1970-12

Modulation of reactor power in the Pennsylvania State TRIGA reactor (PSTR) as a research tool

Cross, Lawrence A., Jr.

Philadelphia, Pennsylvania; Pennsylvania State University

MODULATION OF REACTOR POWER IN THE PENNSYLVANIA STATE TRIGA REACTOR (PSTR) AS A RESEARCH TOOL

By

Lawrence A. Cross

Modulation of Reactor Power in the Pennsylvania State TRIGA Reactor (PSTR) as a Research Tool

by

Lawrence A. Cross, Jr.

The Pennsylvania State University

The Graduate School

Department of Nuclear Engineering

Modulation of Reactor Power in the Pennsylvania State TRIGA Reactor (PSTR) as a Research Tool

A Paper in

Nuclear Engineering

by

Lawrence A. Cross, Jr.

Submitted in partial fulfillment
of the requirements
for the degree of

Master of Science

December 1970

ACKNOWLEDGMENTS

The author wishes to express his appreciation to Professor E. S. Kenney for his initiation of the project and for his assistance during the research.

This research and the author's course of study were made possible under the Naval Postgraduate Education Program.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

I. INTRODUCTION

The utilization of pulsed research reactors has been primarily in the single pulse mode. That is, pulsed reactor research data is generally obtained from a single exposure of an experiment to a pulse of high neutron flux. This mode of operation of pulsed reactors has been fruitful in many research areas. Miley (23) has compiled an extensive summary of the areas of research where TRIGA reactors have been utilized.

A new frontier in the utilization of pulsed research reactors has been recognized in the repetitive pulse mode of operation. Advances in this area began with the successful operation of a periodically pulsed fast reactor in the USSR in 1960. This prototype of future periodically pulsed reactors was the IBR reactor at Dubna. The operating experience gained with the IBR reactor was reported by Bunin (6) in 1964 and more recently by Anan'ev (2) in 1969. As a result of the early reports of successful operation of the IBR reactor, there was a rise of interest in the concept of repetitively pulsed reactors, which was evident at several conferences that followed the 1964 report: the Symposium on Pulsed Neutron Research at Karlsruhe in 1965 (17), the Symposium on Pulsed High Intensity Fission Neutron Sources at Washington in 1965 (32), the Seminar on Intense Neutron Sources at Sante Fe in 1966 (30).

Following the successful operation of the IBR reactor at Dubna, several other designs for repetitively pulsed reactors appeared, notably, the SORA reactor at Ispra (22) and the Brookhaven National Laboratory repetitively pulsed reactor (14). The SORA and Brookhaven reactor designs differ from the IBR reactor in the design of the reactivity pulsing mechanism. All three reactors use a revolving rotor

which passes in close proximity to the reactor core to cause a pulsed reactivity. In the IBR reactor, the rotor tip contains a fuel segment, while in the SORA and Brookhaven designs, the rotor contains a reflector material. At Dubna, a later design for the IBR-2 reactor also uses a moving reflector section for the pulsing mechanism. Several heat transfer problems were encountered with the moving fuel segment in the earlier design. The IBR reactor has been operational since 1960, while the SORA and Brookhaven reactors are still in the design stages. The experience gained from the first IBR reactor has been incorporated in the more advanced IBR-30 and in the design plans for the IBR-2. Technical data for these reactors is summarized in Table 1.

Table 1

Repetitively Pulsed Fast Reactors

	IBR-30 (2)	IBR-2 (2)	SORA (22)	Brookhaven (14)
Pulse repetition rate (seconds^{-1})	0 to 100	5, 10, 25, 50	50	60
Mean thermal power (megawatts)	0.030	4	0.6	30
Peak power (megawatts)	150[1]	7700[1] 700[2]	160	4700
Ratio of peak to background power	--	35,000[1] 3,200[2]	1600	--
Width of power pulse at half peak power (microseconds)	50	90[1]	56	90
Prompt neutron generation time (nanoseconds)	10	42	20	50

[1]Pulse repetition rate of 5 pulses/second

[2]Pulse repetition rate of 50 pulses/second

The repetitively pulsed reactors discussed above are all fast reactors with prompt neutron generation times on the order of 10 to 50 nanoseconds. In contrast, the prompt neutron generation time in a thermal reactor is about 40 microseconds. Very little research has been reported on the possibility of using thermal reactors in a repetitively pulsed mode.

Whittemore (37) has proposed a TRIGA type reactor design which could be used to produce repetitive pulses. The design uses an absorbing material mounted on a rotating disk in the centerline plane of a modified cylindrical TRIGA core. The modifications to the core, in addition to the reactivity pulsing mechanism, are smaller diameter fuel elements and forced water cooling, rather than the natural convective cooling in a standard TRIGA reactor. Whittemore made a preliminary study of the characteristics of this reactor design using the BLOOST digital computer code. The results of that study are shown in Table 2.

