 25, 257 (1970)

Li, Chia-yu et al, "Electrochemical and HPLC characterization of selected benzoquinoneimines", American Laboratory, 36 (Dec 1983)

Liaw, J.-S. and J. E. Bailey, "Regulation of Lumped Periodic Processes", AIChE J, 20(5), 924 (Sep 1974)

Ljung, Lennart and Torsten Söderström, Theory and Practice of Recursive Identification, M.I.T. Press, (1983)

Lotka, Alfred J., "Contribution to the Theory of Periodic Reactions", J Phys Chem, 14, 271 (1910)

Lotka, Alfred J., "Undamped Oscillations Derived from the Law of Mass Action", J Am Chem Soc, 42, 1595 (1920)

Lyberatos, G et al, "Steady-state multiplicity and the bifucation analysis via the Newton polyhedron approach", Chem Eng Sci, 39(4), 947 (1984)

Lyberatos, G. et al, "Versal Matrix Families Normal Forms and Higher order Bifurcations in dynamic chemical systems", Chem Eng Sci, 40(7), 1177-1189 (1985a)

Lyberatos, G. et al, "Bifurcation from the Potential Field Analog of Some Chemical Systems", Chemical Engineering Science, 40(9), 1679-1687 (1985b)

Mabbott, Gary A., "An Introduction to Cyclic Voltammetry", J Chem Ed, 60(9), 697 (Sep 1983)

Macdonald, Digby D., Transient Techniques in Electrochemistry, Plenum Press, N Y, (1977)

Magno, F., et al, "A comparison of some recently developed procedures for digital simulation in electroanalytical research", Anal Chim Acta, 147, 65 (1983)

Maloy, J.T., "Factors Affecting the shape of Current-Potential Curves", J Chem Ed, 60(4), 285 (Apr 1983)

Mankin, J. et al, "Transition between periodic and Chaotic states in a continuous stirred reactor", Chemical Raction Engineering Boston, 196, ed Wei J. and C. Georgakis, ACS Symp Ser, 145 (1982)

Martin, R.S. and J.H. Wilkinson, "Solution of Symmetric and Unsymmetric Band Equations and the Calculation of Eigenvectors of Band Matrices", Numerische Mathematik, 9, 279 (1967)

Maselko, Jerzy and Irving R. Epstein, "Dynamical Behavior of Coupled Chemical Oscillators: Chlorite-Thiosulfate-Iodide-Iodine", J Phys Chem, 88(22), 5324 (Oct 1984)

Matsubara, M. et al, "Relay Feedback Periodic Control of Plate Columns", Chem Eng Sci, 37(5), 753 (1982)

McBride, Lyle E. Jr., "Liquid level sensing system", (to Texas Instruments Incorporated), U. S. Patent 4,253,064, (24 Feb 1981)

McDermott, H. et al, "Experimental investigation of controller-induced bifurcation in a fixed-bed autothermal reactor", Chem Eng Sci, in press

Meyer-Ebrecht, D., "Fast Amplitude Control of a Harmonic Oscillator", Proc IEEE, 60, 736 (June 1972)

Meyerhoff, Mark E. and Paul M. Kovach, "An Ion-Selective Electrode/ Flow-Injection Analysis Experiment", J Chem Ed, 60(9), 766 (Sep 1983)

Minorsky, N., Nonlinear Oscillations, Van Nostrand, Princeton N.J., (1962)

Mittal, Archana et al, "A Novel Type of Chemical Oscillator", Int J of Chem Kin, 13, 321 (1981)

Morton, W. and M. G. Goodman, "Parametric Oscillation in Simple Catalytic Reaction Models", Trans I Chem E, 59, 253 (1981)

Muller, Stefan C. et al, "The Structure of the Core of the Spiral Wave in the Belousov-Zhabotinskii", Science, 230, 661 (8 Nov 1985)

Murdock, L. C., "Oscillator circuit for providing a conductivity ratio of sea water", U. S. Patent 3,906,353, (16 Sep 1975)

Nagy, Arpad, and Ludovit Treindl, "A novel, non-halogen-based chemical oscillator.", Nature, 320(6060), 344-345 (27 March 1986)

Nayfeh, A. H., D. T. Mook, Nonlinear Oscillations, John Wiley & Sons, (1979)

Nicholson, Richard S., "Theory and Application of Cyclic Voltammetry for Measurement of Electrode Reaction Kinetics", Analytical Chemistry, 1351 (Oct 1965)

Nicholson, Richard S. and Irving Shain, "Theory of Stationary Electrode Polarography", Analytical Chemistry, 36(4), 706 (Apr 1964)

Nicolis, & I. Prigogine, Self-Organization in Non-equilibrium Systems, John Wiley & Sons, 71 (1977)

Nitzan, A. et al, "Fluctuation and transistions at chemical instabilities: The analogy to phase transitions", J Chem Phys, 61(3), 1056 (1 Aug 1974)

Nowobilski, Peter and Christos G. Takoudis, "Reaction between Nitric Oxide and Ammonia on Polycrystalline Pt: A Mathematical Model of Rate Oscillations", Chemical Engineering Science, 40(9), 1751-1757 (1985)

Noyes, Richard M. et al, "Oscillations in Chemical Systems. I. Detailed Mechanism in a System Showing Temporal Oscillations", JACS, 94(4), 1394 (23 Feb 1972)

Oktem, Ulku G., Chemical Osc. and Flickering on Catalytic Wires, advisor Eli Ruckenstein, University of Delaware, PhD dissertation, (May 1974)

Oliver, B. M., "The effect of mu-circuit non-linearity on the amplitude stability of RC oscillators", HP Journal, 11(8-10), 1 (Apr-June 1960.)

Orban, M., et al, J Phys Chem, 86, 431 (1982)

Orban M. et al, "Minimal Bromate Oscillator: Bromate-Bromide-Catalyst", JACS, 104(9), 2657 (1982a)

Orban, M. et al, "Chlorite Oscillators: New Experimental Examples Tristability, and Preliminary Classification", JACS, 104, 5911 (1982b)

Orban, Miklos and Irving R. Epstein, "Bistability in the Oxidation of Iron(II) by Nitric Acid", JACS, 104, 5918 (3 Nov 1982)

Ortoleva, P.J. and J. Ross, J Chem Phys, 56, 287 (1972)

Ortoleva, P & John Ross, "A chemical Instability Mechanism for Asymmetric Cell Differentiation", Biophysical Chemistry, 1, 87 (1973)

Pacault, A, "Structure Chimiques Spatiales", J de Chim Physique, 79(10), 705 (1982)

Padmanabhan L., and L. Lapidus, "Control of Chemical Reactors", Chemical Reactor Theory: A Review, ed L. Lapidus and N. R. Amundson, Prentice Hall, 814 (1977)

Palanivel, A. and P. Riyazuddin, "Fabrication of an Inexpensive Ion-Selective Electrode", J Chem Ed, 61(10), 920 (Oct 1984)

Papsin, G. A. et al, "Bistability in the Iodate Oxidation of Arsenous Acid", J Phys Chem, 85, 2575 (1981)

Parry, E.P. and R.A. Osteryoung, "Evaluation of Analytical Pulse Polarography", Anal Chem, 37(13), 1634 (Dec 1965)

Perrin, Charles, <u>Mathematics for Chemists</u>, Wiley-Interscience 1970.

Pile, James E., "Moisture meter", (to Brown & Williamson Tobacco), U. S. Patent 4,228,393, (14 Oct 1980)

Pinkhusovich R. L. et al, "The application of sorption-frequency method and solid electrolyte cells for analysis of substance composition", <u>Meas Prog Sci Tech Proc</u>, <u>8</u>, IMEKO Congr Int Meas Confed, 451 (1979)

Pisman, L.M., "Dynamics of lumped chemically reacting systems near singular bifurcation points", <u>Chem Eng Sci</u>, <u>39</u>(4), 1063 (1984)

Poncet, Pierre et al, "Systems Electrochimiques Autooscillants", <u>J de Chim Physique</u>, <u>74</u>(4), 452 (1977)

Pons, B. Stanley and P. P. Schmidt, "Global Spline Collocation in the Simulation of Electrochemical Diffusion Equations", <u>Electrochimica Acta</u>, <u>25</u>, 987 (1980)

Pons, S., "Polynomial Approximation Techniques for Differential Equations in Electrochemical Problems", <u>Electroanalytical Chemistry, vol 13,</u> ed Allen J. Bard, Marcel Dekker Inc NY, 115 (1984)

Pons, Stanley et al, "Simulation of the Dropping Mercury Electrode by orthogonal collocation", <u>Electrochimica Acta</u>, <u>27</u>(12), 1711 (1982)

Poulisse, H.N.J., "Multicomponent-analysis computations based on Kalman filtering", <u>Analytica Chimica Acta, 112</u>, 361 (1979)

Poulisse, H.N.J., and P. Engelen, "The Kalman filter as an on-line drift compensator in multicomponent analysis determinations", <u>Analytical Letters</u>, <u>13</u>(A14), 1211 (1980)

Poulisse, Hennie N. J. and Rob T. P. Jansen, "Reduction of analytical variance by using a discrete-time data-weighting filter to estimate abrupt changes in batch-type processes", <u>Analytica Chim Acta</u>, <u>151</u>(2), 433 (1 Jul 1983)

Prausnitz, J. M., <u>Molecular Thermodynamics of Fluid-Phase Equilibria</u>, Prentice-Hall, Englewood Cliffs, N. J., (1969)

Prigogine, I. and G. Nicolis, <u>J Chem Phys</u>, <u>46</u>, 3542 (1967)

Prigogine, I et al, <u>Nature, 225</u>, 535 (1970)

Rainville, Earl D., <u>Elementary Differential Equations</u>, The Macmillan Company, New York, 431 (1964)

Raschman, P. et al, "Waves in Distributed Chemical Systems: Experiments and Computations", <u>New approaches to Nonlinear Problems in Dynamics</u>, ed. Philip J. Holmes, SIAM Philadelphia, (1980)

Rastogi, R.P. and R Shabd, "The Thermodynamics of Stability of Nonequilibrium Steady States", J Chem Ed, 60(7), 540 (July 1983)

Ray, W.H. and K.F. Jensen, "Bifurcation Phenomena in Stirred Tanks and Catalytic Reactors", New approaches to Nonlinear Problems in Dynamics, ed. Philip J. Holmes, SIAM Philadelphia, (1980)

Reckley, J. S. and K. Showalter, JACS, 103, 7012 (1981)

Reich, Herbert J., Functional Circuits and Oscillators, D. Van Nostrand Co., Inc. Princeton N. J., 350 (1961)

Rensing, L. and G. Cornelius, "Biological membranes as components of oscillatory Systems", Biol Rundsch, 18(4), 197-209 (1980)

Roelofs, Mark G., "Mechanism of an Oscillating Organic Reaction: Oxidation of Benzaldehyde with O2 Catalysed by Co(II)/Br(I)", JACS, 105, 6329 (1983)

Romagnoli, Jose A. and Rafiqul Gani, "Studies of Distributed Parameter Systems: Decoupling the State-Parameter Estimation Problem", Chem Eng Sci, 38(11), 1831-1843 (1983)

Rony, Peter R., Logic & Memory Experiments Using TTL Integrated Circuits, Book 2, Howard W. Sams & Co., Inc., 173 (1979)

Rovinsky, A. B. and A. M. Zhabotinsky, "Mechanism and Mathematical Model of the Oscillating Bromate-Ferroin-Bromomalonic Acid Reaction", J Phys Chem, 88(25), 6081 (6 Dec 1984)

Ruoff, Peter, "Kjemiske oscillasjoner", Kjemi, 28 (May 1982)

Ruoff, Peter and Bengt Schwitters, "Theoretical Study of Ag+ Induced Oscillations and Excitation in the Classical Homogeneous Belousov-Zhabotinsky Reaction Using the Oregonator Model", J Phys Chem, 88(25), 6424 (6 Dec 1984)

Russell, T. W. F., and M. M. Denn, Introduction to Chemical Engineering Analysis, John Wiley & Sons, Inc., New York, (1972)

Rutan, Sarah C., and Steven D. Brown, "Two-dimensional and Three-dimensional Fitting of Enzyme Kinetic Data with the Kalman Filter", Analytica Chimica Acta, 175, 1/1/1985 (219-229)

Rutan, Sarah C. and Steven D. Brown, "Pulsed Photoacoustic Spectroscopy and Spectral Deconvolution with the Kalman Filter for Determination of Metal Complexation Parameters", Anal Chem, 55(11), 1707 (Sep 1983)

Rutan, Sarah C., and Steven D. Brown, "Estimation of First-Order Kinetic Parameters by Using the Extended Kalman Filter", Analytica Chimica Acta, 167, 23-37 (1985)

Sørensen, Jan P. and Warren E. Stewart, "Collocation Analysis of Multicomponent Diffusion and reactions in porous catalysts", Chem Eng Sci, 37(7), 1103 (1982)

Sørensen, Jan P., et al, "Fixed-Bed Reactor Kalman Filtering and Optimal Control-I, Computational comparison of discrete vs continuous time formulation", Chem Eng Sci, 35, 1223 (1980)

Schoenberg, I. J., "Contributions to the problem of approximation of equidistant data by analytic functions", Quart. Appl. Math., 4, 45-99, 112-141 (1946)

Schwarz, W.M. and Irving Shain, "Investigation of First-Order Chemical Reactions Following Charge Transfer by a Step-Function Controlled Potential Method. The Benzidine Rearr.", J Phys Chem, 69(1), 30 (Jan 1965)

Seeber, Renato and Stefano Stefani, "Explicit Finite Dfference Method in Simulating Electrode Processes", Anal Chem, 53(7), 1011 (Jun 1981)

Seelig, Paul F., and Henry N. Blount, "Experimental Evaluation of Recursive Estimation Applied to Linear Sweep Anodic Stripping Voltammetry for Real Time Analysis", Anal Chem, 51(3), 327 (March 1979)

Seelig, Paul F. and Henry N. Blount, "Application of Recursive Estimation to Real Time Analysis of Trace Metal Analytes by Linear Sweep, Pulse, and Differential Pulse Anodic Stripping VA", Anal Chem, 51(8), 1129 (July 1979)

Sheintuch, Moshe and Dan Luss, "Dynamic Features of Two Ordinary Differential Equations with Widely Separated Time Scales", Chemical Engineering Science, 40(9), 1653-1664 (1985)

Sherrick, P. N., Manual of Chemical Oscillometry, E. H. Sargent, (1954)

Sin-rhu, Lin and Feng Qiang-sheng, "Determination of Chemical Reaction Rate Constants Preceding or Following Electron Transfer by mechanical Square Wave Polarography", Anal Chem, 54, 1362 (1982)

Smith, Donald E., "Thermodynamic and Kinetic Properties of the Electrochemical Cell", J Chem Ed, 60(4), 299 (Apr 1983)

Smith, J. M., Chemical Engineering Kinetics, McGraw-Hill Book Company, New York, (1981)

Spangler, Robert A., "Transfer function analysis of chemical kinetic systems", J. Theor. Biol, 16(3), 366-80 (1967)

Speiser, Bernd et al, "Electroanal. Investigations IV. Use of orthogonal collocation for the simulation of Quasireversible electrode process. under potential scan conditions", Electrochimica Acta, 27(9), 1171 (1982)

Spitler, M.T., "Dye Photooxidation at semiconductor electrodes: A corollary to spectral sensitization in Photography", J Chem Ed, 60(4), 330 (Apr 1983)

Stephen, Stuart J., "Methods and apparatus for measuring electrical conductivity", (to AHI Operations, Limited), U. S. Patent 4,309,660, (5 Jan 1982)

Steuer, Robert R., "Hematocrit measurements by electrical conductivity", U. S. Patent 3,922,598, (25 Nov 1975)

Strejc, V., "Least Square Parameter Estimation", Automatica, 16, 535 (1980)

Tamamushi, Reita, "An 'Electrochemical Oscillator'- an electrochemical system which generates undamped electric oscillation", J Electroanal Chem, 11, 65 (1966)

Tayor, James H., "A general limit cycle analysis method for multivariable systems", New approaches to Nonlinear Problems in Dynamics, ed. Philip J. Holmes, SIAM Philadelphia, (1980)

Technical Staff of the Analytic Sciences Corp., Applied Optimal Estimation, Arthur Gelb, M.I.T. Press, (1974)

Thijssen, P. C. et al, "State Estimation in Discrete Titrations with Kalman Filtering and Fixed-interval Smoothing", Analytica Chimica Acta, 170, 265-278 (1985)

Thompson, J. M. T., Instabilities and Catastrophes in Science & Eng, Wiley-Interscience Publication, (1982)

Tyson, John J., "Some further studies of nonlinear oscillations in chemical systems", J. Chem Phys., 58(9), 3919 (1 May 1973)

Tyson, John J., "A Quantitative Account of Oscillations, Bistability, and Traveling Waves in the Belousov-Zhabotinskii Reaction", Oscillations and Traveling Waves in Chemical Systems, ed. Richard J. Field and Maria Burger, John Wiley & Sons, 93 (1985)

Van Benschoten, James J., et al, "Cyclic Voltammetry Experiment", J Chem Ed, 60(9), 772 (Sep 1983)

Van der Pol, B., "On 'Relaxation-Oscillations'", <u>Philosophical Magazine and Journal of Science</u>, <u>Vol II</u> Seventh Series No.11, 978 (Nov 1926)

Vannerson, Eric and K. C. Smith, "Fast amplitude stabilization of an RC oscillator", <u>IEEE J of Solid State Circuts</u>, <u>SC-9</u>(4), 176 (Aug 1974)

Vannerson, Eric and K. C. Smith, "A low-distortion oscillator with fast amplitude stabilization", <u>Int J Electronics</u>, <u>39</u>(4), 465 (Oct 1975)

Varyu, M. E., "Potential hazard of perchlorate-doped polyacetylene", <u>Chemical and Engineering News</u>, 4 (8 Jul 1985)

Vidal, C. and A. Noyau, "Some differences between thermokinetic and Chemical oscillating reactions", <u>JACS</u>, <u>102</u>, 6666 (1980)

Villadsen, J. V. and W. E. Stewart, "Solution of Boundry-value Problems by orthogonal collocation", <u>Chem Eng Sci</u>, <u>22</u>, 1483 (1967)

Walker, Jearl, "Chemical systems that oscillate", <u>Chemtech</u>, 320 (May 1980)

Wang, Joseph, "On-Line sensors for trace metal", <u>American Laboratory</u>, 14 (Jul 1983)

Whiting, Larry F. and Peter W. Carr, "A Simple, Fast numerical method for the solution of a wide variety of electrochemical diffusion problems", <u>J Electroanal Chem</u>, <u>81</u>, 1 (1977)

Williams, D. C. and J. M. Calo, "The dynamic simulation of oscillatory cheimcal reaction systems", <u>Proc Summer comput. simul conf</u>, 289-92 (1980)

Winfree, A. T., "Rotating Chemical Reactions", <u>Scientfic American</u>, <u>230</u>(6), 82 (June 1974)

Winfree, A. T., "The Prehistory of the Belousov-Zhabotinsky Oscillator", <u>J Chem Ed</u>, <u>61</u>(8), 661 (Aug 1984)

Winfree, Arthur T., "Waves of Chemical Activity", <u>Science</u>, <u>175</u>, 634 (11 Feb 1972)

Winfree, Arthur T., "Scroll-Shaped Waves of Chemical Activity in Three Dimensions", <u>Science</u>, <u>181</u>, 937 (7 Sep 1973)

Wojtowicz, J., "Oscillatory Behavior in Electrochemical Systems", <u>Modern Aspects of Electrochemistry</u>, <u>8</u>, ed. J. O'M. Bockris & B.E. Conway, Plenum Press NY, 47 (1972)

Wool, Kin-Chu, <u>Electronics</u>, 178 (9 Oct 1980)

Yoshikawa, Kenichi and Yasuhiro Matsubara, "Chemoreception by an Excitable Liquid Membrane: Characteristic effect of Alcohols on the Frequency of Electrical Oscillation", <u>JACS</u>, <u>106</u>, 4423 (1984)

Zhabotinskii, A. M., "Periodic Processes of the Oxidation of Malonic Acid in Solution (Study of the Kinetics of Belousov's Reaction)", Biofizika, 9, 306 (1964a)

Zhabotinskii, A. M., "Periodic Liquid Phase Oxidation Reactions", Dokl. Aka. Nauk SSSR, 157, 392 (1964b)

Zimmerman, Henry J. & Samuel J. Mason, "Chap 9. Waveform Generation", Electronic Circuit Theory, John Wiley & Sons New York, 430 (1959)

Appendix

Listing 1. REDISTribute the breakpoints subroutine (FORtran)

```fortran
      SUBROUTINE REDIST(BP,NB,NBD,ALPHA,NA,NAD,S,NS,NSD,NEWBP)
C A routine to estimate an improved set of breakpoints based on
C   a given set of breakpoints and a vector of coefficients
C     NB = the number of breakpoints
C     BP(NB) = vector of breakpoints, dimensioned as BP(NBD)
C     ALPHA(2*NB) = vector of coefficients, dimensioned as ALPHA(NAD)
C     S(NB-2) = vector of accumulated error, dimensioned as S(NSD)
C                  by Russell's (COLSYS) method
C     NEWBP(N) = vector of improved breakpoints
      INTEGER I,J,NB,NA,NS,NBD,NAD,NSD
      REAL*8 BP(NBD),ALPHA(NAD),S(NSD),NEWBP(NBD)
     &        ,HLESS,HPLUS,DELS,NEXTS
C Calculate third derivative (P3) differences
C     as an approximation to the fourth derivative (P4)
      DO 10 I=1,NS
         HLESS=BP(I+1)-BP(I)
         HPLUS=BP(I+2)-BP(I+1)
         S(I) = -12.D0/HLESS**3 * ALPHA(2*I-1)
     &          -6.D0/HLESS**2 * ALPHA(2*I)
     &          +(12.D0/HLESS**3+12.D0/HPLUS**3) * ALPHA(2*I+1)
     &          +(-6.D0/HLESS**2+6.D0/HPLUS**2) * ALPHA(2*I+2)
     &          -12.D0/HPLUS**3 * ALPHA(2*I+3)
     &          +6.D0/HPLUS**2 * ALPHA(2*I+4)
         S(I)=S(I)*2.D0/(HLESS+HPLUS)
 10      CONTINUE
C Let the accumulated error be proportional to the integral of the
C    square root of the absolute value of the fourth derivative
      HLESS=BP(2)-BP(1)
      HPLUS=BP(3)-BP(2)
      S(1) = DSQRT(DABS(S(1)))*(HLESS+HPLUS/2)
      HLESS=HPLUS
      DO 30 I=2,NS-1
         HPLUS=BP(I+2)-BP(I+1)
         S(I)=S(I-1)+ DSQRT(DABS(S(I)))*(HLESS+HPLUS)/2
         HLESS=HPLUS
 30      CONTINUE
      HPLUS=BP(NB)-BP(NB-1)
      S(NS) = S(NS-1) + DSQRT(DABS(S(NS)))*(HLESS/2+HPLUS)
C Normalize the accumulated error to 1.0 total
      DO 40 I=1,NS
         S(I)=S(I)/S(NS)
 40      CONTINUE
C Setup a delta accumulated error for each new break point
      DELS=S(NS)/(NB-1)
      NEXTS=DELS
C Create a vector of redistributed break points such that the
C    error is equidistributed.
      NEWBP(1)=BP(1)
      J=2
      I=0
 50   I=I+1
 60   IF (S(I).LT.NEXTS) GOTO 50
      IF (1.EQ.I) NEWBP(J)=NEXTS/S(1)*((BP(3)+BP(2))/2-BP(1))+BP(1)
      IF (1.LT.I.AND.I.LT.NS)
```

```
&    NEWBP(J)=(NEXTS-S(I-1))/(S(I)-S(I-1))
&         *(BP(I+2)-BP(I))/2+(BP(I)+BP(I+1))/2
 IF (I.EQ.NS)
&    NEWBP(J)=( BP(NB)-(BP(NB-1)+BP(NB-2))/2 ) / (1-S(NS-1))
&                *(NEXTS-S(NS-1))
&                    +(BP(NB-1)+BP(NB-2))/2
 J=J+1
 NEXTS=NEXTS+DELS
 IF (NEXTS.LT.1) GOTO 60
 NEWBP(NB)=BP(NB)
 RETURN
 END
```

Listing 2. ReDiStribution of breakpoints TeST program (FORtran)

```
$STORAGE:2
$NOFLOATCALLS
C  Test of the breakpoint REDISTribution subroutine
C      NB = the number of breakpoints
C      BP(NB) = vector of breakpoints, dimensioned as BP(NBD)
C      ALPHA(2*NB) = vector of coefficients, dimensioned as ALPHA(NAD)
C      S(NB-2) = vector of accumulated error, dimensioned as S(NSD)
C                by Russell's (COLSYS) method
C      NEWBP(N) = vector of improved breakpoints
       INTEGER I,J,NB,NA,NS,NBD,NAD,NSD
       REAL*8 BP(4),ALPHA(8),S(2),NEWBP(4)
       NB=4
       NBD=4
       NA=8
       NAD=8
       NS=2
       NSD=2
       BP(1)=0.5D0
       BP(2)=1.0D0
       BP(3)=1.5D0
       BP(4)=2.0D0
       WRITE(6,5)(BP(I),I=1,NB)
  5    FORMAT(1X,'Old breakpoints: ',4(G11.4,2X))
C
C  Coefficients are those approximating a hyperbola:  y = 1/x
C
       DO 10 I=1,NB
          ALPHA(2*I-1)=1.D0/BP(I)
          ALPHA(2*I)=-1.D0/BP(I)**2
 10       CONTINUE
       CALL REDIST(BP,NB,NBD,ALPHA,NA,NAD,S,NS,NSD,NEWBP)
       WRITE(6,6)(NEWBP(I),I=1,NB)
  6    FORMAT(1X,'New breakpoints:  ',4(G11.4,2X))
       STOP
       END
```

Listing 3. ReDiStribution of breakpoints TeST program output (DATa)

```
Old breakpoints:    .5000       1.000       1.500       2.000
New breakpoints:    .5000       .8265       1.153       2.000
```

Listing 4. FILL matricies with basis function values subroutine (FORtran)

```fortran
      SUBROUTINE FILL(NBP,BP,A,C)
      INTEGER NBP,ID,ROW,I,J
      REAL*8 BP(9),A(17,5),C(17,6),H,AVEC(4),V(2,4)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  On input:
C
C     BP      REAL*8, a vector of breakpoints
C     NBP     INTEGER*2, the number of breakpoints used
C
C  On output:
C
C     A       REAL*8, matrix of the basis function values at the Gauss
C                  points and the first boundary point
C     C       REAL*8, matrix of the second derivatives of the basis
C                  functions at the Gauss points, and 1st boundary point
C
C  Uses:
C
C     AVEC    REAL*8, vector for values returned by INTWTS
C     H       REAL*8, the local step size
C     I       INTEGER*2, a counter
C     ID      INTEGER*2, the derivative value desired (0th, 1st, 2nd)
C     J       INTEGER*2, a counter
C     ROW     INTEGER*2, a counter for which pair of rows to fill
C     V       REAL*8, vector for values returned by BASWTS
C
C  Uses subroutines:
C
C     BASWTS
C     INTWTS
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  fill incomplete rows at beginning
C    A and V:
      H=BP(2)-BP(1)
      ID=0
      CALL BASWTS(ID,H,V)
      A(1,1)=0.D0
      A(1,2)=0.D0
      A(1,3)=V(1,2)
      A(1,4)=V(1,3)
      A(1,5)=V(1,4)
      A(2,1)=0.D0
      A(2,2)=V(2,2)
      A(2,3)=V(2,3)
      A(2,4)=V(2,4)
      A(2,5)=0.D0
C  set up the first rows of C
      ID=2
      CALL INTWTS(ID,BP(1),BP(1),BP(2),AVEC)
      C(1,1)=0.D0
      C(1,2)=0.D0
```

```
      C(1,3)=0.5D0*AVEC(1)
      C(1,4)=0.D0
      C(1,5)=0.5D0*AVEC(3)
      C(1,6)=0.D0
      CALL BASWTS(ID,H,V)
      C(2,1)=0.D0
      C(2,2)=V(1,1)
      C(2,3)=V(1,2)
      C(2,4)=V(1,3)
      C(2,5)=V(1,4)
      C(2,6)=0.D0
      C(3,1)=V(2,1)
      C(3,2)=V(2,2)
      C(3,3)=V(2,3)
      C(3,4)=V(2,4)
      C(3,5)=0.D0
      C(3,6)=0.D0
C  fill complete rows
      DO 40 I=2,NBP-1
C   V(1,1) = phi0(g1)
C   V(2,1) = phi0(g2)
          ROW=2*I-1
          ID=0
          H=BP(I+1)-BP(I)
          CALL BASWTS(ID,H,V)
          A(ROW,1)=0.D0
          A(ROW+1,5)=0.D0
          DO 50 J=1,4
              A(ROW,J+1)=V(1,J)
              A(ROW+1,J)=V(2,J)
   50         CONTINUE
          ID=2
          CALL BASWTS(ID,H,V)
          C(ROW+1,1)=0.D0
          C(ROW+1,6)=0.D0
          C(ROW+2,5)=0.D0
          C(ROW+2,6)=0.D0
          DO 51 J=1,4
              C(ROW+1,J+1)=V(1,J)
              C(ROW+2,J)=V(2,J)
   51         CONTINUE
   40     CONTINUE
C  fill incomplete rows at end
      H=BP(NBP)-BP(NBP-1)
      ID=0
      CALL BASWTS(ID,H,V)
      ROW=2*NBP-3
      A(ROW,1)=0.D0
      A(ROW,2)=V(1,1)
      A(ROW,3)=V(1,2)
      A(ROW,4)=V(1,4)
      A(ROW,5)=0.D0
      A(ROW+1,1)=V(2,1)
      A(ROW+1,2)=V(2,2)
      A(ROW+1,3)=V(2,4)
      A(ROW+1,4)=0.D0
      A(ROW+1,5)=0.D0
```

```
ID=2
CALL BASWTS(ID,H,V)
C(ROW+1,1)=0.D0
C(ROW+1,2)=V(1,1)
C(ROW+1,3)=V(1,2)
C(ROW+1,4)=V(1,4)
C(ROW+1,5)=0.D0
C(ROW+1,6)=0.D0
C(ROW+2,1)=V(2,1)
C(ROW+2,2)=V(2,2)
C(ROW+2,3)=V(2,4)
C(ROW+2,4)=0.D0
C(ROW+2,5)=0.D0
C(ROW+2,6)=0.D0
RETURN
END
```

Listing 5. BASis function WeighTS subroutine (FORtran)

```fortran
      SUBROUTINE BASWTS(ID,H,V)
      REAL*8 S3,V(2,4),H,J
      INTEGER ID
1234567890123456789012345678901234567890123456789012345678901234567890
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   BASis cubic spline WeighTS
C   This rouine returns the values, first, or second derivatives of
C     the Hermite cubic basis functions at the collocation points.
C
C   On input:
C
C      ID = indicator for which value is desired: 0 for value
C                                                 1 for first derivative
C                                                 2 for second derivative
C
C      H = mesh size, X    - X
C                      i+1    i
C
C   On output:
C
C   V = matrix of weights; where v    =  for jth function  1 = v1
C                                 ij         ith            2 = s1
C                                            collocation    3 = v2
C                                            point          4 = s2
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      S3 = DSQRT(3.D0)
      IF (ID-1) 1,2,3
1     V(1,1)=0.5D0+2.0D0/(3.D0*S3)
      V(1,3)=1.D0-V(1,1)
      V(1,2)=(1.D0+1.D0/S3)/12.D0*H
      V(1,4)=(1.D0/S3-1.D0)/12.D0*H
      GO TO 4
2     V(1,1)=-1.D0/H
      V(1,3)=-V(1,1)
      V(1,2)=0.5D0/S3
      V(1,4)=-V(1,2)
      GO TO 4
3     V(1,1)=-2.D0*S3/(H**2)
      V(1,3)=-V(1,1)
      V(1,2)=-(1.D0+S3)/H
      V(1,4)=(1.D0-S3)/H
4     J=(-1.D0)**(ID+1)
      V(2,2)=J*V(1,4)
      V(2,4)=J*V(1,2)
      V(2,3)=-J*V(1,1)
      V(2,1)=-J*V(1,3)
      RETURN
      END
```

Listing 6. mesh INTerior WeighTS subroutine (FORtran)

```fortran
      SUBROUTINE INTWTS(ID,X,XI,XJ,VECT)
      INTEGER ID
      REAL*8 H,X,XI,XJ,VECT(4),Z,Y
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   INTerior WeighTS of Hermite cubic splines
C   This routine calculates the spline function values at any point between
C     or including two mesh points.
C
C   On input:
C
C      ID = value of derivative desired:   function value, 0
C                                          first derivative, 1
C                                          second derivative, 2
C
C      X = point where spline values are desired
C      XI = lower mesh point
C      XJ = higher mesh point
C
C
C   On ouput:
C
C      VECT(1) = value function j (Vj)
C      VECT(2) = slope function j (Sj)
C      VECT(3) = value function j+1 (Vj+1)
C      VECT(4) = slope function j+1 (Sj+1)
C
C   Uses:
C
C      H = mesh size
C      Z = dimensionless distance from XI
C      Y = dimensionless distance from XJ
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      H=XJ-XI
      Z=(X-XI)/H
      Y=(X-XJ)/H
      IF (ID-1) 1,2,3
    1    VECT(3)=(3.D0-2.D0*Z)*Z**2
         VECT(1)=1.D0-VECT(3)
         VECT(4)=H*Y*Z**2
         VECT(2)=H*Z*Y**2
         RETURN
    2    VECT(1)=6.D0*(Z**2-Z)/H
         VECT(3)=-VECT(1)
         VECT(4)=Z**2+2.D0*Z*Y
         VECT(2)=2.D0*Z*Y+Y**2
         RETURN
    3    VECT(1)=6.D0*(2.D0*Z-1.D0)/H**2
         VECT(3)=-VECT(1)
         VECT(2)=(2.D0*Z+4.D0*Y)/H
         VECT(4)=(4.D0*Z+2.D0*Y)/H
         RETURN
      END
```

Listing 7. SPLine TeST program (FORtran)

```
C  A test of the SPLine WeighTS subroutines
C      The subroutine BASWTS (BASis function WeighTS) calculates
C          the values of the spline functions (or their derivatives)
C          at the Gauss points.
C      The subroutine INTWTS (INTernal WeighTS) calculates the values
C          of the spline functions (or their derivatives) at any point
C          between two breakpoints.
C      The difference between the two programs is that if only the
C          values at the Gauss points are needed (collocation) then
C          BASWTS will be much faster and more precise than INTWTS
C
       INTEGER ID
       REAL*8 H,V(2,4),XI,XJ,X,VECT(4),XLO,XHI,DX
       OPEN(1,FILE='CON ',STATUS='NEW',ACCESS='SEQUENTIAL')
       OPEN(2,FILE='CON ',STATUS='OLD',ACCESS='SEQUENTIAL')
  90   CONTINUE
       WRITE(1,10)
  10   FORMAT(1X,'Input first and second breakpoint:  ')
       READ(2,12)XI,XJ
  12   FORMAT(F10.5,F10.5)
       WRITE(1,30)
  30   FORMAT(1X,'Enter id no. for:  function (0), '
      &           ,'1st derivative (1), or 2nd derivative (2):  ')
       READ(2,32)ID
  32   FORMAT(I1)
       H=XJ-XI
       CALL BASWTS(ID,H,V)
       WRITE(6,40)
  40   FORMAT(1X,'                    X          '
      &           ,'value fcn 1, slope fcn 1, value fcn 2, slope fcn 2')
       WRITE(6,50)XI+H/2.D0*(1.D0-1.D0/DSQRT(3.D0))
      &           ,V(1,1),V(1,2),V(1,3),V(1,4)
  50   FORMAT(1X,'Gauss point 1: ',G11.4,4(2X,G11.4))
       WRITE(6,60)XI+H/2.D0*(1.D0+1.D0/DSQRT(3.D0))
      &           ,V(2,1),V(2,2),V(2,3),V(2,4)
  60   FORMAT(1X,'Gauss point 2: ',G11.4,4(2X,G11.4))
       DX=(XJ-XI)/20.D0
       X=XI
       WRITE(6,65)
  65   FORMAT(1X)
       DO 70 I=1,21
          CALL INTWTS(ID,X,XI,XJ,VECT)
 110      WRITE(6,80)X,(VECT(J),J=1,4)
  80      FORMAT(1X,'                ',G11.4,4(2X,G11.4))
          X=X+DX
  70      CONTINUE
       GOTO 90
       END
```

Listing 8. SPLine TeST program output (DATa)
(bold face is for user response)

C>SPLTST

Input first and second breakpoint: **0.D0,1.D0**

Enter id no. for : function (0), 1st derivative (1), or 2nd derivative (2): **0**
File name missing or blank – Please enter name
UNIT 6? **CON**

	X	value fcn 1,	slope fcn 1,	value fcn 2,	slope fcn 2
Gauss point 1:	.2113	.8849	.1314	.1151	-.3522E-01
Gauss point 2:	.7887	.1151	.3522E-01	.8849	-.1314
	.0000	1.000	.0000	.0000	.0000
	.5000E-01	.9928	.4513E-01	.7250E-02	-.2375E-02
	.1000	.9720	.8100E-01	.2800E-01	-.9000E-02
	.1500	.9393	.1084	.6075E-01	-.1913E-01
	.2000	.8960	.1280	.1040	-.3200E-01
	.2500	.8438	.1406	.1563	-.4688E-01
	.3000	.7840	.1470	.2160	-.6300E-01
	.3500	.7183	.1479	.2818	-.7962E-01
	.4000	.6480	.1440	.3520	-.9600E-01
	.4500	.5748	.1361	.4252	-.1114
	.5000	.5000	.1250	.5000	-.1250
	.5500	.4253	.1114	.5747	-.1361
	.6000	.3520	.9600E-01	.6480	-.1440
	.6500	.2817	.7962E-01	.7183	-.1479
	.7000	.2160	.6300E-01	.7840	-.1470
	.7500	.1562	.4687E-01	.8438	-.1406
	.8000	.1040	.3200E-01	.8960	-.1280
	.8500	.6075E-01	.1912E-01	.9393	-.1084
	.9000	.2800E-01	.9000E-02	.9720	-.8100E-01
	.9500	.7250E-02	.2375E-02	.9928	-.4512E-01
	1.000	.0000	.4930E-31	1.000	.2220E-15

Input first and second breakpoint: **0.D0,1.D0**

Enter id no. for : function (0), 1st derivative (1), or 2nd derivative (2): **1**

	X	value fcn 1,	slope fcn 1,	value fcn 2,	slope fcn 2
Gauss point 1:	.2113	-1.000	.2887	1.000	-.2887
Gauss point 2:	.7887	-1.000	-.2887	1.000	.2887
	.0000	.0000	1.000	.0000	.0000
	.5000E-01	-.2850	.8075	.2850	-.9250E-01
	.1000	-.5400	.6300	.5400	-.1700
	.1500	-.7650	.4675	.7650	-.2325
	.2000	-.9600	.3200	.9600	-.2800
	.2500	-1.125	.1875	1.125	-.3125
	.3000	-1.260	.7000E-01	1.260	-.3300
	.3500	-1.365	-.3250E-01	1.365	-.3325
	.4000	-1.440	-.1200	1.440	-.3200
	.4500	-1.485	-.1925	1.485	-.2925
	.5000	-1.500	-.2500	1.500	-.2500
	.5500	-1.485	-.2925	1.485	-.1925
	.6000	-1.440	-.3200	1.440	-.1200

```
        .6500        -1.365        -.3325         1.365        -.3250E-01
        .7000        -1.260        -.3300         1.260         .7000E-01
        .7500        -1.125        -.3125         1.125         .1875
        .8000        -.9600        -.2800         .9600         .3200
        .8500        -.7650        -.2325         .7650         .4675
        .9000        -.5400        -.1700         .5400         .6300
        .9500        -.2850        -.9250E-01     .2850         .8075
       1.000         .1332E-14     .4441E-15     -.1332E-14    1.000
Input first and second breakpoint:  0.D0,1.D0

Enter id no. for :  function (0), 1st derivative (1), or 2nd derivative (2):  2
              X       value fcn 1, slope fcn 1, value fcn 2, slope fcn 2
Gauss point 1:  .2113        -3.464        -2.732         3.464        -.7321
Gauss point 2:  .7887         3.464         .7321        -3.464         2.732

        .0000        -6.000        -4.000         6.000        -2.000
        .5000E-01    -5.400        -3.700         5.400        -1.700
        .1000        -4.800        -3.400         4.800        -1.400
        .1500        -4.200        -3.100         4.200        -1.100
        .2000        -3.600        -2.800         3.600        -.8000
        .2500        -3.000        -2.500         3.000        -.5000
        .3000        -2.400        -2.200         2.400        -.2000
        .3500        -1.800        -1.900         1.800         .1000E+00
        .4000        -1.200        -1.600         1.200         .4000
        .4500        -.6000        -1.300         .6000         .7000
        .5000        -.6661E-15    -1.000         .6661E-15     1.000
        .5500         .6000        -.7000        -.6000         1.300
        .6000        1.200         -.4000        -1.200         1.600
        .6500        1.800         -.1000E+00    -1.800         1.900
        .7000        2.400          .2000        -2.400         2.200
        .7500        3.000          .5000        -3.000         2.500
        .8000        3.600          .8000        -3.600         2.800
        .8500        4.200         1.100         -4.200         3.100
        .9000        4.800         1.400         -4.800         3.400
        .9500        5.400         1.700         -5.400         3.700
       1.000         6.000         2.000         -6.000         4.000
Input first and second breakpoint:  0.D0,2.D0

Enter id no. for :  function (0), 1st derivative (1), or 2nd derivative (2):  0
              X       value fcn 1, slope fcn 1, value fcn 2, slope fcn 2
Gauss point 1:  .4226         .8849         .2629         .1151        -.7044E-01
Gauss point 2:  1.577         .1151         .7044E-01     .8849        -.2629

        .0000        1.000          .0000         .0000         .0000
        .1000         .9928         .9025E-01     .7250E-02    -.4750E-02
        .2000         .9720         .1620         .2800E-01    -.1800E-01
        .3000         .9393         .2168         .6075E-01    -.3825E-01
        .4000         .8960         .2560         .1040        -.6400E-01
        .5000         .8438         .2813         .1563        -.9375E-01
        .6000         .7840         .2940         .2160        -.1260
        .7000         .7183         .2958         .2817        -.1593
        .8000         .6480         .2880         .3520        -.1920
        .9000         .5748         .2723         .4252        -.2228
       1.000          .5000         .2500         .5000        -.2500
       1.100          .4253         .2228         .5747        -.2723
       1.200          .3520         .1920         .6480        -.2880
       1.300          .2818         .1593         .7183        -.2958
```

1.400	.2160	.1260	.7840	-.2940
1.500	.1562	.9375E-01	.8438	-.2813
1.600	.1040	.6400E-01	.8960	-.2560
1.700	.6075E-01	.3825E-01	.9393	-.2167
1.800	.2800E-01	.1800E-01	.9720	-.1620
1.900	.7250E-02	.4750E-02	.9928	-.9025E-01
2.000	.0000	.9861E-31	1.000	.4441E-15