Table 2

Repetitively Pulsed Modified TRIGA Reactor Design (Whittemore) (37)

Pulse repetition rate	50 pulses/second
Mean thermal power	7 megawatts
Peak power	40 megawatts
Ratio of peak to background power	67
Width of power pulse at half peak power	3 milliseconds
Prompt neutron generation time	12 microseconds

Kurstedt (20) has experimentally determined the response of a standard TRIGA reactor to repetitive pulses of reactivity at frequencies up to 3.3 pulses per minute. He found that the heat generated in the

first pulse in a series strongly effected the response of the reactor to the sequential pulses. At the time of the second pulse, the reactor temperature is elevated above ambient due to the heat from the first pulse. The result is that the total reactivity in the second pulse from the combined effects of the transient rod and the residual negative reactivity due to temperature is much less than the reactivity in the first pulse from the transient rod alone. Therefore, the maximum power in the second pulse is less and the resulting residual negative reactivity due to temperature at the third pulse is less. In a series of pulses, the result of these effects is a damped oscillation of the values of total reactivity in a pulse, as shown in Figure 1, for a series of $2.00 reactivity pulses. The resulting equilibrium peak power, or peak power after the damped oscillations had died out is shown in Figure 2 for various pulse repetition rates.

There is an important difference in the repetitively pulsed reactors studied by Whittemore and Kurstedt in the effect which terminates a power pulse in the two reactors. In the low frequency (Kurstedt) study, the pulse is terminated by the large negative temperature coefficient of reactivity in a TRIGA reactor. As a result, each pulse in a series of pulses effects the following pulse by producing a residual negative reactivity due to temperature. In the high frequency (Whittemore) study, the power pulse is not terminated by thermal effects. The rate of change of reactor power is determined entirely by whether the prompt reactivity[1] in the pulse is greater than, or less than, zero. The peak power level in a pulse occurs when the prompt reactivity equals zero. The result

---

[1] Prompt reactivity is the total reactivity minus the delayed neutron fraction.