Input first and second breakpoint: 0.D0,2.D0

Enter id no. for : function (0), 1st derivative (1), or 2nd derivative (2): 1

	X	value fcn 1,	slope fcn 1,	value fcn 2,	slope fcn 2
Gauss point 1:	.4226	-.5000	.2887	.5000	-.2887
Gauss point 2:	1.577	-.5000	-.2887	.5000	.2887

.0000	.0000	1.000	.0000	.0000
.1000	-.1425	.8075	.1425	-.9250E-01
.2000	-.2700	.6300	.2700	-.1700
.3000	-.3825	.4675	.3825	-.2325
.4000	-.4800	.3200	.4800	-.2800
.5000	-.5625	.1875	.5625	-.3125
.6000	-.6300	.7000E-01	.6300	-.3300
.7000	-.6825	-.3250E-01	.6825	-.3325
.8000	-.7200	-.1200	.7200	-.3200
.9000	-.7425	-.1925	.7425	-.2925
1.000	-.7500	-.2500	.7500	-.2500
1.100	-.7425	-.2925	.7425	-.1925
1.200	-.7200	-.3200	.7200	-.1200
1.300	-.6825	-.3325	.6825	-.3250E-01
1.400	-.6300	-.3300	.6300	.7000E-01
1.500	-.5625	-.3125	.5625	.1875
1.600	-.4800	-.2800	.4800	.3200
1.700	-.3825	-.2325	.3825	.4675
1.800	-.2700	-.1700	.2700	.6300
1.900	-.1425	-.9250E-01	.1425	.8075
2.000	.6661E-15	.4441E-15	-.6661E-15	1.000

Input first and second breakpoint: 0.D0,2.D0

Enter id no. for : function (0), 1st derivative (1), or 2nd derivative (2): 2

	X	value fcn 1,	slope fcn 1,	value fcn 2,	slope fcn 2
Gauss point 1:	.4226	-.8660	-1.366	.8660	-.3660
Gauss point 2:	1.577	.8660	.3660	-.8660	1.366

.0000	-1.500	-2.000	1.500	-1.000
.1000	-1.350	-1.850	1.350	-.8500
.2000	-1.200	-1.700	1.200	-.7000
.3000	-1.050	-1.550	1.050	-.5500
.4000	-.9000	-1.400	.9000	-.4000
.5000	-.7500	-1.250	.7500	-.2500
.6000	-.6000	-1.100	.6000	-.1000E+00
.7000	-.4500	-.9500	.4500	.5000E-01
.8000	-.3000	-.8000	.3000	.2000
.9000	-.1500	-.6500	.1500	.3500
1.000	-.1665E-15	-.5000	.1665E-15	.5000
1.100	.1500	-.3500	-.1500	.6500
1.200	.3000	-.2000	-.3000	.8000
1.300	.4500	-.5000E-01	-.4500	.9500
1.400	.6000	.1000	-.6000	1.100

1.500	.7500	.2500	-.7500	1.250
1.600	.9000	.4000	-.9000	1.400
1.700	1.050	.5500	-1.050	1.550
1.800	1.200	.7000	-1.200	1.700
1.900	1.350	.8500	-1.350	1.850
2.000	1.500	1.000	-1.500	2.000

Input first and second breakpoint: ^C

Listing 9. BANded matrix MULtiply subroutine (FORtran)

```fortran
      SUBROUTINE BANMUL(ND,N,MTD,MT,M1,A,B,M,MD,C)
      INTEGER ND,N,MTD,MT,M1,SUB,I,I1,I2,J,J1
      REAL*8 A(ND,MTD),B(MD),C(ND)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   BANded MULtiply
C   This routine multiplies a matrix, A, by a vector, B, and returns the
C      result in vector C.
C
C   On entry:
C
C      A = the banded matrix in compressed form
C            the full matrix is N by M
C      N = the full row order of A
C      M = the full column order of A
C      ND = the actual row dimension of A in the calling routine
C      MT = the total bandwidth of A
C      MTD = the actual column dimension of A in the calling routine
C      M1 = the number of subdiagonals in A
C      B = the multiplier vector
C      MD = the actual dimension of vector B in the calling routine
C
C   On exit:
C
C      C = the product vector
C
C   Uses:
C
C      SUB = an integer subscript
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C   beginning with the incomplete rows
      DO 10 I=1,M1
         C(I)=0.D0
C   J1 is the first good column of the incomplete row
         J1=M1+2-I
         DO 20 J=J1,MT
            SUB=J-M1-1+I
            C(I)=A(I,J)*B(SUB)+C(I)
20       CONTINUE
10    CONTINUE
C   continuing with the complete rows
C      I1 is the first complete row, I2 is the last complete row
      I1=M1+1
      I2=M-MT+M1+1
      DO 30 I=I1,I2
         C(I)=0.D0
         DO 40 J=1,MT
            SUB=I-M1-1+J
            C(I)=A(I,J)*B(SUB)+C(I)
40       CONTINUE
30    CONTINUE
C   and finishing with the ending incomplete rows
C      I1 is the first incomplete row at the end of the matrix
```

```
      I1=I2+1
      DO 70 I=I1,N
        C(I)=0.D0
C  J2 is the last good column of the incomplete row at the end
        J2=M1+1+M-I
        DO 80 J=1,J2
          SUB=J-M1-1+I
          C(I)=A(I,J)*B(SUB)+C(I)
80        CONTINUE
70    CONTINUE
      I1=SUB+1
      RETURN
      END
```

Listing 10. BaNded matrix MULtiply TeST program (FORtran)

```
$STORAGE:2
$NOFLOATCALLS
      INTEGER ND,N,MTD,MT,M1,SUB,I,I1,I2,J,J1
      REAL*8 A(5,3),B(5),C(5)
      N=5
      ND=5
      M=5
      MT=3
      MTD=3
      MD=5
      M1=1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  This routine multiplies a matrix, A, by a vector, B, and returns the
C     result in vector C.
C
C  Matrix A:
C
C
C
C          | 2  3  0  0  0 |                     | 0  2  3 |
C          |               |                     |         |
C          | 1  2  3  0  0 |                     | 1  2  3 |
C          |               |                     |         |
C          | 0  1  2  3  0 |     is stored as    | 1  2  3 |
C          |               |                     |         |
C          | 0  0  1  2  3 |                     | 1  2  3 |
C          |               |                     |         |
C          | 0  0  0  1  2 |                     | 1  2  0 |
C          -               -                     -         -
C
C
C  Vector B:                        Vector C should be:
C
C
C          -   -                            -    -
C          |   |                            |    |
C          | 1 |                            |  8 |
C          |   |                            |    |
C          | 2 |                            | 14 |
C          |   |                            |    |
C          | 3 |                            | 20 |
C          |   |                            |    |
C          | 4 |                            | 26 |
C          |   |                            |    |
C          | 5 |                            | 14 |
C          |   |                            |    |
C          -   -                            -    -
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      A(1,1)=0.D0
      A(1,2)=2.D0
      A(1,3)=3.D0
      DO 10 I=2,4
         A(I,1)=1.D0
         A(I,2)=2.D0
```

```
        A(I,3)=3.D0
10      CONTINUE
      A(5,1)=1.D0
      A(5,2)=2.D0
      A(5,3)=0.D0
      B(1)=1.D0
      B(2)=2.D0
      B(3)=3.D0
      B(4)=4.D0
      B(5)=5.D0
      CALL BANMUL(ND,N,MTD,MT,M1,A,B,M,MD,C)
      DO 20 I=1,5
        WRITE(6,30)(A(I,J),J=1,3),B(I),C(I)
30      FORMAT(1X,F4.0,F4.0,F4.0,2X,F4.0,2X,F4.0)
20      CONTINUE
      STOP
      END
```

Listing 11. BaNded matrix MULtiply TeST program output (DATa)

```
0.  2.  3.    1.    8.
1.  2.  3.    2.   14.
1.  2.  3.    3.   20.
1.  2.  3.    4.   26.
1.  2.  0.    5.   14.
```

Listing 12. BANded matrix DEcomposiTion subroutine (FORtran)

```fortran
      SUBROUTINE BANDET(ND,N,MTD,MT,M1D,M1,A,XM,IP,IV,IFAIL)
      INTEGER SUB,ND,N,MTD,MT,M1D,M1,IP(ND),IV,IFAIL
      REAL*8 A(ND,MTD),XM(ND,M1D),X
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  BANded matrix DEcomposiTion
C  This is a routine to do the LU decomposition of a banded matrix.
C
C  Based on:  A. G. Dixon "Solution of Packed-Bed Heat-Exchanger Models by
C          Orthogonal(sic) Collocation Using Piecewise Cubic Hermite
C          Functions" Mathematics Research Center Technical Summary
C          Report #2116, University of Wisconsin-Madison, September 1980
C           and the algorithims 'bandet 1' and 'bansol 1' from R. S. Martin
C          and J. H. Wilkinson "Solution of Symmetric and Unsymmetric Band
C          Equations and the Calculation of Eigenvectors of Band Matrices"
C          Numerische Mathematik, vol 9, 279-301 (1967).
C
C  On entry:
C
C     ND = actual dimension declared for matrices:  IP, A, & XM
C     MTD = actual dimension declared for matrix:  A
C     M1D = actual dimension declared for matrix:  XM
C     A = the banded matrix in compressed form
C     N = order of banded matrix
C     M1 = number of subdiagonal elements in banded matrix
C     MT = total bandwidth of banded matrix
C
C  On exit:
C
C     A = the upper triangular matrix (U)
C     XM = the lower triangular matrix (L)
C     IP = a vector of pivot information
C     IFAIL = an error flag (0: no problem, 1:   value of K in IV)
C     IV = error value, row of A where problem was found
C
C  Uses:
C
C     X = a temporary scalar
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      IFAIL = 0
      L=M1
      DO 30 I=1,M1
        IJ=M1+2-I
        DO 25 J=IJ,MT
          SUB=J-L
25        A(I,SUB)=A(I,J)
        L=L-1
        JL=MT-L
        DO 30 J=JL,MT
          A(I,J)=0.D0
30        CONTINUE
      L=M1
      DO 100 K=1,N
```

```
        X=A(K,1)
        I=K
        IF(L.LT.N)L=L+1
        IF(K+1.GT.L)GOTO 40
            IK=K+1
        DO 35 J=IK,L
            IF (DABS(A(J,1)).LE.DABS(X))GOTO 35
                X=A(J,1)
            I=J
35          CONTINUE
40      IP(K)=I
    IF (X.NE.0.D0)GOTO 50
        IFAIL=1
        IV=K
        GO TO 101
50  IF(I.EQ.K)GOTO 70
        DO 65 J=1,MT
            X=A(K,J)
            A(K,J)=A(I,J)
65          A(I,J)=X
70          IF(K+1.GT.L)GO TO 100
                IK=K+1
                DO 90 I=IK,L
                    SUB=I-K
                    XM(K,SUB)=A(I,1)/A(K,1)
                    X=XM(K,SUB)
                    DO 80 J=2,MT
                        SUB=J-1
80                      A(I,SUB)=A(I,J)-X*A(K,J)
90                  A(I,MT)=0.D0
100             CONTINUE
101     CONTINUE
    RETURN
    END
```

Listing 13. BANded matrix SOLver subroutine (FORtran)

```fortran
      SUBROUTINE BANSOL(ND,N,MTD,MT,M1D,M1,A,XM,IP,B)
      INTEGER SUB,ND,N,MTD,MT,M1D,M1,IP(ND)
      REAL*8 A(ND,MTD),XM(ND,M1D),X,B(ND)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   BANded linear system SOLver
C   This routine solves the system:  L U u = b   for a banded system.
C                                     = = -   -
C     On entry:
C     ND = actual dimension declared for matrices:  IP, A, & XM
C     MTD = actual dimension declared for matrix:  A
C     M1D = actual dimension declared for matrix:  XM
C     A = the upper triangular matrix (U)
C     XM = the lower triangular matrix (L)
C     IP = a vector of information from BANDET
C     B = the right hand side vector
C
C     On exit:
C     B = the solution vector
C
C     Uses:
C     X = a temporary scalar
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      L=M1
      DO 40 K=1,N
         I=IP(K)
         IF (I.EQ.K) GOTO 10
            X=B(K)
            B(K)=B(I)
            B(I)=X
10       IF (L.LT.N) L=L+1
         IK=K+1
         DO 20 I=IK,L
            SUB=I-K
            X=XM(K,SUB)
20          B(I)=B(I)-X*B(K)
40    CONTINUE
      L=1
      DO 70 I1=1,N
         I=N+1-I1
         X=B(I)
         IW=I-1
         IF(L.EQ.1)GO TO 65
            DO 60 K=2,L
               SUB=IW+K
60             X=X-A(I,K)*B(SUB)
65       B(I)=X/A(I,1)
         IF (L.LT.MT)L=L+1
70    CONTINUE
      RETURN
      END
```

Listing 14. BaNded MATrix solver TeST program (FORtran)

```
$STORAGE:2
$NOFLOATCALLS
      INTEGER ND,N,MTD,MT,M1,M1D,IP(5)
     &        ,SUB,IV,IFAIL
      REAL*8 A(5,3),XM(5,1),X,B(5),C(5)
      N=5
      ND=5
      M=5
      MT=3
      MTD=3
      MD=5
      M1=1
      M1D=1
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   Matrix A:
C
C             _               _             _           _
C            |                 |           |             |
C            |  2  3  0  0  0  |           |  0  2  3  |
C            |                 |           |           |
C            |  1  2  3  0  0  |           |  1  2  3  |
C            |                 |           |           |
C            |  0  1  2  3  0  |  is stored as |  1  2  3  |
C            |                 |           |           |
C            |  0  0  1  2  3  |           |  1  2  3  |
C            |                 |           |           |
C            |  0  0  0  1  2  |           |  1  2  0  |
C            |                 |           |           |
C             _               _             _           _
C
C
C   Vector C:                    Vector B should be:
C
C             _     _                     _     _
C            |       |                   |       |
C            |   8   |                   |   1   |
C            |       |                   |       |
C            |  14   |                   |   2   |
C            |       |                   |       |
C            |  20   |                   |   3   |
C            |       |                   |       |
C            |  26   |                   |   4   |
C            |       |                   |       |
C            |  14   |                   |   5   |
C            |       |                   |       |
C             _     _                     _     _
C
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      A(1,1)=0.D0
      A(1,2)=2.D0
      A(1,3)=3.D0
      DO 10 I=2,4
         A(I,1)=1.D0
         A(I,2)=2.D0
```

```
         A(I,3)=3.D0
10       CONTINUE
      A(5,1)=1.D0
      A(5,2)=2.D0
      A(5,3)=0.D0
      B(1)=8.D0
      B(2)=14.D0
      B(3)=20.D0
      B(4)=26.D0
      B(5)=14.D0
      WRITE(6,21)
21    FORMAT(1X,'A:')
      DO 26 I=1,5
         WRITE(6,22)A(I,1),A(I,2),A(I,3)
22       FORMAT(1X,3(G11.4,2X))
26       CONTINUE
      WRITE(6,23)
23    FORMAT(1X,'C:')
      DO 24 I=1,5
         WRITE(6,25)B(I)
25       FORMAT(1X,G11.4)
24       CONTINUE
      CALL BANDET(ND,N,MTD,MT,M1D,M1,A,XM,IP,IV,IFAIL)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C  This is a routine to do the LU decomposition of a banded matrix.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      CALL BANSOL(ND,N,MTD,MT,M1D,M1,A,XM,IP,B)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  This routine solves the system:  L U u = b  for a banded system.
C                                    = = -   -
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      WRITE(6,32)
32    FORMAT(1X,'B:')
      DO 20 I=1,5
         WRITE(6,30)B(I)
30       FORMAT(1X,G11.4)
20       CONTINUE
      STOP
      END
```

Listing 15. BaNded MATrix solver TeST program output (DATa)

```
A:
   .0000      2.000      3.000
  1.000      2.000      3.000
  1.000      2.000      3.000
  1.000      2.000      3.000
  1.000      2.000       .0000
C:
  8.000
 14.00
 20.00
 25.00
 14.00
B:
 1.000
 2.000
 3.000
 4.000
 5.000
```

Listing 16. RUnge-KUtta-FEhlberg initial value solver subroutine (FORtran)

```
      SUBROUTINE RUKUFE(N,ND,X,DELX,H,Y4,Y5,TOL,DYDX
     &                 ,WORK,INDEX,DEBUG,YPGVN)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  This subroutine integrates the differential equation one step
C  using the Runge-Kutta-Fehlberg method.
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      EXTERNAL DYDX
      LOGICAL DEBUG,DONE,YPGVN
      INTEGER N,ND,INDEX,STPNUM,STPTOT
      REAL*8 X,DELX,Y4(ND),Y5(ND),H,TOL(ND),WORK(ND,7)
     +,XTEMP,ERROR,ETEMP
     +,C1,C2,C3,C4,C5,C6,C7,C8,C9,C10
     +,C11,C12,C13,C14,C15,C16,C17,C18,C19,C20,C21
      COMMON /Z0Z0Z0/C1,C2,C3,C4,C5,C6,C7,C8,C9,C10
     +              ,C11,C12,C13,C14,C15,C16,C17,C18,C19,C20,C21
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  N = the number of dependent variables
C  ND = the actual dimension of the array in the calling routine
C  X = the value of the independant variable at the start of the step
C        on return it has the value X+DELX
C  DELX = the increment value for the independent variable (step size)
C  H = locally used step size
C         (note that DELX/H = 2**m, where m is an integer)
C  Y4 = the value of the 4th order estimate at the end of the step
C  Y5 = the value of the 5th order estimate at the end of the step
C  TOL = vector of desired tolerances (relative)
C  DYDX = subroutine to calculate the rate of change of Y with respect to X
C         it must have arguments (n,nd,X,Y,Y') where Yi'=f(X,Y1,Y2...Yn)
C  XTEMP = x value within step
C  WORK = vector for scratch pad, must be dimensioned to 7*N
C  INDEX = flag and error code, on first call INDEX must be 1
C               on return, if INDEX = 0  then all was well
C               but if INDEX = -1  then there are more than
C               100 sub-steps in DELX and the problem may be too stiff.
C  DEBUG = a logical variable specifing extended print output for debugging
C  YPGVN = a logical variable indicating if on entry dY/dX is supplied
C               as vector WORK(I,1), (YPrimeGiVeN)
C  C = a vector of constants used repeatedly,
C               stored in labeled COMMON:  Z0Z0Z0
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  These constants only need to be calculated once!
C
      IF (INDEX.LT.1) GO TO 5
        C1 = 25.D0/216.D0
        C2 = 1408.D0/2565.D0
        C3 = 2197.D0/4104.D0
        C4 = -1.D0/5.D0
        C5 = 16.D0/135.D0
        C6 = 6656.D0/12825.D0
```

```
        C7 = 28561.D0/56430.D0
        C8 = -9.D0/50.D0
        C9 = 2.D0/55.D0
        C10 = 3.D0/32.D0
        C11 = 9.D0/32.D0
        C12 = 1932.D0/2197.D0
        C13 = -7200.D0/2197.D0
        C14 = 7296.D0/2197.D0
        C15 = 439.D0/216.D0
        C16 = 3680.D0/513.D0
        C17 = -845.D0/4104.D0
        C18 = -8.D0/27.D0
        C19 = -3544.D0/2565.D0
        C20 = 1859.D0/4104.D0
        C21 = -11.D0/40.D0
        IF (H.EQ.0.D0.OR.H.GT.DELX)H=DELX
        INDEX = 0
   5    CONTINUE
      IF(H.GT.DELX)H=DELX
      DONE=.FALSE.
      STPNUM = 1
      STPTOT = IFIX(SNGL(DELX/H+0.001D0))
      IF(DEBUG)WRITE(1,106)STPTOT
  106 FORMAT(1X,'SUB-STEP TOTAL = ',I4)
      IF(YPGVN)GOTO 150
    8 CONTINUE
C  if there are more than 500 steps then there is trouble with convergence
      IF (STPTOT.LT.500)GOTO 160
          INDEX=-1
          RETURN
C
C   1/4 step
C
  160 CALL DYDX(N,ND,X,Y5,WORK)
  150 DO 10 I=1,N
          WORK(I,2)=H*WORK(I,1)
   10     CONTINUE
C
  170 XTEMP = X+H/4.D0
      DO 19 I=1,N
          Y4(I)=Y5(I)+WORK(I,2)/4.D0
   19     CONTINUE
      CALL DYDX(N,ND,XTEMP,Y4,WORK)
      DO 20 I=1,N
          WORK(I,3)=H*WORK(I,1)
   20     CONTINUE
C
C   3/8 step
C
      XTEMP = X+0.375D0*H
      DO 29 I=1,N
          Y4(I)=Y5(I)+C10*WORK(I,2)+C11*WORK(I,3)
   29     CONTINUE
      CALL DYDX(N,ND,XTEMP,Y4,WORK)
      DO 30 I=1,N
          WORK(I,4)=H*WORK(I,1)
   30     CONTINUE
```

```
C
C   12/13 step
C
      XTEMP = X+12.D0*H/13.D0
      DO 39 I=1,N
         Y4(I)=Y5(I)+C12*WORK(I,2)+C13*WORK(I,3)
     +                    +C14*WORK(I,4)
  39     CONTINUE
      CALL DYDX(N,ND,XTEMP,Y4,WORK)
      DO 40 I=1,N
         WORK(I,5)=H*WORK(I,1)
  40     CONTINUE
C
C   full step, first approximation
C
      XTEMP = X+H
      DO 49 I=1,N
         Y4(I)=Y5(I)+C15*WORK(I,2)-8.D0*WORK(I,3)
     +                    +C16*WORK(I,4)
     +                    +C17*WORK(I,5)
  49     CONTINUE
      CALL DYDX(N,ND,XTEMP,Y4,WORK)
      DO 50 I=1,N
         WORK(I,6)=H*WORK(I,1)
  50     CONTINUE
C
C   1/2 step
C
      XTEMP = X+H/2.D0
      DO 59 I=1,N
         Y4(I)=Y5(I)+C18*WORK(I,2)+2.D0*WORK(I,3)
     +                    +C19*WORK(I,4)
     +                    +C20*WORK(I,5)
     +                    +C21*WORK(I,6)
  59     CONTINUE
      CALL DYDX(N,ND,XTEMP,Y4,WORK)
      DO 60 I=1,N
         WORK(I,7)=H*WORK(I,1)
  60     CONTINUE
C
C   full step, fourth order approximation
C
      DO 70 I=1,N
         Y4(I)=Y5(I)+C1*WORK(I,2)+C2*WORK(I,4)
     +                  +C3*WORK(I,5)+C4*WORK(I,6)
  70     CONTINUE
C
C   full step, fifth order approximation
C
      DO 80 I=1,N
         WORK(I,3)=Y5(I)+C5*WORK(I,2)+C6*WORK(I,4)
     +                    +C7*WORK(I,5)+C8*WORK(I,6)+C9*WORK(I,7)
         IF(TOL(I).LE.0.D0)GOTO 80
            WORK(I,1)=DABS(WORK(I,3)-Y4(I))
  80     CONTINUE
      DONE=.TRUE.
C
```

```
C  check to see if tolerance was met, and adjust step size
C
      DO 85 I=1,N
         IF(TOL(I).LE.0.D0)GO TO 85
         IF(WORK(I,1).GT.TOL(I)) DONE=.FALSE.
  85     CONTINUE
         IF(DONE)GOTO 120
         IF(.NOT.DEBUG) GO TO 202
            WRITE(1,108)STPNUM,STPTOT,X
 108        FORMAT(1X,'STEP NUMBER ',I3,' OUT OF ',I3
     +            ' X=',G11.4,/1X,'Y4=',11X,'Y5=')
            DO 203 I=1,N
               WRITE(1,109)Y4(I),WORK(I,3)
 203           CONTINUE
 109           FORMAT(1X,2(G11.4,2X))
            DO 215 I=1,N
               IF(TOL(I).LE.0.D0)GOTO 215
               WRITE(1,210)I,WORK(I,1),TOL(I)
 215           CONTINUE
 202        H=H/2.D0
               STPNUM=STPNUM*2-1
               STPTOT=STPTOT*2
               IF(STPNUM.NE.1)GOTO 8
               DO 216 I=1,N
                  WORK(I,2)=WORK(I,2)/2.D0
 216              CONTINUE
               GO TO 170
 120 X = X+H
      DO 100 I=1,N
         Y5(I)=WORK(I,3)
 100     CONTINUE
      IF(.NOT.DEBUG)GO TO 200
         WRITE(1,108)STPNUM,STPTOT,X
         DO 201 I=1,N
            WRITE(1,109)Y4(I),Y5(I)
 201        CONTINUE
 200     CONTINUE
      DO 205 I=1,N
         IF(TOL(I).LE.0.D0)GOTO 205
         IF(DEBUG)WRITE(1,210)I,WORK(I,1),TOL(I)
 210        FORMAT(1X,'element='I2,' error=',G11.4,5X
     &               ,'tolerance=',G11.4)
         IF (.NOT.(      (WORK(I,1).LE.TOL(I)/20.D0)
     +               .AND.(   MOD(STPNUM,2).EQ.0
     +                     .OR.STPTOT.EQ.1       ) ) )
     +            GO TO 121
 205     CONTINUE
         H=H*2.D0
         STPNUM=STPNUM/2
         STPTOT=STPTOT/2
 121 CONTINUE
      STPNUM=STPNUM+1
C
      IF (STPNUM.LE.STPTOT) GO TO 8
C
         RETURN
         END
```

Listing 17. program to solve DAVIS problem 1.5 (FORtran)
(test for RUKUFE subroutine)

```
$STORAGE:2
$NOFLOATCALLS
C  Program to calculate a reactor problem from
C  M. Davis' text Num. Anal. for ChE, example 5, page 34 and 41
C  integrating by a Runge-Kutta-Fehlberg method
C
      EXTERNAL RATE
      LOGICAL DEBUG,YPGVN
      INTEGER I,N,ND,INDEX
      REAL*8 Z,Y4(2),Y5(2)
     +,CONC(11),TEMP(11)
     +,H,TOL(2),DELZ,WORK(2,7)
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  CONC = dimensionless concentration of reactant
C  TEMP = dimensionless temperature
C  Z = dimensionless axial coordinate
C  Y4 = fourth order estimate of solution
C  Y5 = fifth order estimate of solution
C  DELZ = axial increment
C  WORK = scratch pad area
C  H = step size used in numerical routine
C  TOL = the desired maximum error in the integration
C  YPGVN = logical variable indicating that YPrime is GiVeN on calling
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      OPEN(1,FILE='CON')
C  This starts the actual integration of the diff eqn
      DEBUG=.FALSE.
      YPGVN=.FALSE.
      TOL(1)=1.D-4
      TOL(2)=1.D-4
      Y5(1)=1.D0
      Y5(2)=1.D0
      DELZ=0.1D0
      H=0.1D0
      N=2
      ND=2
      Z=0.D0
      CONC(1)=1.D0
      TEMP(1)=1.D0
      INDEX=1
      WRITE(1,82)
  82  FORMAT(1X,'Start of the RUNGE-KUTTA-FEHLBERG routine')
      DO 200 I=2,11
         CALL RUKUFE(N,ND,Z,DELZ,H,Y4,Y5,TOL,RATE,WORK
     &               ,INDEX,DEBUG,YPGVN)
         IF (INDEX.LT.0) WRITE(1,128)
 128     FORMAT(1X,'convergence problems in RKF')
         CONC(I)=Y5(1)
         TEMP(I)=Y5(2)
         WRITE(1,129)Z,H,Y5(1),Y5(2)
 129     FORMAT(1X,'Z = ',G11.4,'  H = ',G11.4,
     +              '  Y = ',G11.4,'  T = ',G11.4)
```

```
 200     CONTINUE
      WRITE(1,85)
  85  FORMAT(1X,'End of the RUNGE-KUTTA-FEHLBERG routine')
C  Output the values
      DO 140 I=1,11
         Z=0.1*FLOAT(I-1)
         WRITE(6,130)Z,CONC(I),TEMP(I)
 130     FORMAT(1X,F4.2,2(3X,G12.5))
 140     CONTINUE
      END

      SUBROUTINE RATE(N,ND,Z,Y,YP)
      INTEGER N,ND
      REAL*8 RATE,Z,Y(2),YP(2),ARR
C     WRITE(1,20)N,Z,Y
  20  FORMAT(1X,'        N=',I1,' Z=',G11.4,' Y=',2(G11.4,','))
      ARR=DEXP(3.21D0/Y(2))*Y(1)
      YP(1)=-0.1744D0*ARR
      YP(2)=0.06984D0*ARR
C     WRITE(1,10)YP
  10  FORMAT(1X,'                        DYDX=',2(G11.4,','))
      RETURN
      END
```

Listing 18. results from program to solve DAVIS problem 1.5 (DATa)
              (test for RUKUFE subroutine)


```
Start of the RUNGE-KUTTA-FEHLBERG routine
Z =    .1000    H =    .1000    Y =    .7009    T =    1.120
Z =    .2000    H =    .2000    Y =    .5295    T =    1.188
Z =    .3000    H =    .2000    Y =    .4140    T =    1.235
Z =    .4000    H =    .2000    Y =    .3301    T =    1.268
Z =    .5000    H =    .2000    Y =    .2666    T =    1.294
Z =    .6000    H =    .2000    Y =    .2173    T =    1.313
Z =    .7000    H =    .2000    Y =    .1783    T =    1.329
Z =    .8000    H =    .2000    Y =    .1470    T =    1.342
Z =    .9000    H =    .2000    Y =    .1217    T =    1.352
Z =    1.000    H =    .2000    Y =    .1010    T =    1.360
End of the RUNGE-KUTTA-FEHLBERG routineFile name missing or blank - Please enter
name
UNIT 6? CON

   .00     1.0000          1.0000
   .10      .70087         1.1198
   .20      .52954         1.1884
   .30      .41398         1.2347
   .40      .33010         1.2683
   .50      .26663         1.2937
   .60      .21732         1.3134
   .70      .17830         1.3291
   .80      .14701         1.3416
   .90      .12169         1.3517
  1.00      .10103         1.3600
```

Listing 19. QUASI-reversible electrochemical kinetics subroutine
             (FORtran)

```
      SUBROUTINE QUASI(N,ND,T,ALPHA,ADOT)
      INTEGER ND,N,NTD,MTD,MT,M1D,M1,IP(17),M1C,M,NT
      REAL*8 U(17,5),L(17,5),C(17,6)
     &       ,ALPHA(ND),P,KB,KF
     &       ,T,BETA,PZERO,PDOT
     &       ,ADOT(ND),A1(17),B2,B3,BNM1,BN
      COMMON NTD,MTD,MT,M1D,M1,IP,M,M1C,NT
     &       ,U,L,C,B2,B3,BNM1,BN
     &       ,BETA,PZERO,PDOT
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  On entry:
C
C     ALPHA   REAL*8,  the vector of dependent variables
C     N       INTEGER*2, the actual number of dependent variables
C     ND      INTEGER*2, the dimension of the arrays in the calling program
C     T       REAL*8, the independent variable (time)
C
C  On exit:
C
C     ADOT    REAL*8,  the vector of dALPHA/dt
C
C  Variables used:
C
C     A1      REAL*8, temporary vector of coefficients
C     I       INTEGER*2, counter
C     KB      REAL*8, backwards rate constant
C     KF      REAL*8, forwards rate constant
C     P       REAL*8, dimensionless potential
C
C  Global variables passed in an un-named common:
C
C     A       REAL*8, matrix of basis function values at Gauss and boundary pts
C     B2      REAL*8, element of b vector in equation:  A adot = C a + b
C     B3      REAL*8,                        "
C     BETA    REAL*8, electrochemical charge transfer coeficient
C     BN      REAL*8, element of b vector in equation:  A adot = C a + b
C     BNM1    REAL*8,                        "
C     C       REAL*8, matrix of 2nd deriv of basis fcns at Gauss and bndry pts
C     IP      INTEGER*2, a vector of pivot information used by BANSOL & BANDET
C     L       REAL*8, lower triangular matrix (from decomposition of A)
C     M       INTEGER*2, order of matrix A (square)
C     M1      INTEGER*2, number of subdiagonals in banded matrix A
C     M1C     INTEGER*2, number of subdiagonals in banded matrix C
C     M1D     INTEGER*2, actual column dimension for L matrix (used by BANDET)
C     MT      INTEGER*2, total bandwidth of banded matrix A
C     MTD     INTEGER*2, actual column dimension of A
C     NT      INTEGER*2, total bandwidth of banded matrix C
C     NTD     INTEGER*2, actual column dimension of C
C     PDOT    REAL*8, dimensionless potential scan rate
C     PZERO   REAL*8, initial dimensionless potential (for calculating P)
C     U       REAL*8, upper triangular matrix (from decomposition of A)
C
```

```
C
C  Uses subroutines:
C      BANMUL
C      BANSOL
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C
C  Useful for debugging:
C
C      WRITE(1,65) ND,N,NTD,NT,M1C,N,ND
C  65  FORMAT(1X,'ND=',I2,' N=',I2,' NTD=',I2,' NT=',I2
C      &          ,' M1C=',I2,' N=',I2,' ND=',I2)
C      IF (ALPHA(1).LT.0.1D0) WRITE(1,60)(ALPHA(I),I=1,N)
C  60         FORMAT(1X,'ALPHA:  ',5(G11.4,2X)
C      &              /1X,'           ',5(G11.4,2X))
C
C
C  find C * alpha
C       =    -----
C
       CALL BANMUL(ND,N,NTD,NT,M1C,C,ALPHA,N,ND,ADOT)


C
C  Useful for debugging:
C
C      IF (ALPHA(1).LT.0.1D0) WRITE(1,66)(ADOT(I),I=1,N)
C  66  FORMAT(1X,'C * alpha  ',5(G11.4,2X)
C      &              /1X,'           ',5(G11.4,2X))


C
C  calculate C * alpha + B
C            =    -----   -
C
       P=PZERO+T*PDOT
       KF=DEXP(-BETA*P)
       KB=DEXP((1.D0-BETA)*P)
C
C  following expression satisfies the diff eq at the surface of the electrode
C
       ADOT(1)=ADOT(1) - ( ALPHA(1)*KF - (1.D0-ALPHA(1))*KB )
C
C
C  Useful for debugging:
C
C      WRITE(1,67)BNM1,BN
C  67     FORMAT(1X,'b(n-1),b(n):  ',3(G11.4,2x))
C
         ADOT(2)=ADOT(2)+B2*ADOT(1)
         ADOT(3)=ADOT(3)+B3*ADOT(1)
         ADOT(N-1)=ADOT(N-1)+BNM1
         ADOT(N)=ADOT(N)+BN
C
C  Useful for debugging:
C
C      IF (ALPHA(1).LT.0.1D0) WRITE(1,68)(ADOT(I),I=1,N)
C  68     FORMAT(1X,'C * alpha + b  ',4(G11.4,2X)/12X,5(G11.4,2X))
```

```
C
C
C   find vector alpha-dot such that A * alpha-dot = C * alpha + B
C                                   =   ----------   =   -----   -
C

      DO 52 I=1,N-1
         A1(I)=ADOT(I+1)
  52       CONTINUE
      CALL BANSOL(ND,M,MTD,MT,M1D,M1,U,L,IP,A1)
      DO 53 I=1,N-1
         ADOT(I+1)=A1(I)
  53       CONTINUE
C
C   Useful for debugging:
C
      IF (ALPHA(1).LT.0.1D0) WRITE(1,50)(ADOT(I),I=1,N)
C
  50  FORMAT(1X,'ADOT:   ',4(G11.4,2X)/1X,'          ',5(G11.4,2X))
      RETURN
      END
```

Listing 20. QUASi-reversible ELECtrochemical model program (FORtran)

```
$LIST
$STORAGE:2
$NOFLOATCALLS
$PAGESIZE:60
C
C  Main program for calculating diffusion-reaction
C  for a 'quasi-reversible' electrochemical system
C  under conditions of scanning voltammetry.
C
      EXTERNAL QUASI
      LOGICAL DEBUG,YPGVN,PRINT
      INTEGER NBP,N,INDEX,IP(17),ND,NTD,MTD,M1D,MT,M1
     &       ,NS,NA,M,M1C,NT,IPOINT
      REAL*8 A(17,5),C(17,6),U(17,5),L(17,5)
     &       ,B2,B3,BNM1,BN,BP(9),T
     &       ,TSWCH,PSWCH,PSTART
     &       ,BETA,PZERO,PDOT,ALPHA(17),V(2,4)
     &       ,DELT,H,Y4(17),TOL(17),WORK(119),ADOT(17)
     &       ,AVEC(18),S(7),NEWBP(9),LASTI,P
      COMMON NTD,MTD,MT,M1D,M1,IP,M,M1C,NT
     &       ,U,L,C,B2,B3,BNM1,BN
     &       ,BETA,PZERO,PDOT
      EQUIVALENCE (A(1,1),U(1,1))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  Variables used in program:
C
C     ADOT    REAL*8, vector dALPHA/dt
C     ALPHA   REAL*8, vector of coefficients
C     AVEC    REAL*8, temporary storage for full vector of coef.
C     BP      REAL*8, vector of breakpoints
C     DEBUG   LOGICAL*2, set bebugging mode in RUKUFE
C     DELT    REAL*8, time step (delta time)
C     H       REAL*8, step size used by RUKUFE
C     I       INTEGER*2, counter
C     ID      INTEGER*2, indicates which derivative is desired 0th, 1st, or 2nd
C     IFAIL   INTEGER*2, failure code for BANDET subroutine
C     INDEX   INTEGER*2, state indicator for RUKUFE
C     IPOINT  INTEGER*2, counter for peaks as a function of scan rate
C     ISTEP   INTEGER*2, present step along voltammogram
C     IV      INTEGER*2, failure row number, used by BANDET
C     J       INTEGER*2, counter
C     LASTI   REAL*8, last value of current (i), used to detect peak current
C     N       INTEGER*2, number of coefficients
C     NA      INTEGER*2, total number of coefficients used
C     NAD     INTEGER*2, actual dimension of AVEC
C     NBD     INTEGER*2, actual dimension of breakpoint vector
C     NBP     INTEGER*2, number of breakpoints used
C     ND      INTEGER*2, actual dimension of ALPHA and ADOT
C     NEWBP   REAL*8, vector of redistributed breakpoints
C     NS      INTEGER*2, number of error intervals, used by REDIST
C     NSD     INTEGER*2, actual dimension of S (vector of errors in REDIST)
C     NSTEPS  INTEGER*2, number of steps to take in voltammogram
C     P       REAL*8, present value of dimensionless potential
```

```
C     PRINT  LOGICAL*2, sets extended print output mode of RUKUFE
C     PSTART REAL*8, starting & equilibrium potential for electrochemical system
C     PSWCH  REAL*8, potential switch point for reversing the scan direction
C     S      REAL*8, vector of approximation errors estimates, used by REDIST
C     T      REAL*8, time
C     TOL    REAL*8, truncation error tolerance used by RUKUFE
C     TSWCH  REAL*8, time for potential switch (reversing scan direction)
C     V      REAL*8, vector for values returned by BASWTS
C     WORK   REAL*8, work area for RUKUFE
C     Y4     REAL*8, fourth order estimate of dependent vars. returned by RUKUFE
C     YPGVN  LOGICAL*2, switch to tell RUKUFE that the first value of dY/dt
C                       is already calculated on entry (may save computation time)
C
C   These variables are global and are passed by an un-named COMMON:
C
C     A      REAL*8, matrix of basis function values at Gauss and boundary pts
C     B2     REAL*8, element of b vector in equation:  A adot = C a + b
C     B3     REAL*8,                          "
C     BETA   REAL*8, electrochemical charge transfer coeficient
C     BN     REAL*8, element of b vector in equation:  A adot = C a + b
C     BNM1   REAL*8,                          "
C     C      REAL*8, matrix of 2nd deriv of basis fcns at Gauss and bndry pts
C     IP     INTEGER*2, a vector of pivot information used by BANSOL & BANDET
C     L      REAL*8, lower triangular matrix (from decomposition of A)
C     M      INTEGER*2, order of matrix A (square)
C     M1     INTEGER*2, number of subdiagonals in banded matrix A
C     M1C    INTEGER*2, number of subdiagonals in banded matrix C
C     M1D    INTEGER*2, actual column dimension for L matrix (used by BANDET)
C     MT     INTEGER*2, total bandwidth of banded matrix A
C     MTD    INTEGER*2, actual column dimension of A
C     NT     INTEGER*2, total bandwidth of banded matrix C
C     NTD    INTEGER*2, actual column dimension of C
C     PDOT   REAL*8, dimensionless potential scan rate
C     PZERO  REAL*8, initial dimensionless potential (for calculating P)
C     U      REAL*8, upper triangular matrix (from decomposition of A)
C
C
C
C   Uses subroutines:
C
C      BANDET
C      BASWTS
C      FILL
C      QUASI
C      REDIST
C      RUKUFE
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C   A variety of suggested breakpoint sets for different potential scan
C      rates, 5 and 9 breakpoints.
C
C for NBP=5
C      DATA BP/0.D0,6.5D0,13.5D0,23.3D0,200.D0
C      &       ,0.D0,0.D0,0.D0,0.D0/
C for NBP=9
C      DATA BP/0.D0,3.25,6.5D0,10.D0,13.5D0,18.4,23.3D0
```

```
C      &          ,100.D0,200.D0/
C for edot=6.25
       DATA BP/0.D0,0.2D0,0.6D0,1.2D0,1.8D0,2.4D0,3.D0
      &          ,4.7D0,18.D0/
C for edot=0.25
C      DATA BP/0.D0,.9D0,1.8D0,4.4D0,5.8D0,7.4D0,9.5D0
C      &          ,12.6D0,25.D0/
C for edot=0.001
C      DATA BP/0.D0,1.D0,2.4D0,10.1D0,29.9D0,72.4D0,115.D0
C      &          ,157.5D0,200.D0/
       DEBUG=.FALSE.
       YPGVN=.FALSE.
       NBP=5
C  the number of ivp's to integrate is one less than the number
C  of Gauss points.
       N=NBP*2-1
       NBD=9
       NAD=18
       NSD=7
       ND=17
       MTD=5
       NTD=6
       M1D=5
       NT=6
       MT=5
       M1=2
       M=N-1
       M1C=2
       DO 20 I=1,N
          TOL(I)=1.D-5
   20     CONTINUE
       BETA=0.5D0
       PSTART=6.D0
       PSWCH=-6.D0
C      PDOT=0.001D0
       OPEN(3,FILE='CON',STATUS='NEW')
       OPEN(4,FILE='CON',STATUS='OLD')
C
C  Loop for calculating peaks as a function of scan rate
C
C      PSWCH=-30.D0
C      DO 100 IPOINT=1,100,2
C          PDOT=(0.042D0*DBLE(IPOINT))**2
C
          PZERO=PSTART
       WRITE(3,82)
   82  FORMAT(1X,'Enter potential ramp rate:  ')
       READ(4,84)PDOT
   84  FORMAT(F11.5)
       PDOT=DABS(PDOT)
       TSWCH=(PSWCH-PZERO)/(-PDOT)
       WRITE(3,1)
    1  FORMAT(1X,'Enter NSTEPS:  ')
       READ(4,2)NSTEPS
    2  FORMAT(I4)
C  Common value for NSTEPS is 120
       DELT=TSWCH/FLOAT(NSTEPS)
```

```
      H=DELT
C
C   take the vector of breakpoints and
C   fill the banded matrices A & C with the appropriate values
C   for a matrix representation of an electrochemical diffusion sys.
C
C  set up initial condition
C
      ALPHA(1)=DEXP(PZERO)/(1.D0+DEXP(PZERO))
      ALPHA(2)=0.D0
      DO 10 I=2,NBP-1
         ALPHA(2*I-1)=ALPHA(1)
         ALPHA(2*I )=ALPHA(2)
  10     CONTINUE
      ALPHA(2*NBP-1)=ALPHA(2)
      ALPHA(N)=0.D0
C
C  needed constants for boundary condition #1
C
      H=BP(2)-BP(1)
      ID=0
      CALL BASWTS(ID,H,V)
      B2=-V(1,1)
      B3=-V(2,1)
C
C  needed constants for boundary condition #2
C
      H=BP(NBP)-BP(NBP-1)
      ID=2
      CALL BASWTS(ID,H,V)
      BNM1=V(1,3)
      BN=V(2,3)
C
C  move spline values into working area
      CALL FILL(NBP,BP,A,C)
C
C
C    A (also U) starts with base spline values,
C      and is factored to upper triangular / lower triangular
C
C    A alpha' = C alpha + b
C    = -----   = -----   -
C
C    U L alpha' = C alpha + b
C    = = -----   = -----   -
C
      DO 57 I=1,M
         DO 53 J=1,5
            L(I,J)=0.D0
  53     CONTINUE
  57     CONTINUE
C
C  Useful for debugging:
C
C      WRITE(1,61)
C  61 FORMAT(1X,'A (U): ')
C      DO 62 I=1,M
```

```
C          WRITE(1,65)(A(I,J),J=1,MT)
C  62      CONTINUE
C       WRITE(1,63)
C  63   FORMAT(1X,'C:  ')
C       DO 64 I=1,N
C          WRITE(1,65)(C(I,J),J=1,NT)
C  65      FORMAT(4X,G11.4,5(2X,G11.4))
C  64      CONTINUE
C
C  set up boundary condition #2, constant bulk concentration
C
       BNM1=ALPHA(1)*BNM1
       BN=ALPHA(1)*BN
C
C  Useful for debugging:
C
C       WRITE(1,15)BNM1,BN
C  15   FORMAT(1X,'BNM1=',G11.4,' BN=',G11.4)
C       WRITE(1,56)ND,M,MTD,MT,M1D,M1
C  56   FORMAT(1X,'ND=',I2,' M=',I2,' MTD=',I2,' MT=',I2
C     &,' M1D=',I2,' M1=',I2)
C
C
C  decompose matrix U (A)
C
       CALL BANDET(ND,M,MTD,MT,M1D,M1,U,L,IP,IV,IFAIL)
       IF(IFAIL.EQ.0)GOTO 41
          WRITE(1,30)IV
  30      FORMAT(1X,'failure in BANDET, problem row is ',I2)
          STOP
  41   CONTINUE
C
C  Useful for debugging:
C
C       WRITE(1,21)
C  21   FORMAT(1X,'A (U):  ')
C       DO 22 I=1,M
C          WRITE(1,65)(A(I,J),J=1,MT)
C  22      CONTINUE
C       WRITE(1,23)
C  23   FORMAT(1X,'L:  ')
C       DO 24 I=1,M
C          WRITE(1,65)(L(I,J),J=1,MT)
C  24      CONTINUE
       INDEX=1
       H=DELT
       NA=2*NBP
       NS=NBP-2
       LASTI=0.D0
  90   CONTINUE
       T=0.D0
       PZERO=PSTART
       PDOT=-PDOT
       PRINT=.TRUE.
       DO 37 J=1,2*NBP-2
          AVEC(J)=ALPHA(J)
  37      CONTINUE
```

```fortran
      AVEC(2*NBP-1)=ALPHA(1)
      AVEC(2*NBP)=ALPHA(9)
C
C  Useful for debugging:
C
C     WRITE(0,46)(AVEC(J),J=1,11)
C     WRITE(1,46)(AVEC(J),J=1,11)
C
          DO 32 ISTEP=1,NSTEPS
   38         CONTINUE
C
C  Used for greater precision in finding peak values
C
C             IF(    ((DELT/H).GT.31)
C     &            .OR.(MOD(ISTEP,2).NE.0)
C     &            .OR.(NSTEP.LT.200)       )GOTO 39
C             DELT=DELT*2.D0
C             ISTEP=ISTEP/2
C             NSTEPS=NSTEPS/2

   39         CALL RUKUFE(N,ND,T,DELT,H,Y4,ALPHA,TOL,QUASI
     &              ,WORK,INDEX,DEBUG,YPGVN)
          P=PZERO+PDOT*T
          WRITE(0,42)P,ALPHA(2)
          WRITE(1,42)P,ALPHA(2)
   42         FORMAT(1X,G15.8,',',G15.8,)
C
C  Useful for debugging:
C
C             WRITE(1,60)(ALPHA(I),I=1,N)
C  60         FORMAT(1X,'ALPHA:  ',5(G11.4,2X)
C     &             /1X,'        ',5(G11.4,2X))
C
   31         CONTINUE
          IF(INDEX.LT.0)GOTO 70
          GOTO 44
              IF(P.GT.0.D0)GOTO 44
              IF((ALPHA(2).GT.LASTI).OR.(.NOT.PRINT))GOTO 44
C
C  Output peak values
C             WRITE(1,25)
C  25         FORMAT(1X,'negative going peak')
C             WRITE(1,42)DSQRT(DABS(PDOT)),LASTI
C             WRITE(0,42)PZERO+PDOT*(T-DELT),LASTI
C             WRITE(0,42)P,ALPHA(2)
C             WRITE(0,43)PDOT,LASTI/DSQRT(DABS(PDOT))
C  43         FORMAT(1X,'PDOT=',G11.4,'  SLOPE=',G11.4)

              DO 45 J=1,2*NBP-2
                 AVEC(J)=ALPHA(J)
   45            CONTINUE
              AVEC(2*NBP-1)=1.D0
              AVEC(2*NBP)=ALPHA(2*NBP-1)
              WRITE(0,46)(AVEC(J),J=1,2*nbp)
              WRITE(1,46)(AVEC(J),J=1,2*nbp)
   46            FORMAT(1X,'Coefficients: '
     &                 /1X,5(2X,G11.4)
```

```
     &                      /1X,5(2X,G11.4)
     &                      /1X,5(2X,G11.4))
                   CALL REDIST(BP,NBP,NBD,AVEC,NA,NAD,S,NS,NSD,NEWBP)
                   WRITE(0,47)(NEWBP(J),J=1,nbp)
                   WRITE(1,47)(NEWBP(J),J=1,nbp)
  47                FORMAT(1X,'Redist B P: ',5(G11.4,2X))
                   PRINT=.FALSE.
C                   GOTO 110
  44                CONTINUE
           LASTI=ALPHA(2)
  32       CONTINUE
 110       CONTINUE
C
C  Useful for debugging:
C
C          DO 120 J=1,NBP
C              BP(J)=0.2D0*NEWBP(J)+0.8D0*BP(J)
C 120          CONTINUE
C              WRITE(0,48)(BP(J),J=1,nbp)
C  48           FORMAT(1X,'  New B P: ',5(G11.4,2X))
C
 100       CONTINUE
        PZERO=PSWCH
        T=0.D0
        PDOT=-PDOT
        PRINT=.TRUE.
        DO 52 ISTEP=1,NSTEPS-1
                CALL RUKUFE(N,ND,T,DELT,H,Y4,ALPHA,TOL,QUASI
     &                        ,WORK,INDEX,DEBUG,YPGVN)
                P=PZERO+PDOT*T
                WRITE(0,42)P,ALPHA(2)
                WRITE(1,42)P,ALPHA(2)
C                WRITE(1,60)(ALPHA(I),I=1,N)
                IF(INDEX.LT.0)GOTO 70
                GOTO 54
                    IF(P.LT.0.D0)GOTO 54
                    IF((ALPHA(2).LT.LASTI).OR.(.NOT.PRINT))GOTO 54
C  Output peak values
C                   WRITE(1,26)
C  26                FORMAT(1X,'positive going peak')
C                   WRITE(1,42)PZERO+PDOT*(T-DELT),LASTI
C                   WRITE(1,42)P,ALPHA(2)
C                   PRINT=.FALSE.
                GOTO 54
                DO 55 J=1,2*NBP-2
            AVEC(J)=ALPHA(J)
  55                CONTINUE
                AVEC(2*NBP-1)=0.9975D0
                AVEC(2*NBP)=ALPHA(2*NBP-1)
                WRITE(0,46)(AVEC(J),J=1,2*nbp)
                WRITE(1,46)(AVEC(J),J=1,2*nbp)
                CALL REDIST(BP,NBP,NBD,AVEC,NA,NAD,S,NS,NSD,NEWBP)
                WRITE(0,47)(NEWBP(J),J=1,nbp)
                WRITE(1,47)(NEWBP(J),J=1,nbp)
                PRINT=.FALSE.
  54            CONTINUE
           LASTI=ALPHA(2)
```

```
   52      CONTINUE
C        GO TO 90
         STOP
   70  WRITE(1,72)
   72  FORMAT(1X,'Error stop, too many steps in RuKuFe')
         STOP
       END
```

Listing 21. BINary level of applied POTential subroutine (FORtran)

```
$LIST
      SUBROUTINE BINPOT(NK,NKD,T,ALPHA,ADOT)
      LOGICAL HIGH
      INTEGER ND,N,NTD,MTD,MT,M1D,M1,IP(17),M1C,M,NT
      REAL*8 U(17,5),L(17,5),C(17,6)
     &      ,ALPHA(NKD),KFH,KBH,KFL,KBL
     &      ,T,BETA,PHIGH,PLOW
     &      ,ADOT(NKD),A1(18),B2,B3,BNM1,BN
      COMMON HIGH
     &      ,NTD,MTD,MT,M1D,M1,IP,M,M1C,NT
     &      ,U,L,C,B2,B3,BNM1,BN
     &      ,BETA,PHIGH,PLOW,KFH,KBH,KFL,KBL
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  Variables used.
C
C  On entry:
C
C     ALPHA  REAL*8,  the vector of dependent variables
C     NK       INTEGER*2, the actual number of dependent variables
C     NKD      INTEGER*2, the dimension of the arrays in the calling program
C     T        REAL*8, the independent variable (time)
C
C  On exit:
C
C     ADOT   REAL*8,  the vector of dALPHA/dt
C
C  Variables used:
C
C     A1     REAL*8, temporary vector of coefficients
C     I      INTEGER*2, counter
C     KB     REAL*8, backwards rate constant
C     KF     REAL*8, forwards rate constant
C     N      INTEGER*2, row order of matrix A
C     ND     INTEGER*2, actual row dimension of matrix A
C     P      REAL*8, dimensionless potential
C
C
C  These variables are global and are passed by an un-named COMMON:
C
C     A      REAL*8, matrix of basis function values at Gauss and boundary pts
C     B2     REAL*8, element of b vector in equation:  A adot = C a + b
C     B3     REAL*8,                          "
C     BETA   REAL*8, electrochemical charge transfer coeficient
C     BN     REAL*8, element of b vector in equation:  A adot = C a + b
C     BNM1   REAL*8,                          "
C     C      REAL*8, matrix of 2nd deriv of basis fcns at Gauss and bndry pts
C     HIGH   LOGICAL*2, state of feedback element output
C     IP     INTEGER*2, a vector of pivot information used by BANSOL & BANDET
C     KBH    REAL*8, backward rate constant for high applied potential
C     KBL    REAL*8, backward rate constant for low applied potential
C     KFH    REAL*8, forward rate constant for high applied potential
C     KFL    REAL*8, forward rate constant for low applied potential
C     L      REAL*8, lower triangular matrix (from decomposition of A)
```

```
C    M        INTEGER*2, order of matrix A (square)
C    M1       INTEGER*2, number of subdiagonals in banded matrix A
C    M1C      INTEGER*2, number of subdiagonals in banded matrix C
C    M1D      INTEGER*2, actual column dimension for L matrix (used by BANDET)
C    MT       INTEGER*2, total bandwidth of banded matrix A
C    MTD      INTEGER*2, actual column dimension of A
C    NT       INTEGER*2, total bandwidth of banded matrix C
C    NTD      INTEGER*2, actual column dimension of C
C    PDOT     REAL*8, dimensionless potential scan rate
C    PHIGH    REAL*8, high value for dimensionless potential
C    PLOW     REAL*8, low value for dimensionless potential
C    PZERO    REAL*8, initial dimensionless potential (for calculating P)
C    U        REAL*8, upper triangular matrix (from decomposition of A)
C
C  Uses subroutines:
C
C    BANMUL, banded matrix multiply
C    BANSOL, banded matrix solver
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
      N=NK-1
      ND=NKD-1
C
C  Useful for debugging:
C
C    WRITE(1,65) ND,N,NTD,NT,M1C,N,ND
C 65  FORMAT(1X,'ND=',I2,' N=',I2,' NTD=',I2,' NT=',I2
C    &          ,' M1C=',I2,' N=',I2,' ND=',I2)
C    IF (ALPHA(1).LT.0.1D0) WRITE(1,60)(ALPHA(I),I=1,N)
C 60        FORMAT(1X,'ALPHA: ',5(G11.4,2X)
C    &              /1X,'       ',5(G11.4,2X))

C
C  find C * alpha
C       =   -----
C
      CALL BANMUL(ND,N,NTD,NT,M1C,C,ALPHA,N,NKD,ADOT)

C
C  Useful for debugging:
C
C    IF (ALPHA(1).LT.0.1D0) WRITE(1,66)(ADOT(I),I=1,N)
C 66  FORMAT(1X,'C * alpha ',5(G11.4,2X)
C    &              /1X,'          ',5(G11.4,2X))

C
C  calculate C * alpha + B
C            =   -----   -
C
C
C  Useful for debugging:
C
C    WRITE(1,67)BNM1,BN
C 67     FORMAT(1X,'b(n-1),b(n):  ',3(G11.4,2x))
C
C  following expression satisfies the diff eq at the surface of the electrode
C
```

```
               IF (HIGH) THEN
               ADOT(1)=ADOT(1)-(ALPHA(1)*KFH-(1.D0-ALPHA(1))*KBH)
                  ELSE
               ADOT(1)=ADOT(1)-(ALPHA(1)*KFL-(1.D0-ALPHA(1))*KBL)
                     ENDIF
           ADOT(2)=ADOT(2)+B2*ADOT(1)
           ADOT(3)=ADOT(3)+B3*ADOT(1)
           ADOT(N-1)=ADOT(N-1)+BNM1
           ADOT(N)=ADOT(N)+BN
C
C  Useful for debugging:
C
C        IF (ALPHA(1).LT.0.1D0) WRITE(1,68)(ADOT(I),I=1,N)
C  68     FORMAT(1X,'C * alpha + b  ',4(G11.4,2X)/12X,5(G11.4,2X))


C
C  find vector alpha-dot such that A * alpha-dot = C * alpha + B
C                                  =   ----------   =   -----   -
       DO 52 I=1,N-1
          A1(I)=ADOT(I+1)
   52     CONTINUE
       CALL BANSOL(ND,M,MTD,MT,M1D,M1,U,L,IP,A1)
       DO 53 I=1,N-1
          ADOT(I+1)=A1(I)
   53        CONTINUE
C
C  Useful for debugging:
C
C        IF (ALPHA(1).LT.0.1D0) WRITE(1,50)(ADOT(I),I=1,N)
C  50  FORMAT(1X,'ADOT:  ',4(G11.4,2X)/1X,'        ',5(G11.4,2X))


C
C  To integrate the dimensionless current flowing to the electrode
C
       ADOT(NK)=ALPHA(2)
       RETURN
       END
```

Listing 22. OSCilloGENic instrument simulation program (FORtran)

```
$LIST
$STORAGE:2
$NOFLOATCALLS
$PAGESIZE:60


C  Main program for calculating diffusion-reaction
C  for a 'quasi-reversible' electrochemical system
C  used in an oscillogenic instrument

      EXTERNAL BINPOT
      LOGICAL DEBUG,YPGVN,PRINT,HIGH
      INTEGER NBP,N,INDEX,IP(17),ND,NTD,MTD,M1D,MT,M1
     &       ,NS,NA,M,M1C,NT,CYCLE
     &       ,NK,NKD
      REAL*8 A(17,5),C(17,6),U(17,5),L(17,5)
     &       ,KFH,KBH,KFL,KBL
     &       ,B2,B3,BNM1,BN,BP(9),T
     &       ,PSTART,ISWITCH,QHIGH,QLOW
     &       ,BETA,PHIGH,PLOW,ALPHA(18),V(2,4)
     &       ,DELT,H,Y4(18),TOL(18),WORK(126),ADOT(18)
     &       ,AVEC(18),S(7),NEWBP(9),LASTI,P
      COMMON HIGH
     &       ,NTD,MTD,MT,M1D,M1,IP,M,M1C,NT
     &       ,U,L,C,B2,B3,BNM1,BN
     &       ,BETA,PHIGH,PLOW,KFH,KBH,KFL,KBL
      EQUIVALENCE (A(1,1),U(1,1))
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  Variables used in program:
C
C    ADOT    REAL*8, vector dALPHA/dt
C    ALPHA   REAL*8, vector of coefficients
C    AVEC    REAL*8, temporary storage for full vector of coef.
C    BP      REAL*8, vector of breakpoints
C    CYCLE   INTEGER*2, counts the number of full cycles of waveform
C    DEBUG   LOGICAL*2, set bebugging mode in RUKUFE
C    DELT    REAL*8, time step (delta time)
C    H       REAL*8, step size used by RUKUFE
C    I       INTEGER*2, counter
C    ID      INTEGER*2, indicates which derivative is desired 0th, 1st, or 2nd
C    IFAIL   INTEGER*2, failure code for BANDET subroutine
C    INDEX   INTEGER*2, state indicator for RUKUFE
C    ISWITCH REAL*8, value of current to switch on (feed back element)
C    IV      INTEGER*2, failure row number, used by BANDET
C    J       INTEGER*2, counter
C    LASTI   REAL*8, last value of current (i), used to detect peak current
C    N       INTEGER*2, number of coefficients
C    NA      INTEGER*2, total number of coefficients used
C    NAD     INTEGER*2, actual dimension of AVEC
C    NBD     INTEGER*2, actual dimension of breakpoint vector
C    NBP     INTEGER*2, number of breakpoints used.
C    ND      INTEGER*2, actual row dimension of matrix A
C    NEWBP   REAL*8, vector of redistributed breakpoints
C    NK      INTEGER*2, total number of equations to integrate (incl. charge)
```

```
C    NKD     INTEGER*2, actual dimension of arrays ALPHA and ADOT
C    NS      INTEGER*2, number of error intervals, used by REDIST
C    NSD     INTEGER*2, actual dimension of S (vector of errors in REDIST)
C    NSTEPS  INTEGER*2, number of steps to take in voltammogram
C    P       REAL*8, present value of dimensionless potential
C    PRINT   LOGICAL*2, sets extended print output mode of RUKUFE
C    PSTART  REAL*8, starting & equilibrium potential for electrochemical system
C    PSWCH   REAL*8, potential switch point for reversing the scan direction
C    QHIGH   REAL*8, higher  charge switch point for feedback element
C    QLOW    REAL*8, lower charge switch point for feedback element
C    S       REAL*8, vector of approximation errors estimates, used by REDIST
C    T       REAL*8, time
C    TOL     REAL*8, truncation error tolerance used by RUKUFE
C    TSWCH   REAL*8, time for potential switch (reversing scan direction)
C    V       REAL*8, vector for values returned by BASWTS
C    WORK    REAL*8, work area for RUKUFE
C    Y4      REAL*8, fourth order estimate of dependent vars. returned by RUKUFE
C    YPGVN   LOGICAL*2, switch to tell RUKUFE that the first value of dY/dt
C                      is already calculated on entry (may save computation time)
C
C  These variables are global and are passed by an un-named COMMON:
C
C    A       REAL*8, matrix of basis function values at Gauss and boundary pts
C    B2      REAL*8, element of b vector in equation:  A adot = C a + b
C    B3      REAL*8,                          "
C    BETA    REAL*8, electrochemical charge transfer coeficient
C    BN      REAL*8, element of b vector in equation:  A adot = C a + b
C    BNM1    REAL*8,                          "
C    C       REAL*8, matrix of 2nd deriv of basis fcns at Gauss and bndry pts
C    HIGH    LOGICAL*2, state of feedback element output
C    IP      INTEGER*2, a vector of pivot information used by BANSOL & BANDET
C    KBH     REAL*8, backward rate constant for high applied potential
C    KBL     REAL*8, backward rate constant for low applied potential
C    KFH     REAL*8, forward rate constant for high applied potential
C    KFL     REAL*8, forward rate constant for low applied potential
C    L       REAL*8, lower triangular matrix (from decomposition of A)
C    M       INTEGER*2, order of matrix A (square)
C    M1      INTEGER*2, number of subdiagonals in banded matrix A
C    M1C     INTEGER*2, number of subdiagonals in banded matrix C
C    M1D     INTEGER*2, actual column dimension for L matrix (used by BANDET)
C    MT      INTEGER*2, total bandwidth of banded matrix A
C    MTD     INTEGER*2, actual column dimension of A
C    NT      INTEGER*2, total bandwidth of banded matrix C
C    NTD     INTEGER*2, actual column dimension of C
C    PDOT    REAL*8, dimensionless potential scan rate
C    PHIGH   REAL*8, high value for dimensionless potential
C    PLOW    REAL*8, low value for dimensionless potential
C    PZERO   REAL*8, initial dimensionless potential (for calculating P)
C    U       REAL*8, upper triangular matrix (from decomposition of A)
C
C
C  Uses subroutines:
C
C    BANDET, banded matrix UL decomposition
C    BASWTS, basis function weights
C    FILL, fill A & C with basis fcns
C    BINPOT, binary valued potential rate calculation
```

```
C       REDIST, redistribute the breakpoints
C       RUKUFE, Runge-Kutta-Fehlberg integrator
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C
C  Two different breakpoint sets, for 5 and 9 breakpoints
C for NBP=5
      DATA BP/0.D0,6.5D0,13.5D0,23.3D0,500.D0
     &          ,0.D0,0.D0,0.D0,0.D0/
C for NBP=9
C      DATA BP/0.D0,3.25,6.5D0,10.D0,13.5D0,18.4,23.3D0
C     &          ,100.D0,200.D0/
C
      DEBUG=.FALSE.
      YPGVN=.FALSE.
      NBP=5
C the number of ivp's to integrate is one less than the number
C of coeficients plus one to integrate the current
      N=NBP*2-1
      NK=N+1
      NKD=18
      NBD=9
      NAD=18
      NSD=7
      ND=17
      MTD=5
      NTD=6
      M1D=5
      NT=6
      MT=5
      M1=2
      M=N-1
      M1C=2
      DO 20 I=1,N
         TOL(I)=1.D-5
  20     CONTINUE
      OPEN(3,FILE='CON',STATUS='NEW')
      OPEN(4,FILE='CON',STATUS='OLD')
C
C  Typical values:
C
C      BETA=0.5D0
C      PSTART=0.D0
C      NSTEPS=320
C      PHIGH=1.D0
C      PLOW=-1.D0
C      DELT=1.D0
C
C  Get the run parameters
C
      WRITE(3,88)
  88  FORMAT(1X,'Input equilibrium potential ')
      READ(4,81)PSTART
      WRITE(3,89)
  89  FORMAT(1X,'Input charge transfer coefficient (beta) ')
      READ(4,81)BETA
```

```
      WRITE(3,80)
  80  FORMAT(1X,'Input low potential ')
      READ(4,81)PLOW
  81  FORMAT(F10.5)
      WRITE(3,82)
  82  FORMAT(1X,'Input high potential ')
      READ(4,81)PHIGH
C
C
C

      WRITE(3,86)
  86  FORMAT(1X,'Input low value of charge transferred:  ')
      READ(4,81)QLOW
      WRITE(3,87)
  87  FORMAT(1X,'Input high value of charge transferred:  ')
      READ(4,81)QHIGH
C
C  set switch point for switching based on current flow
C
C     WRITE(3,83)
C 83  FORMAT(1X,'Input switching current ')
C     READ(4,81)ISWITCH
      WRITE(3,84)
  84  FORMAT(1X,'Input time step ')
      READ(4,81)DELT
      WRITE(3,85)PLOW,PHIGH,ISWITCH,DELT
      WRITE(1,85)PLOW,PHIGH,ISWITCH,DELT
  85  FORMAT(1X,'PLOW=',G11.4,' PHIGH=',G11.4
     &            ,' ISWITCH=',G11.4,' DELT=',G11.4)
C
C   take the vector of breakpoints and
C   fill the banded matrices A & C with the appropriate values
C   for a matrix representation of an electrochemical diffusion sys.
C
C  set up initial condition
C
      ALPHA(1)=DEXP(PSTART)/(1.D0+DEXP(PSTART))
      ALPHA(2)=0.D0
      DO 10 I=2,NBP-1
         ALPHA(2*I-1)=ALPHA(1)
         ALPHA(2*I)=ALPHA(2)
  10     CONTINUE
      ALPHA(2*NBP-1)=ALPHA(2)
      ALPHA(N)=0.D0
C
C  calculate present voltage for boundary condition #1
      P=PHIGH
C  calculate voltage dependent terms
      KFH=DEXP(-BETA*P)
      KBH=DEXP((1.D0-BETA)*P)
C  calculate present voltage
      P=PLOW
C  calculate voltage dependent terms
      KFL=DEXP(-BETA*P)
      KBL=DEXP((1.D0-BETA)*P)
C  needed constants for boundary condition #1
      H=BP(2)-BP(1)
```

```
      ID=0
      CALL BASWTS(ID,H,V)
      B2=-V(1,1)
      B3=-V(2,1)
C needed constants for boundary condition #2
      H=BP(NBP)-BP(NBP-1)
      ID=2
      CALL BASWTS(ID,H,V)
      BNM1=V(1,3)
      BN=V(2,3)
C move spline values into working area
C      WRITE(3,200)
C 200 FORMAT(1X,'Ready to call FILL')
      CALL FILL(NBP,BP,A,C)
C      WRITE(3,201)
C 201 FORMAT(1X,'Back from FILL')
C
C    A (also U) starts with base spline values,
C       and is factored to upper triangular / lower triangular
C
C    A alpha' = C alpha + b
C    = -----   = -----   -
C
C    U L alpha' = C alpha + b
C    = = -----   = -----   -
C
      DO 57 I=1,M
         DO 53 J=1,5
            L(I,J)=0.D0
  53        CONTINUE
  57     CONTINUE
C
C Helpful information during debugging
C
C      WRITE(1,61)
C 61   FORMAT(1X,'A (U):  ')
C      DO 62 I=1,M
C         WRITE(1,65)(A(I,J),J=1,MT)
C 62      CONTINUE
C      WRITE(1,63)
C 63   FORMAT(1X,'C:  ')
C      DO 64 I=1,N
C         WRITE(1,65)(C(I,J),J=1,NT)
C 65      FORMAT(4X,G11.4,5(1X,G11.4))
C 64      CONTINUE
C
C set up boundary condition #2, constant bulk concentration
C
C      WRITE(1,15)BNM1,BN,ALPHA(1)
      BNM1=ALPHA(1)*BNM1
      BN=ALPHA(1)*BN
C      WRITE(1,15)BNM1,BN,ALPHA(1)
C 15   FORMAT(1X,'BNM1=',G11.4,' BN=',G11.4,' ALPHA(1)=',G11.4)
C
C decompose matrix U (A)
C
C      WRITE(3,204)ND,M,MTD,MT,M1D,M1
```

```
C 204  FORMAT(1X,'ND=',I2,' M=',I2,' MTD=',I2
C      &            ,' MT=',I2,' M1D=',I2,' M1=',I2)
C       WRITE(3,202)
C 202  FORMAT(1X,'Calling BANDET')
       CALL BANDET(ND,M,MTD,MT,M1D,M1,U,L,IP,IV,IFAIL)
C       WRITE(3,203)
C 203  FORMAT(1X,'Back from BANDET')
       IF(IFAIL.EQ.0)GOTO 41
          WRITE(1,30)IV
  30      FORMAT(1X,'failure in BANDET, problem row is ',I2)
          STOP
  41   CONTINUE
C       WRITE(1,21)
C  21  FORMAT(1X,'A (U):  ')
C       DO 22 I=1,M
C          WRITE(1,65)(A(I,J),J=1,MT)
C  22      CONTINUE
C       WRITE(1,23)
C  23  FORMAT(1X,'L:  ')
C       DO 24 I=1,M
C          WRITE(1,65)(L(I,J),J=1,MT)
C  24      CONTINUE
       INDEX=1
       H=DELT
       NA=2*NBP
       NS=NBP-2
       LASTI=1.D38
  90   CONTINUE
       T=0.D0
       PRINT=.TRUE.
       HIGH=.FALSE.
       P=PLOW
       CYCLE=1
       DO 37 J=1,2*NBP-2
          AVEC(J)=ALPHA(J)
  37      CONTINUE
       AVEC(2*NBP-1)=ALPHA(1)
       AVEC(2*NBP )=ALPHA(2*NBP-1)
C       WRITE(0,46)(AVEC(J),J=1,11)
C      WRITE(1,46)(AVEC(J),J=1,11)
C       DO 32 ISTEP=1,NSTEPS
  38      CONTINUE
C
C  Input voltage switching algorithims:
C     external feedback systems:
C
C
C     switching based on charge
C
          IF (HIGH) THEN
C
C         if output voltage is high, and charge is less than qlow
C            then switch to low output voltage
C
             IF (ALPHA(NK).LT.QLOW) THEN
                P=PLOW
                HIGH=.FALSE.
```

```
                CYCLE=CYCLE+1
C
C  This section is for precise timing of period
C
C                IF (CYCLE.GT.15) WRITE(1,42)T,P,ALPHA(2),ALPHA(N),ALPHA(NK)
C                IF (CYCLE.EQ.16) DELT=DELT/100.D0
C                IF (CYCLE.EQ.18) STOP
                 WRITE(1,42)T,P,ALPHA(2),ALPHA(N),ALPHA(NK)
                 ENDIF
C
C          if output voltage is low, and charge is greater than qhigh
C             then switch to high output voltage
C
           ELSEIF (ALPHA(NK).GT.QHIGH) THEN
              P=PHIGH
              HIGH=.TRUE.
           ENDIF
C
C     switching based on current
C
C         IF (DABS(LASTI).LT.ISWITCH) THEN
C            IF (HIGH) THEN
C                 P=PLOW
C                 HIGH=.FALSE.
C                 CYCLE=CYCLE+1
C                 WRITE(1,42)T,P,ALPHA(2),ALPHA(N),ALPHA(NK)
C              ELSE
C                 P=PHIGH
C                 HIGH=.TRUE.
C              ENDIF
C            ENDIF
C
C     forcing function (non-oscillogenic)
C
C         IF (MOD(ISTEP,80).LT.40) THEN
C                 P=PLOW
C                 HIGH=.FALSE.
C              ELSE
C                 P=PHIGH
C                 HIGH=.TRUE.
C              ENDIF
            CALL RUKUFE(NK,NKD,T,DELT,H,Y4,ALPHA,TOL,BINPOT
     &                    ,WORK,INDEX,DEBUG,YPGVN)
            DO 101 I=1,NBP-1
               IF(ALPHA(2*I-1).GT.0.9975D0)ALPHA(2*I-1)=0.9975D0
  101          CONTINUE
            IF(ALPHA(2*NBP-1).LT.0.D0)ALPHA(2*NBP-1)=0.D0
C           WRITE(0,42)P,ALPHA(2)
C           WRITE(1,42)P,ALPHA(2)
            WRITE(3,42)T,P,ALPHA(2),ALPHA(N),ALPHA(NK)
C           WRITE(1,42)T,P,ALPHA(2),ALPHA(2*NBP-1)
  42           FORMAT(1X,G12.5,4(',',G11.4))
C           WRITE(1,60)(ALPHA(I),I=1,N)
  60        FORMAT(1X,'ALPHA:  ',5(G11.4,2X)
     &             /1X,'        ',5(G11.4,2X))
  31        CONTINUE
            IF(INDEX.LT.0)GOTO 70
```

```
C
C  Useful for debugging:
C
C                    DO 45 J=1,2*NBP-2
C            AVEC(J)=ALPHA(J)
C  45                    CONTINUE
C                   AVEC(2*NBP)=ALPHA(2*NBP-1)
C                   WRITE(1,46)(AVEC(J),J=1,2*nbp)
C  46                    FORMAT(1X,'Coeficients:   '
C      &                      /1X,5(2X,G11.4)
C      &                      /1X,5(2X,G11.4)
C      &                      /1X,5(2X,G11.4))
C               CALL REDIST(BP,NBP,NBD,AVEC,NA,NAD,S,NS,NSD,NEWBP)
C               WRITE(0,47)(NEWBP(J),J=1,nbp)
C               WRITE(1,47)(NEWBP(J),J=1,nbp)
C  47           FORMAT(1X,'Redist B P:  ',5(G11.4,2X))
   44              CONTINUE
         LASTI=ALPHA(2)
C         STOP          for oscgen
         IF(CYCLE.LT.200)GOTO 38
   32      CONTINUE
  110      CONTINUE
C
C  Useful for debugging:
C
C         DO 120 J=1,NBP
C             BP(J)=0.2D0*NEWBP(J)+0.8D0*BP(J)
C 120         CONTINUE
C             WRITE(0,48)(BP(J),J=1,nbp)
   48            FORMAT(1X,'  New B P:  ',5(G11.4,2X))
  100      CONTINUE
         STOP
   70  WRITE(1,72)
   72  FORMAT(1X,'Error stop, too many steps in RuKuFe')
         STOP
       END
```

Listing 23 Lab Tech Notebook settings

```
                    NORMAL DATA ACQUISITION / CONTROL SETUP
Number of Channels (Pages) (1..50)            4
Current Page (1..4)                           1    <-
Channel Type [0:  Analog Input,               1
   1: Analog Output (Control)]    (0..1)
DA Board Number (0..4)                         0
DA Channel Number (0..4)                       0
DA Offset Constant                        0.000
DA Scale Factor                           1.000
Loop [0:open, 1:closed] (0..1)                 0
Waveform file (WFn.prn) (0..30000)             1
Sampling Rate, Hz (7.0E-03..600.0)       25.000
Loop Duration, sec. (0.0..1.0E+06)       12.000

                    NORMAL DATA ACQUISITION / CONTROL SETUP
Number of Channels (Pages) (1..50)            4
Current Page (1..4)                           2    <-
Channel Type [0:  Analog Input,               0
   1: Analog Output (Control)]    (0..1)
AD Board Number (0..4)                         0
AD Channel Number (0..15)                      0
Number of Loops (1..32767)                     1
AD Input Range, volts [0:±10,                  0
   1:±5, 2:±2.5, 3:±1.25] (0..3)
AD Scale Factor                           1.000
AD Offset Constant                        0.000
Buffer Size (10..32767)                    2048
Number of Stages (1..4)                       2

Sampling Rate, Hz (7.0E-03..600.0)       25.000    25.000
Stage Duration, sec.  (0.0..1.0E+06)      4.000     8.000
Starting Method [0: Normal                    0         0
   1: Trigger, 2: Time Delay] (0..2)
Trigger Pattern to AND (0..255)               0         0
Trigger Pattern to XOR (0..255)               0         0
Time Delay, sec. (0.0..1.0E+06)           0.000     0.000
  AD: ANALOG INPUT, DA: ANALOG OUTPUT, DI: DIGITAL INPUT, DO: DIGITAL OUTPUT
```

```
                 NORMAL DATA ACQUISITION / CONTROL SETUP
Number of Channels (Pages) (1..60)            4
Current Page (1..4)                           3    <-
Channel Type [0:  Analog Input,               0
   1: Analog Output (Control)]   (0..1)
AD Board Number (0..4)                         0
AD Channel Number (0..15)                      1
Number of Loops (1..32767)                     1
AD Input Range, volts [0:±10,                  3
   1:±5, 2:±2.5, 3:±1.25] (0..3)
AD Scale Factor                           1.000
AD Offset Constant                        0.000
Buffer Size (10..32767)                   2048
Number of Stages (1..4)                       2

Sampling Rate, Hz (7.0E-03..600.0)       25.000    25.000
Stage Duration, sec.  (0.0..1.0E+06)      4.000     8.000
Starting Method [0: Normal                    0         0
   1: Trigger, 2: Time Delay] (0..2)
Trigger Pattern to AND (0..255)               0         0
Trigger Pattern to XOR (0..255)               0         0
Time Delay, sec. (0.0..1.0E+06)           0.000     0.000
   AD: ANALOG INPUT, DA: ANALOG OUTPUT, DI: DIGITAL INPUT, DO: DIGITAL OUTPUT

                 NORMAL DATA ACQUISITION / CONTROL SETUP
Number of Channels (Pages) (1..60)            4
Current Page (1..4)                           4    <-
Channel Type [0:  Analog Input,               0
   1: Analog Output (Control)]   (0..1)
AD Board Number (0..4)                         0
AD Channel Number (0..15)                      2
Number of Loops (1..32767)                     1
AD Input Range, volts [0:±10,                  0
   1:±5, 2:±2.5, 3:±1.25] (0..3)
AD Scale Factor                           1.000
AD Offset Constant                        0.000
Buffer Size (10..32767)                   2048
Number of Stages (1..4)                       1

Sampling Rate, Hz (7.0E-03..600.0)       25.000
Stage Duration, sec.  (0.0..1.0E+06)     12.000
Starting Method [0: Normal                    0
   1: Trigger, 2: Time Delay] (0..2)
Trigger Pattern to AND (0..255)               0
Trigger Pattern to XOR (0..255)               0
Time Delay, sec. (0.0..1.0E+06)           0.000
   AD: ANALOG INPUT, DA: ANALOG OUTPUT, DI: DIGITAL INPUT, DO: DIGITAL OUTPUT
```

406

WINDOW SETUP

Number of Windows (0..5)	2	
Left Limit, x0 (0.0..1.0)	0.100	0.100
Lower Limit, y0 (0.0..1.0)	0.700	0.100
Right Limit, x1 (0.0..1.0)	0.990	0.990
Upper Limit, y1 (0.0..1.0)	0.990	0.500
Y Axis Title	volts	microamp
Time Axis Title	seconds	seconds
Length of Time Axis in sec.	12.000	12.000
Time Tic Start Value	0.000	0.000
Time Tic End Value	12.000	12.000
Number of Time Tics (0..11)	7	7
Y Tic Start Value	-1.200	-10.000
Y Tic End Value	0.000	10.000
Number of Y Tics (0..11)	3	5
Color (0: black, 1: blue, 2: magenta, 3:white) (0..3)	0	0

TRACES SETUP

Number of Traces (0..5)	2	
Board Number (0..4)	0	0
Channel Number (0..4)	1	0
Window Number (0..4)	0	1
Line Color (0: black, 1: blue, 2: magenta, 3:white) (0..3)	2	2
Minimum Displayed Value	-1.200	-10.000
Maximum Displayed Value	0.000	10.000

FILES SETUP

Number of Data Files (0..15)	1
Current Data Files (1..1)	1
Data File Name	va
Data Storage Mode [0: AR	0
1: BI, 2: AI, 3:BR] (0..3)	
Number of Header Lines (0..4)	2
Header Line 1	SLIDE,LINEAR,-.8,-.2,.2,LINEAR,-10,10,5
Header Line 2	Scanning Voltammetry of Cd(II)/Pb(II)
Header Line 3	
Header Line 4	
Number of Channels in File (0..5)	3

Board Number (0..4)	0	0	0
Channel Number (0..15)	1	2	0
Channel Units	potentia	(volts)	

AR = ASCII REAL, BI = BINARY INTEGER, AI = ASCII INTEGER, BR = BINARY REAL

Global Setup Checking -- Setup OK

MODE--Normal

BOARD	CHANNEL	TYPE	DURATION	MAX RATE	SAMPLES	FILE	WINDOW
0	0	DA	1.200E+01	2.500E+01	300	wf1	
0	0	AD	1.200E+01	2.500E+01	300	va	1
0	1	AD	1.200E+01	2.500E+01	300	va	0
0	2	AD	1.200E+01	2.500E+01	300	va	

Listing 24 BASIC program to create waveform file for Lab Tech Notebook
'open loop' control.

```
10 OPEN "B:WF1.PRN" FOR OUTPUT AS 1
20 PRINT #1,"411"           'number of points in open loop file for LTN
30 FOR I=0 TO 205           'negative going sweep
40 PRINT #1,-I/204.8
50 NEXT I
60 FOR I=204 TO 0 STEP -1   'positive going sweep
70 PRINT #1,-I/204.8
80 NEXT I
90 CLOSE
100 END
```

Listing 25 BASIC program to scan the average voltage and measure the oscillogenic frequency.

```
10  '*****************************************************************
20  '*                                                              *
30  '*              using the dash-8 to measure oscillogenic frequency  *
40  '*              while varying the average applied voltage       *
42  '*                                                              *
45  '*   adapted by P. Kip Mercure from programs supplied by        *
50  '*          MetraByte Corporation            Rev. 1.15  6-6-84  *
60  '*****************************************************************
70  '
80  '      This program shows you how to measure frequency i.e. use your DASH-8
90  ' as a frequency counter using Mode 15 of the CALL.
100 '
110 SCREEN 0,0,0:KEY OFF:WIDTH 80:CLS
120 '
130 '----- Load CALL routine & initialize - Contract workspace to 48K ---------
140 'Customize this load section to your own requirements:-
150 CLEAR, 49152!
160 DEFINT A-Z
170 DEF SEG = 0
180 SG = 256 * PEEK(&H511) + PEEK(&H510)
190 SG = SG + 49152!/16
200 DEF SEG = SG
210 BLOAD "A:DASH8.BIN", 0    'use .BIN disk file on A: drive
230 BASADR%=816                ' address: 330 Hex
250 DASH8 = 0
260 FLAG% = 0
270 MD% = 0
272 BOTTOM.RANGE#=-5
274 RANGE#=10
280 CALL DASH8 (MD%, BASADR%, FLAG%)
290 IF FLAG% <>0 THEN PRINT"INSTALLATION ERROR"
300 '
302 BASE% = 832   'Metrabyte DAC base address
304 CH% = 0       '                   channel number
310 FRQ%(0)=1000 'measurement (gate) time in milliseconds
320 '
570 MD%=15
580 INPUT "starting voltage = ";VSTART#
590 INPUT "switching voltage = ";VSWITCH#
600 INPUT "number of steps in half cycle = ";N
610 DV#=(VSWITCH#-VSTART#)/N
620 DIM V#(2*N),F#(2*N)
630 VOLTS#=VSTART#:GOSUB 1290
640 PRINT "press any key to start"
650 A$=INKEY$
660 IF A$="" THEN 650
670 ' DAC voltage scan

680 FOR INDEX=0 TO N       'forward voltage scan
690 VOLTS#=VSTART#+INDEX*DV#
700 GOSUB 1290
```

```
710 'Make measurement
720 FRQ%(1)=0
730 CALL DASH8 (MD%, FRQ%(0), FLAG%)
740 IF FRQ%(1) >= 0 THEN FREQ# = FRQ%(1) ELSE FREQ# = FRQ%(1) + 65536!
750 FREQ# = FREQ# * 1000 / (FRQ%(0) * .99985)
760 PRINT USING "Frequency is  ##########  Hz";FREQ#
770 V#(INDEX)=VOLTS#:F#(INDEX)=FREQ#
780 NEXT INDEX

790 FOR INDEX=1 TO N       'reverse voltage scan
800 VOLTS#=VSWITCH#-INDEX*DV#
810 GOSUB 1290
820 'Make measurement
830 FRQ%(1)=0
840 CALL DASH8 (MD%, FRQ%(0), FLAG%)
850 IF FRQ%(1) >= 0 THEN FREQ# = FRQ%(1) ELSE FREQ# = FRQ%(1) + 65536!
860 FREQ# = FREQ# * 1000 / (FRQ%(0) * .99985)
870 PRINT USING "Frequency is  ##########  Hz";FREQ#
880 V#(N+INDEX)=VOLTS#:F#(N+INDEX)=FREQ#
890 NEXT INDEX

910 SOUND 600,10:SOUND 560,15     'make a noise when done to alert operator
915 SOUND 600,10:SOUND 560,15
920 INPUT "enter filename for data:   ",A$
930 OPEN A$ FOR OUTPUT AS 1
940 FOR INDEX=0 TO 2*N
950 PRINT #1, USING "##.####, ######";V#(INDEX);F#(INDEX)
955 PRINT USING "##.####, ######";V#(INDEX);F#(INDEX)
960 NEXT INDEX
970 CLOSE #1
980 END
990 ''
1290 '' calculate DAC data word value
1300 DATA.VALUE#=(VOLTS#-BOTTOM.RANGE#)*4096/RANGE#
1310 DATA.VALUE#=CINT(DATA.VALUE#)
1320 IF DATA.VALUE# > 4095 THEN DATA.VALUE# = 4095
1330 '' calculate and print out rounded DAC output voltage
1340 PRINT
1350 VOLTS#=DATA.VALUE#*RANGE#/4096
1360 VOLTS#=VOLTS#+BOTTOM.RANGE#
1370 PRINT "Actual DAC ouput voltage will be ";
1380 PRINT USING "###.###";VOLTS#;
1390 PRINT " volts"
1400 XH%=INT(DATA.VALUE#/16) : XL% = 16 * (DATA.VALUE#-XH%*16)
1410 OUT BASE% + CH%*2, XL% : OUT BASE% +1 + CH%*2, XH%
1420 RETURN
```

411

Listing 26. GRAPH program (BASic)

```
10 GOTO 140
20 P%=LEN(P$)                'output string to plotter
30 FOR N%=1 TO P%            'for every character in the string,
40 A$=MID$(P$,N%,1)
50 WAIT &H3FE,32             'wait for handshake line clear
60 OUT &H3F8,ASC(A$)         ' output character
70 NEXT N%
80 RETURN
90 AA = XA +SX * (X-X1)      'scale coordinates for screen plot
100 BB = YA -SY * (Y-Y1)
101 OF=0
110 IF AA < 9 OR BB < 9 THEN OF=-1:RETURN
120 IF AA > 271 OR BB > 151 THEN OF=-1:RETURN
130 RETURN
140 CLEAR(1000)
150 DIM TI(25)
160 DEFDBL D,L
170 LN=LOG(10)
180 PRINT
190 PRINT "This is a program to draw graphs on the HP7470A plotter,"
200 PRINT "and provide graph previews on an IBM PC screen."
210 PRINT "Version 3.0  16-Oct-84   by Peter Kip Mercure"
220 PRINT
230 LINE INPUT "Do you want a screen graph? (Yes or No) ";A$
240 IF "N"=CHR$(ASC(LEFT$(A$,1))AND -33) THEN SG=0 ELSE SG=-1  'Screen Graph
250 PRINT "File will be output to the ";
260 IF SG THEN PRINT "screen." ELSE PRINT "plotter."
270 LINE INPUT "Does file contain formatting information? (Yes or No) ";A$
280 IF "N"=CHR$(ASC(LEFT$(A$,1))AND -33) THEN FI=0 ELSE FI=-1  'File Info
290 IF FI THEN PRINT "Assuming formating information to be in data file."
300 IF NOT(FI) THEN PRINT "Will take formating information from keyboard."
310 PRINT "Type HELP for the filename if you want to know the data format"
320 PRINT
330 LINE INPUT "Data filename?";A$
340 IF NOT(A$="HELP" OR A$="help" OR A$="Help") THEN GOTO 590
350 PRINT"    The structure of the data file is as follows:"
360 PRINT"Line 1:  the information for scaling the axes:  ";
370 PRINT"ty,xs,XL,XH,XD,ys,YL,YH,YD"
380 PRINT"  ty = graph type:  Slide (with title at the top),"
390 PRINT"                    Portrait (";CHR$(219);"  with title at bottom)"
400 PRINT"                 or Landscape (";CHR$(220);CHR$(220);"  with title at
the bottom)"
410 PRINT"                                        X-axis  Y-axis"
420 PRINT"  linear or logarithmic scale            xs      ys"
430 PRINT"  low limit to the axis                  XL      YL"
440 PRINT"  high limit to the axis                 XH      YH"
450 PRINT"  distance between tic marks on axis     XD      YD"
460 PRINT"Line 2:  the graph title, ";
470 PRINT"if this line is blank then draw no axes"
480 PRINT"Line 3:  if not a SLIDE then this is used for 2nd title line"
490 PRINT"Line 3/4:  the label for the X-axis"
500 PRINT"Line 4/5:  the label for the Y-axis"
```

```
510 PRINT"Line 5/6 and on:  X,Y pairs.  If the X value is less than the XL "
520 PRINT" value, then a new curve is started.  The Y value will then determine"
530 PRINT" the action:  Y=0 pause(change pen), Y>0 draw points only:  1)circle,"
540 PRINT" 2)crescent, 3)triangle, 4)square, 5)five pointed star, 6)six pointed"
550 PRINT" star, Y<0 draw connecting lines only: -1)solid, -2)dashed 50-50,"
560 PRINT" -3)dotted, -4)dash-dot, -5)long dash-short dash, -6)long dash two"
570 PRINT" short dashes, -7)at following location (x,y) write label"
580 GOTO 320
590 OPEN "I",#2,A$
600 IF SG THEN 790
610 OUT &H3FB,128           'inititate RS-232C port
620 OUT &H3F8,24            '4800 baud
630 OUT &H3F9,0
640 OUT &H3FB,3             '8 bits/char, 1 stop bit, no parity
650 OUT &H3FC,3
660 ' plotter switches set: 0   0   0   1   1   0   0   1
670 '                       S2  S1  Y   US  B4  B3  B2  B1
680 '
690 ' RS-232C lines are:  HP end is male, IBM end is female connector
700 '                     HP7470A pin 2  to IBM PC pin 2
710 '                        "    pin 3 to     "   pin 3
720 '                        "    pin 20 to    "   pin 6
730 '                        "    pin 7 to     "   pin 7
740 '                     IBM PC pin 5 to IBM PC pin 20
750 '
760 P$="IP710,1240,9210,7200;IW710,1240,9210,7200;"        'set plot limits
770 GOSUB 20                                      '1 plotter unit = 0.025 mm
780 '
790 IF FI THEN 1040
800 IF SG THEN TY$="S":GOTO 860
810 PRINT"Input graph type:  Slide (with title at the top),"
820 PRINT"                   Portrait (";CHR$(219);"  with title at bottom)"
830 PRINT"                   or Landscape (";CHR$(220);CHR$(220);"  with title at
the bottom)"
840 LINE INPUT TY$
850 TY$=CHR$(ASC(LEFT$(TY$,1)) AND -33)             'force 1st letter to upper case
860 LINE INPUT "'linear' or 'logarithmic' x scale:  ";XS$
870 INPUT "low limit to the x axis = ";XL
880 INPUT "high limit to the x axis = ";XH
890 INPUT "distance between tic marks on x axis = ";XD
900 LINE INPUT "'linear' or 'logarithmic' y scale:  ";YS$
910 INPUT "low limit to the y axis = ";YL
920 INPUT "high limit to the y axis = ";YH
930 INPUT "distance between tic marks on y axis = ";YD
940 LINE INPUT "Graph title:  ";T1$                 'get first title line
950 IF T1$="" THEN 990                              'skip axes
960 IF TY$<>"S" THEN LINE INPUT "2nd line:  ";T2$ 'if not Slide then get second
title line
970 LINE INPUT "X Axis title:  ";X$                 'get x-axis label
980 LINE INPUT "Y Axis title:  ";Y$                 'get y-axis label
990 LINE INPUT "Is this a line plot? (Yes or No)";A$
1000 IF "N"=CHR$(ASC(LEFT$(A$,1))AND -33) THEN LNE=0 ELSE LNE=-1  'Line plot
1010 PRINT "Enter number for ";:IF LNE THEN PRINT "line "; ELSE PRINT "point ";
1020 INPUT "style:  ";STYLE:STYLE=ABS(STYLE)
1030 GOTO 1110
```

```
1040 INPUT #2,TY$,XS$,XL,XH,XD,YS$,YL,YH,YD         'get scaling information
1050 TY$=CHR$(ASC(LEFT$(TY$,1)) AND  -33)           'force 1st letter to upper case
1060 LINE INPUT #2,T1$                              'get first title line
1070 IF T1$="" THEN 1110                            'skip axes
1080 IF TY$<>"S" THEN LINE INPUT #2,T2$ 'if not Slide then get second title line
1090 LINE INPUT #2,X$                               'get x-axis label
1100 LINE INPUT #2,Y$                               'get y-axis label
1110 XS$=MID$(XS$,2,1):IF XS$="I" OR XS$="i" THEN XS=-1 ELSE XS=0 'Linear X
scale
1120 YS$=MID$(YS$,2,1):IF YS$="I" OR YS$="i" THEN YS=-1 ELSE YS=0 'Linear Y
scale
1130 '                  TY$ = graph TYpe:  Slide, Landscape, Portrait
1140 '                  XS = X Scale, -1 = Linear, 0 = Logarithmic
1150 '                  YS = Y Scale, -1 = Linear, 0 = Logarithmic
1160 '
1170 IF SG THEN 4500
1180 IF TY$="S" THEN X1=8350:Y1=6581:X2=1184:Y2=1575     'scale axes values
1190 IF TY$="L" THEN X1=8350:Y1=6095:X2=1144:Y2=1300
1200 IF TY$="P" THEN X1=2100:Y1=8105:X2=6766:Y2=768
1210 IF XS THEN XA=(X2-X1)/(XH-XL) ELSE XA=(X2-X1)/(LOG(XH)-LOG(XL))
1220 IF XS THEN XB=X1-XA*XL ELSE XB=X1-XA*LOG(XL)
1230 IF YS THEN YA=(Y2-Y1)/(YH-YL) ELSE YA=(Y2-Y1)/(LOG(YH)-LOG(YL))
1240 IF YS THEN YB=Y1-YA*YL ELSE YB=Y1-YA*LOG(YL)
1250 IF TY$="S" THEN P$="SI0.2612,0.3725;" ELSE P$="SI0.1741,.2793;" 'title size
1260 GOSUB 20
1270 IF TY$="P" THEN P$="SP1;DI0,1;" ELSE P$="SP1;DI-1,0"
1280 GOSUB 20
1290 IF T1$="" THEN 2330              'if title is null then skip axes
1300 '
1310 S=XS:D=XD:H=XH:L=XL:S$="X":GOSUB 3700:XT%=T%:DX=DD:XD=D 'check X axis
labels
1320 S=YS:D=YD:H=YH:L=YL:S$="Y":GOSUB 3700:YT%=T%:DY=DD:YD=D 'check Y axis
labels
1330 '
1340 T%=LEN(T1$)/2                                        'output graph title
1350 IF TY$="S" THEN B$="DI-1,0;SP1;PA4960,1410;LB"
1360 IF TY$="L" THEN B$="DI-1,0;SP1;PA4960,6965;LB"
1370 IF TY$="P" THEN B$="DI0,1;SP1;PA8975,4200;LB"
1380 C$=STRING$(T%,8)
1390 P$=B$+C$+T1$+CHR$(3):GOSUB 20
1400 IF TY$="S" OR T2$="" THEN 1460
1410 T%=LEN(T2$)/2
1420 C$=STRING$(T%,8)
1430 IF TY$="L" THEN B$="PA4960,7135;LB" ELSE B$="PA9145,4200;LB"
1440 P$=B$+C$+T2$+CHR$(3):GOSUB 20
1450 '
1460 X%=LEN(X$)/2                                         'output x axis label
1470 A$="SI0.1959,0.2793;"
1480 IF TY$="S" THEN B$="PA4747,7142;LB"
1490 IF TY$="L" THEN B$="PA4747,6535;LB"
1500 IF TY$="P" THEN B$="PA8545,4413;LB"
1510 C$=STRING$(X%,8)
1520 P$=A$+B$+C$+X$+CHR$(3):GOSUB 20
1530 IF TY$="P" THEN A$="PA9210,7200;" ELSE A$="PA710,7200;"
1540 B$="UC-99,-13,7,99,2,8,-99,3,0,99,-4,-4,1,-4,2,0,2,3,-3,-7,4,6,"
```

```
1550 C$="1,1,1,0,1,-1,0,-1,-1,-1,-1,0,-1,1,-99,0,4,99,1,0,0,1,-1,0,0,-1;"
1560 P$=A$+B$+C$:GOSUB 20
1570 '
1580 Y%=LEN(Y$)/2                                    'output y axis label
1590 IF TY$="S" THEN B$="PA9080,4078;DI0,-1;LB"
1600 IF TY$="L" THEN B$="PA9080,3810;DI0,-1;LB"
1610 IF TY$="P" THEN B$="PA4425,1370;DI-1,0;LB"
1620 C$=STRING$(Y%,8)
1630 P$=B$+C$+Y$+CHR$(3)
1640 IF TY$="P" THEN P$=P$+"DI0,1;" ELSE P$=P$+"DI-1,0;"
1650 GOSUB 20
1660 X=XL:Y=YL                                       'draw axes
1670 GOSUB 3320
1680 P$="LT;PD":GOSUB 20
1690 X=XH
1700 GOSUB 3320
1710 Y=YH
1720 GOSUB 3320
1730 X=XL
1740 GOSUB 3320
1750 Y=YL
1760 GOSUB 3320
1770 P$="PU;":GOSUB 20
1780 S=XS:D=XD:H=XH:L=XL:T%=XT%:DD=DX:GOSUB 4030:XT%=T%  'calc X-axis tic values
1790 GOSUB 2660                              '      print X label
1800 IF XT%=0 THEN 1860
1810 FOR I%=1 TO XT%
1820 X=TI(I%):GOSUB 3320
1830 IF TY$="P" THEN GOSUB 2620 ELSE GOSUB 2610 '     tic #3 Port., #4 Land.
1840 GOSUB 2660                              '      print X label
1850 NEXT I%
1860 X=XH:GOSUB 3320
1870 GOSUB 2660                              '      print X label
1880 IF XT%=0 THEN 1940
1890 Y=YH
1900 FOR I%=1 TO XT%
1910 X=TI(I%):GOSUB 3320
1920 IF TY$="P" THEN GOSUB 2600 ELSE GOSUB 2630 '     tic #1 Port., #3 Land.
1930 NEXT I%
1940 GOTO 1960
1950 '
1960 S=YS:D=YD:H=YH:L=YL:T%=YT%:DD=DY:GOSUB 4030:YT%=T%  'calc Y-axis tic values
1970 IF TY$<>"P" THEN 2160                   'Landscape or Portrait ?
1980 X=XL:Y=YL:GOSUB 3320                    '   Portrait
1990 GOSUB 2800                              '      print Portrait Y label
2000 IF YT%=0 THEN 2060
2010 FOR I%=1 TO YT%
2020 Y=TI(I%):GOSUB 3320
2030 GOSUB 2630                              '      tic #4
2040 GOSUB 2800                              '      print Portrait Y label
2050 NEXT I%
2060 Y=YH:GOSUB 3320
2070 GOSUB 2800                              '      print Portrait Y label
2080 IF YT%=0 THEN 2140
2090 X=XH
```

```
2100 FOR I%=1 TO YT%
2110 Y=TI(I%):GOSUB 3320
2120 GOSUB 2610                                    '        tic #2
2130 NEXT I%
2140 GOTO 2330
2150 '
2160 X=XL:Y=YL:GOSUB 3320                          '     Landscape
2170 GOSUB 2740                                    '       print Landscape Y label
2180 IF YT%=0 THEN 2240
2190 FOR I%=1 TO YT%
2200 Y=TI(I%):GOSUB 3320
2210 GOSUB 2620                                    '        tic #3
2220 GOSUB 2740                                    '         print Landscape Y label
2230 NEXT I%
2240 Y=YH:GOSUB 3320
2250 GOSUB 2740                                    '         print Landscape Y label
2260 IF YT%=0 THEN 2330
2270 X=XH
2280 FOR I%=1 TO YT%
2290 Y=TI(I%):GOSUB 3320
2300 GOSUB 2600                                    '        tic #1
2310 NEXT I%
2320 '
2330 FRST=-1
2340 Y=STYLE:IF FI THEN 2360
2350 IF LNE THEN GOSUB 3560 ELSE GOSUB 3420
2360 IF EOF(2) THEN 2530          'start main loop
2370 INPUT #2,X,Y
2380 IF X<XL THEN GOSUB 3370:GOTO 2360
2390 IF NOT(LNE) THEN 2480
2400 IF NOT(FRST) THEN GOSUB 3320:GOTO 2360            '   first point?
2410 FRST=0                       '      yes, mark as 1st pt done
2420 P$="PU;"
2430 GOSUB 20
2440 GOSUB 3320                                    '        goto 1st pt
2450 P$="PD;"                                      '        pen down for subsequent pts
2460 GOSUB 20
2470 GOTO 2360
2480 P$="PU;":GOSUB 20                             '          point
2490 GOSUB 3320
2500 P$=PT$                                        '  get string of pt shape
2510 GOSUB 20
2520 GOTO 2360
2530 P$="PU;LT;SP0;PA710,700;"
2540 GOSUB 20
2550 CLOSE
2560 END
2570 '
2580 'these routines draw the tic marks, they are numbered to correspond to
2590 '       the sides of the final thesis page:
2600 P$="PD;PR80,0;PU;PR-80,0;":GOTO 20     'print tic #1, top
2610 P$="PD;PR0,-80;PU;PR0,80;":GOTO 20     'print tic #2  right
2620 P$="PD;PR-80,0;PU;PR80,0;":GOTO 20     'print tic #3  bottom
2630 P$="PD;PR0,80;PU;PR0,-80;":GOTO 20     'print tic #4  left
2640 'thus tic #3 is the on the Y axis in Landscape & on the X axis in Portrait
```

```
2650 '
2660 A=X                                  'print X axis value
2670 GOSUB 2860                           '  get string for value
2680 A=LEN(A$)/2
2690 IF A>2.5 THEN PRINT "X-axis value too large":END
2700 A=INT(A)
2710 IF TY$="P" THEN B$="PR0,-40;" ELSE B$="PR40,0;"
2720 P$=B$+"LB"+CHR$(10)+STRING$(A,8)+A$+CHR$(3):GOTO 20
2730 '
2740 A=Y                                  'print Landscape Y axis value
2750 GOSUB 2860                           '  get string for value
2760 P$="PR50,55;LB"
2770 A%=LEN(A$):IF A%>5 THEN PRINT "Y-axis value too large":END
2780 P$=P$+STRING$(A%,8)+A$+CHR$(3):GOTO 20
2790 '
2800 A=Y                                  'print Portrait Y axis value
2810 GOSUB 2860                           '  get string for value
2820 P$="PR55,-50;LB"
2830 A%=LEN(A$):IF A%>5 THEN PRINT "Y-axis value too large":END
2840 P$=P$+STRING$(A%,8)+A$+CHR$(3):GOTO 20
2850 '
2860 IF S THEN 3090                       'Linear axis?
2870 L=LOG(A)/LN+.001          'log axis
2880 E=INT(L)                       'get exponent
2890 M=L-E                          'get mantissa
2900 'if only one decade covered   then get first digit
2910 IF D<>1 THEN DI=INT(10^M):DI$=MID$(STR$(DI),2,1):GOTO 2980
2920 E=E-1:M=M+1:DI=INT(10^M)            'else set up to get first two digits
2930 '          if not 15*pwr of ten then only need first digit
2940 IF DI<>15 THEN DI=DI\10:E=E+1:DI$=MID$(STR$(DI),2,1):GOTO 2980
2950 IF E>-1 THEN DI$="15"     'else set up string to print
2960 IF E=-1 THEN DI$="1.5":E=0
2970 IF E<-1 THEN DI$="15":E=E+1
2980 IF E=-4 THEN A$=".000"+DI$     'add on zeros to get right pwr of ten
2990 IF E=-3 THEN A$="0.00"+DI$
3000 IF E=-2 THEN A$="0.0"+DI$
3010 IF E=-1 THEN A$="0."+DI$
3020 IF E=0 THEN A$=DI$
3030 IF E=1 THEN A$=DI$+"0"
3040 IF E=2 THEN A$=DI$+"00"
3050 IF E=3 THEN A$=DI$+"000"
3060 IF E=4 THEN A$=DI$+"0000"
3070 RETURN
3080 '
3090 'this gets a string(A$) from a value(A) rounded to DD decimal places
3100 IF (-10^(-DD)/2)>A OR A<(10^(-DD)/2) THEN 3120
3110 A$="0."+STRING$(DD,"0"):SIGN$="":GOTO 3300
3120 IF A>0 THEN A=A+(10^(-DD))/2 ELSE A=A-(10^(-DD))/2
3130 A$=STR$(A)
3140 SIGN$=MID$(A$,1,1):IF SIGN$=" " THEN SIGN$=""
3150 A$=MID$(A$,2)
3160 E%=INSTR(A$,"E")
3170 IF E%<>0 THEN B$=MID$(A$,1,E%-1):E%=VAL(MID$(A$,E%+1)) ELSE B$=A$
3180 DP%=INSTR(B$,".")
3190 IF DP%<>0 THEN A$=MID$(B$,1,DP%-1):B$=MID$(B$,DP%+1) ELSE A$=B$:B$=""
```

```
3200 IF E%=0 THEN 3280
3210 IF E%<0 THEN 3240
3220 IF E%<=LEN(B$) THEN A$=A$+MID$(B$,1,E%):B$=MID$(B$,E%+1):GOTO 3280
3230 A$=A$+B$+STRING$(E%-LEN(B$),"0"):B$="":GOTO 3280
3240 E%=-E%
3250 IF E%>LEN(A$) THEN 3270
3260 B$=MID$(A$,LEN(A$)-E%)+B$:A$=MID$(A$,1,LEN(A$)-E%-1):GOTO 3280
3270 B$=STRING$(E%-LEN(A$),"0")+A$+B$:A$=""
3280 IF A$="" THEN A$="0"
3290 A$=SIGN$+A$+"."+MID$(B$,1,DD)
3300 RETURN
3310 '
3320 IF XS THEN X1=XA*X+XB ELSE X1=XA*LOG(X)+XB          'go to a point
3330 IF YS THEN Y1=YA*Y+YB ELSE Y1=YA*LOG(Y)+YB
3340 IF TY$="P" THEN X2=X1:X1=Y1:Y1=X2                   'switch x & y values
3350 P$="PA"+STR$(INT(X1))+","+STR$(INT(Y1))+";"
3360 GOTO 20
3370 IF Y<>0 THEN GOTO 3400                             'change point or line style
3380 LINE INPUT "plotter needs attention, hit RETURN when done";A$
3390 RETURN
3400 IF Y<0 THEN GOTO 3550        'set up new point or line style
3410 LNE=0
3420 P$="LT;":GOSUB 20
3430 ON Y GOTO 3460,3470,3480,3490,3500,3520            'point pattern is in string
PT$
3440 '              circle,crescent,triangle,square, 5 ptd star, 6 ptd star
3450 PRINT "Error in point set parameter, value is ",Y:END
3460 PT$="CI30,30;":RETURN
3470 IF TY$="P" THEN PT$="PR0,40;PD;AR0,-40,-270,30;AR32,8,240,30;PU;":RETURN
ELSE PT$="PR-40,0;PD;AR40,0,-270,30;AR-8,32,240,30;PU;":RETURN
3480 IF TY$="P" THEN PT$="PR20,-36;PD0,72,-62,-36,62,-36;PU;":RETURN ELSE
PT$="PR-36,20;PD72,0,-36,-62,-36,62;PU;":RETURN
3490 PT$="PR-24,-24;PD48,0,0,48,-48,0,0,-48;PU;":RETURN
3500 IF TY$="P" THEN
PT$="PR40,-28;PD-32,10,-22,-28,0,36,-32,10,32,10,0,36,22,-28,32,10;" ELSE
PT$="PR-28,40;PD10,-32,-28,-22,36,0,10,-32,10,32,36,0,-28,22,10,32;"
3510 IF TY$="P" THEN PT$=PT$+"PD-20,-28,20,-28;PU;":RETURN ELSE
PT$=PT$+"PD-28,-20,-28,20;PU;":RETURN
3520 IF TY$="P" THEN PT$="PR-48,0;PD24,-14,0,-26,24,12,24,-12,0,26,24,14;"
ELSE PT$="PR0,-48;PD-14,24,-26,0,12,24,-12,24,26,0,14,24;"
3530 IF TY$="P" THEN PT$=PT$+"PD-24,14,0,26,-24,-12,-24,12,0,-26,-24,-14;PU;"
ELSE PT$=PT$+"PD14,-24,26,0,-12,-24,12,-24,-26,0,-14,-24;PU;"
3540 RETURN
3550 LNE=-1:FRST=-1:Y=-Y
3560 ON Y GOTO 3580,3590,3600,3610,3620,3630,3640
3570 PRINT "Error in line set parameter, value is ",Y:END
3580 P$="LT;":GOTO 20
3590 P$="LT2;":GOTO 20
3600 P$="LT1,1;":GOTO 20
3610 P$="LT4;":GOTO 20
3620 P$="LT5;":GOTO 20
3630 P$="LT6;":GOTO 20
3640 INPUT #2,X,Y
3650 P$="PU;":GOSUB 20
3660 GOSUB 3320
```

```
3670 LINE INPUT #2,A$
3680 P$="CP.33,-.25;LB"+A$+CHR$(3):GOTO 20
3690 '
3700 ' check the axis labels, must fit in 5 character positions
3710 IF S THEN 3780
3720 A$=S$+" axis values cannot fit into 5 characters"
3730 D=INT(LOG(H/L)/2.303+.001)          'no. of decades
3740 IF D=1 AND L<.001 THEN PRINT A$:END
3750 IF D<>1 AND L<.0001 THEN PRINT A$:END
3760 IF H>10000 THEN PRINT A$:END
3770 GOTO 4010
3780 A$="-axis divisions, "
3790 IF ABS(D)<.0001 THEN PRINT S$;A$;D;" are too small":END
3800 IF D>10000 OR D<-1000 THEN PRINT A$;D;" are too big":END
3810 DD=-INT(LOG(ABS(D))/2.303):IF DD<0 THEN DD=0
3820 A$="-axis values cannot fit into 5 characters"
3830 IF L=0 THEN GOTO 3910
3840 LL=LOG(ABS(L))/2.303
3850 IF L<0 THEN GOTO 3890
3860 IF DD<>0 AND LL+DD>4 THEN PRINT S$;A$:END
3870 IF DD= 0 AND LL    >5 THEN PRINT S$;A$:END
3880 GOTO 3910
3890 IF DD<>0 AND LL+DD>3 THEN PRINT S$;A$:END
3900 IF DD= 0 AND LL    >4 THEN PRINT S$;A$:END
3910 IF H=0 THEN GOTO 3990
3920 LL=LOG(ABS(H))/2.303
3930 IF H<0 THEN GOTO 3970
3940 IF DD<>0 AND LL+DD>4 THEN PRINT S$;A$:END
3950 IF DD= 0 AND LL    >5 THEN PRINT S$;A$:END
3960 GOTO 3990
3970 IF DD<>0 AND LL+DD>3 THEN PRINT S$;A$:END
3980 IF DD= 0 AND LL    >4 THEN PRINT S$;A$:END
3990 T%=(H-L)/D-1
4000 IF T%>15 THEN PRINT "Too many divisions along ";S$;"-axis (15 allowed)":END
4010 RETURN
4020 '
4030 ' axis values calculation
4040 IF S THEN 4450
4050 ON D GOTO 4070,4130,4210,4310,4400,4400,4400,4400
4060 PRINT "Error in axis scaling":END
4070 T%=9                                'one decade
4080 TI(1)=1.5*L
4090 FOR I%=2 TO 9
4100 TI(I%)=I%*L
4110 NEXT I%
4120 RETURN
4130 T%=9                                'two decades
4140 FOR I%=1 TO 5
4150 TI(I%)=I%*2*L
4160 NEXT I%
4170 FOR I%=6 TO 9
4180 TI(I%)=(I%-5)*20*L
4190 NEXT I%
4200 RETURN
4210 T%=8                                'three decades
```

```
4220 TI(1)=2*L
4230 TI(2)=4*L
4240 TI(3)=10*L
4250 TI(4)=20*L
4260 TI(5)=40*L
4270 TI(6)=100*L
4280 TI(7)=200*L
4290 TI(8)=400*L
4300 RETURN
4310 T%=7                                'four decades
4320 TI(1)=3*L
4330 TI(2)=10*L
4340 TI(3)=30*L
4350 TI(4)=100*L
4360 TI(5)=300*L
4370 TI(6)=1000*L
4380 TI(7)=3000*L
4390 RETURN
4400 T%=D-1                              'five+ decades
4410 TI(1)=10*L
4420 FOR I%=2 TO T%
4430 TI(I%)=10*TI(I%-1)
4440 NEXT I%
4450 RETURN
4460 FOR I%=1 TO T%                      'linear scale
4470 TI(I%)=I%*D+L
4480 NEXT I%
4490 RETURN
4500 '****************************************************************************
4510 '* Screen Graph section, adapted from                                      *
4520 '*                        LINEAR PLOT PROGRAM                               *
4530 '*  MetraByte Corporation                    Rev 1.10    8/16/83           *
4540 '*  modified 15-Oct-84 by Peter Kip Mercure
4550 '****************************************************************************
4560 '
4570 '    This section is where the starting graph coordinates and screen scale
4580 'ratio is set. OX,OY = starting point of graph. QX,QY = screen scale x:y.
4590 '
4600 KEY OFF:SCREEN 0,0,0
4610 OX=70:OY=0:QX=1.8:QY=.9000001
4620 '
4630 '
4640 '   This is the main link to the plotting program.  The file RLINPLT.LNK
4650 'has the file name and disk of the file(s) to be plotted.
4660 '
4670 '
4680 '
4690 '   This sets the X,Y Max/Min values
4700 '
4710 IF XS THEN X1=XL ELSE X1=LOG(XL)
4720 IF XS THEN X2=XH ELSE X2=LOG(XH)
4730 IF YS THEN Y1=YL ELSE Y1=LOG(YL)
4740 IF YS THEN Y2=YH ELSE Y2=LOG(YH)
4750 '
4760 '     This section establishes the actual scale factors used to plot the
```

```
4770 'data as the files are read. These values are also the same values saved
4780 'on the disk when a Save command is executed.
4790 '
4800 CLS:COLOR 7,0,0:SCREEN 2
4810 DX= ABS(CINT((X2-X1)/25+.5))
4820 DY = ABS(CINT(((Y2-Y1)/25)+.5))
4830 SX=260 / (X2-X1)
4840 SY=140 /(Y2-Y1)
4850 '
4860 '    This section locates the Axis x,y and plots them out with the small
4870 'tick marks identifing the scale. The axis labels are also printed.
4880 '
4890 '
4900 IF Y2<=0 THEN YA=10:GOTO 4940
4910 IF Y1=>0 THEN YA=150:GOTO 4940
4920 YA=10 + SY * Y2
4930 '
4940 'set the range limits
4950 '
4960 IF X2 <= 0 THEN XA=270:GOTO 4990
4970 IF X1 => 0 THEN XA=10:GOTO 4990
4980 XA=10 - SX * X1
4990 FOR YLBL = 1 TO 16
5000   LOCATE 2+YLBL,6:PRINT MID$(Y$,YLBL,1);
5010 NEXT YLBL
5020 LOCATE 1,1:PRINT USING "#.##^^^^";Y2;
5030 LOCATE 19,1:PRINT USING "#.##^^^^";Y1;
5040 LOCATE 20,10:PRINT USING "#.##^^^^";X1;
5050 LOCATE 20,68:PRINT USING "#.##^^^^";X2;
5060 LINE (QX*XA+OX,QY*10+OY)-(QX*XA+OX,QY*150+OY)
5070 LINE (QX*10+OX,QY*YA+OY)-(QX*270+OX,QY*YA+OY)
5080 '
5090 ' Mark the x axis ticks
5100 '
5110 K=0
5120 B=YA-2
5130 C=YA+2
5140 K=K+1
5150 A=DX * K
5160 AA=XA+SX*A
5170 IF AA>271 GOTO 5200
5180 LINE (QX*AA+OX,QY*B+OY)-(QX*AA+OX,QY*C+OY)
5190 GOTO 5140
5200 K=0
5210 K=K+1
5220 A=DX*K
5230 AA=XA-SX*A
5240 IF AA<9 GOTO 5300
5250 LINE (QX*AA+OX,QY*B+OY)-(QX*AA+OX,QY*C+OY)
5260 GOTO 5210
5270 '
5280 ' Mark the y axis ticks
5290 '
5300 K=0
5310 A=XA-2
```

```
5320 C=XA+2
5330 K=K+1
5340 B=DY*K
5350 BB=YA-SY*B
5360 IF BB<9 GOTO 5390
5370 LINE (QX*A+OX,QY*BB+OY)-(QX*C+OX,QY*BB+OY)
5380 GOTO 5330
5390 K=0
5400 K=K+1
5410 B=DY*K
5420 BB=YA+SY*B
5430 IF BB>151 GOTO 5460
5440 LINE (QX*A+OX,QY*BB+OY)-(QX*C+OX,QY*BB+OY)
5450 GOTO 5400
5460 LOCATE 22,1:PRINT T1$:PRINT T2$
5470 LOCATE 20,22:PRINT X$;:LOCATE 23,1
5480 '
5490 '   This section plots the points one by one.
5500 '
5510 FRST=-1
5520 IF EOF(2) THEN 5720       'main loop
5530 INPUT #2,X,Y
5540 IF NOT(XS) THEN X=LOG(X)
5550 IF NOT(YS) THEN Y=LOG(Y)
5560 IF X>=XL THEN 5670
5570 FRST=-1                   'mark first point of a series
5580 IF Y=0 THEN 5520
5590 IF Y<>-7 THEN 5650
5600 INPUT #2,X,Y
5610 GOSUB 90:IF OF THEN 5520
5620 CIRCLE (QX*AA+OX, QY*BB+OY),4
5630 LINE INPUT #2,A$
5640 GOTO 5520
5650 IF Y<0 THEN LNE=-1 ELSE LNE=0    'if y<0 then draw lines
5660 GOTO 5520
5670 GOSUB 90:IF OF THEN 5520
5680 IF NOT(LNE) THEN 5710         'plot data
5690 IF FRST THEN FRST=0:GOTO 5710
5700 LINE-(QX*AA+OX,QY*BB+OY):GOTO 5520 'line plot
5710 PSET (QX*AA+OX,QY*BB+OY):F=0:GOTO 5520    'dot plot
5720 IF INKEY$="" THEN GOTO 5720
5730 END
```

Listing 27 Program for RECURSive parameter estimation in FORtran.

```fortran
$STORAGE:2
$NOFLOATCALLS
C
C  Main program for recursive parameter estimation
C.    inputs data values and outputs parameter estimates
C
      INTEGER N
      REAL*8 P(2,2),THETA(2),PHI(2),LAMBDA
     &       ,TEMP,TIME,TINF,K
      N=2
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  I      INTEGER*2     counter
C  J      INTEGER*2     counter
C  K      REAL*8        rate constant (1/sec)
C  LAMBDA REAL*8        weighting factor
C  N      INTEGER*2     dimension of estimator
C  P      REAL*8        covariance matrix
C  PHI    REAL*8        vector of independent variables
C  TEMP   REAL*8        observed variable (Temperature)
C  THETA  REAL*8        parameter vector
C  TIME   REAL*8        time of observation
C  TINF   REAL*8        Temperature at infinite time
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      OPEN(3,FILE='CON',STATUS='NEW')
      OPEN(4,FILE='CON',STATUS='OLD')
C
C  init variables
C
      DO 10 I=1,N
         DO 20 J=I,N
            IF (I.EQ.J) THEN
                 P(I,I)=1.D8
               ELSE
                 P(I,J)=0.D0
                 P(J,I)=P(I,J)
            ENDIF
 20      CONTINUE
 10   CONTINUE
      THETA(1)=0.9D0
      THETA(2)=0.4D0
      LAMBDA=1.D0
      PHI(2)=1.D0
      TEMP=90.D0
 40   CONTINUE
         PHI(1)=TEMP
         READ(4,30)TIME,TEMP
 30      FORMAT(2G11.4)
         WRITE(3,50)TIME,TEMP
 50      FORMAT(1X,'t=',G11.4,' T=',G11.4)
```

```
          CALL RECURS(N,P,THETA,PHI,TEMP,LAMBDA)
          K=-LOG(THETA(1))/0.1D0
          TINF=THETA(2)/(1.D0-THETA(1))
          WRITE(3,60)K,TINF
   60     FORMAT(1X,G11.4,',',G11.4)
          GOTO 40
      STOP
      END

      SUBROUTINE RECURS(N,P,THETA,PHI,OBS,LAMBDA)
      INTEGER N,I,J,K
      REAL*8 P(N,N),THETA(N),PHI(N),OBS,LAMBDA
     &       ,ERR,PPHI(2),D,L(2),W(2,2)

CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
C
C  D       REAL*8       denominator term (phi(trans)*P*phi + 1/lambda)
C  ERR     REAL*8       error term (y - theta*phi)
C  I       INTEGER*2    counter
C  J       INTEGER*2    counter
C  K       INTEGER*2    counter
C  L       REAL*8       gain vector
C  LAMBDA  REAL*8       weighting factor
C  N       INTEGER*2    dimension of estimator
C  OBS     REAL*8       observation (measurement)
C  P       REAL*8       covariance matrix
C  PHI     REAL*8       vector of independent variables
C  PPHI    REAL*8       P*phi
C  THETA   REAL*8       parameter vector
C  W       REAL*8       temporary storage (I-L*phi)P
C
CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

C
C  err = obs - theta*phi
C
      ERR=OBS
      DO 10 I=1,N
          ERR = ERR - THETA(I)*PHI(I)
   10     CONTINUE
          WRITE(3,1000)ERR
 1000     FORMAT(1X,'ERR=',G11.4)
C
C  pphi = p*phi
C
      DO 20 I=1,N
      PPHI(I) = 0.D0
          DO 30 J=1,N
              PPHI(I) = PPHI(I)+P(I,J)*PHI(J)
   30         CONTINUE
          WRITE(3,1010)I,PPHI(I)
 1010     FORMAT(1X,'PPHI(',I1,')=',G11.4)
   20     CONTINUE
C
C  d = phi(transpose)*p*phi + 1/lambda
```

```
C
      D=1.D0/LAMBDA
      DO 40 I=1,N
         D=D+PHI(I)*PPHI(I)
 40      CONTINUE
         WRITE(3,1020)D
 1020    FORMAT(1X,'D=',G11.4)
C
C  I = pphi/d
C  theta(k+1) = theta(k) + I*err
C
      DO 50 I=1,N
         L(I)=PPHI(I)/D
         THETA(I)=THETA(I) + L(I)*ERR
         WRITE(3,1030)I,L(I),I,THETA(I)
 1030    FORMAT(1X,'L(',I1,')=',G11.4,' THETA(',I1,')=',G11.4)
 50      CONTINUE
C
C  w = I - I*phi(transpose)
C
      DO 60 I=1,N
         DO 70 J=I,N
            IF (I.EQ.J) THEN
                W(I,I)=1.D0 - L(I)*PHI(I)
               ELSE
                W(I,J)= - L(I)*PHI(J)
                W(J,I)= W(I,J)
               ENDIF
 70         CONTINUE
 60      CONTINUE
C
C  p(k+1) = w*p(k)
C    using pphi for temporary storage
C
         WRITE(3,1040)P(1,1),P(1,2),P(2,1),P(2,2)
 1040    FORMAT(1X,'P:',2(G11.4,' ')
     &          /1X,'    ',2(G11.4))
         WRITE(3,1050)W(1,1),W(1,2),W(2,1),W(2,2)
 1050    FORMAT(1X,'W:',2(G11.4,' ')
     &          /1X,'    ',2(G11.4))
      DO 80 I=1,N
         DO 90 J=1,N
            PPHI(J)=0.0D0
            DO 100 K=1,N
               PPHI(J)=PPHI(J)+W(J,K)*P(K,I)
 100           CONTINUE
 90         CONTINUE
         DO 110 J=1,N
            P(J,I)=PPHI(J)
 110        CONTINUE
 80      CONTINUE
      DO 120 I=1,N
         DO 130 J=I,N
            IF(I.NE.J)THEN
C       take symmetry of P into account to compensate for roundoff errors
```

```
                P(I,J)=(P(I,J)+P(J,I))/2.D0
                P(J,I)=P(I,J)
             ENDIF
130       CONTINUE
120    CONTINUE
       WRITE(3,1040)P(1,1),P(1,2),P(2,1),P(2,2)
C
       RETURN
       END
```

Colophon

The text for this book was produced with an Apple Macintosh computer
using the standard MacPaint and MacWrite applications. Figures which
were smaller than a page were imbedded within the text file in MacWrite.
Most of the text was printed in a scientific font called Princeton (D. E.
Doughterty and S. H. Lam, Department of Mechanical & Aerospace
Engineering, Princeton University, Princeton, New Jersey 08544). Some of
the accented characters (ü, ø, é) were done in the standard Apple supplied
Geneva font. The computer listings and other text which required a
non-proportional font were done in a font that the author modified from
the standard Apple supplied Monaco font; the modifications allowed
certain characters to be distinguished more easily (a sequence of
characters in Monaco: 1 l I 0 0, the same sequence of characters in
Blacksburg: 1 l I 0 0). Most of the output was on an Imagewriter printer
in high quality mode. The listings, some of the figures, and Chapters 7
and 8 were done on an Apple LaserWriter. Some of the small graphs were
produced using Microsoft MacChart. The full page graphs were produced on
an HP7470A plotter using a program by the author (listing 26 in the
appendix).