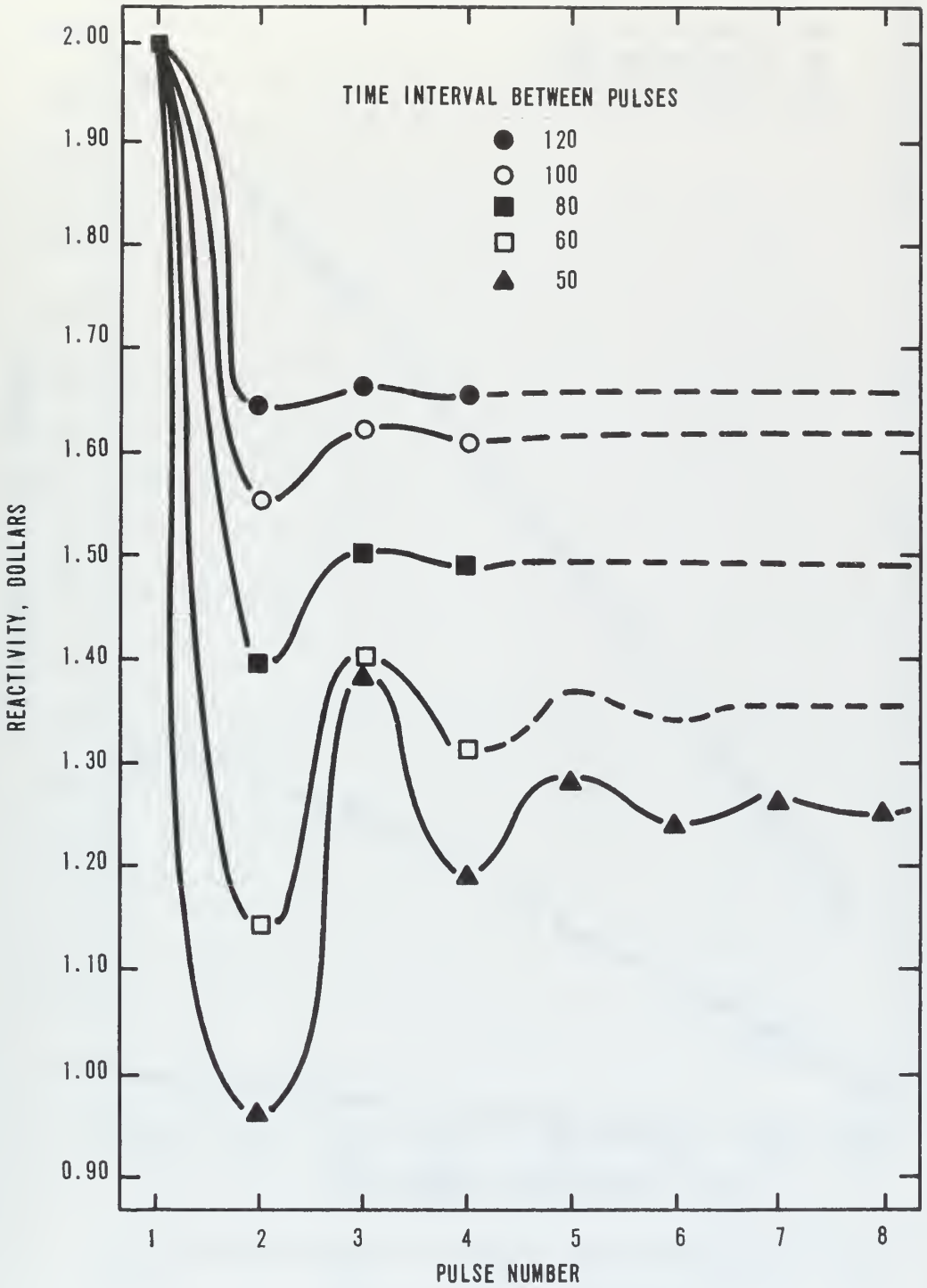

Fig. 1 Series Pulse Reactivities for Various Time Intervals between Pulses

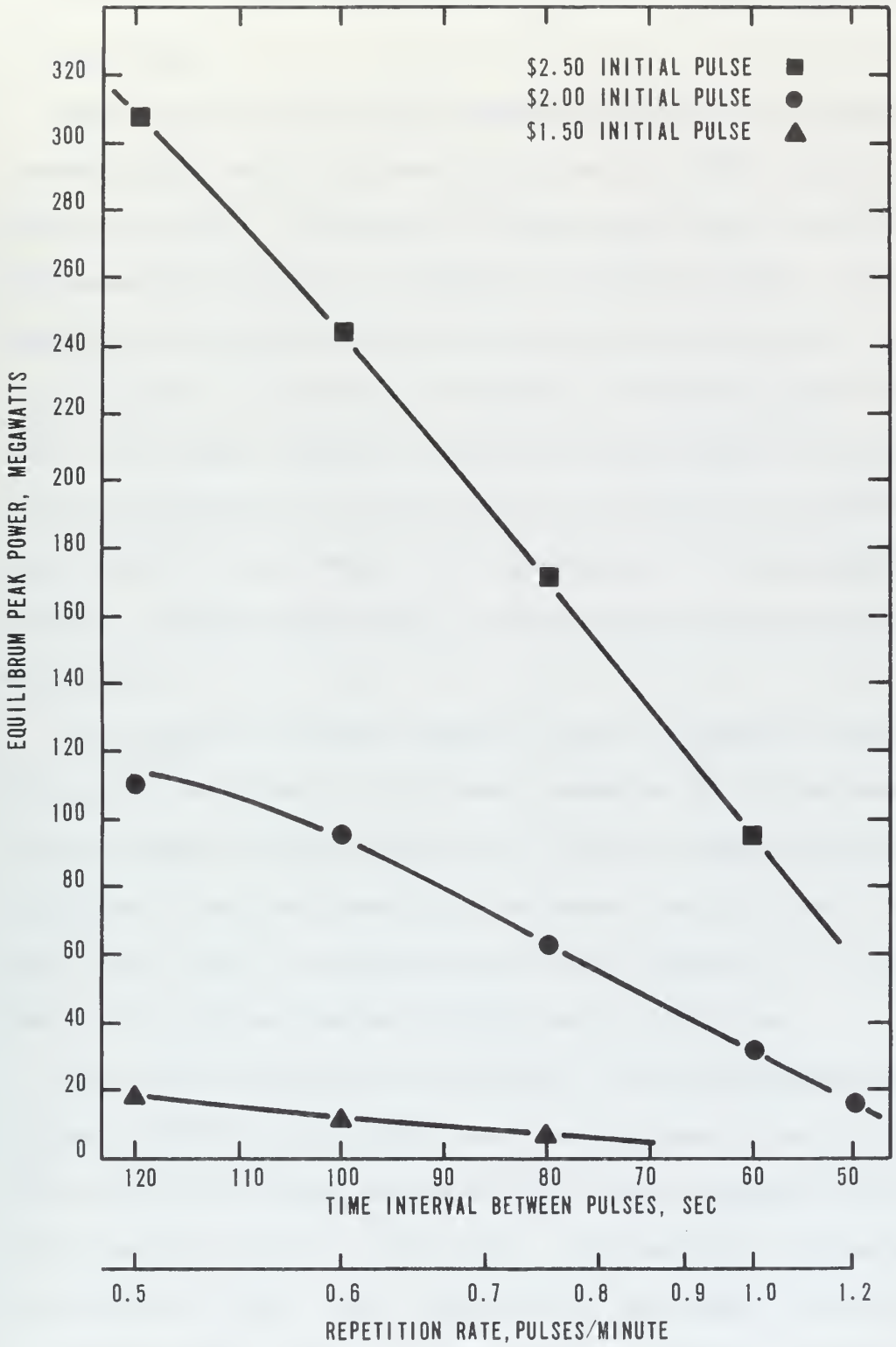

Fig. 2 The Effect of Repetition Rate on Equilibrium Peak Power

of this effect is that each pulse in a series is independent of the previous pulses.

The study of Kurstedt (20) thoroughly describes the response of a standard TRIGA reactor for reactivity pulse repetition rates up to 3.3 pulses per minute. The study of a TRIGA reactor design proposed by Whittemore (37) suggests the response of a modified TRIGA reactor for reactivity pulse repetition rates up to 50 pulses per second. Neither study describes the response of a reactor to reactivity pulse repetition rates in the range between 3.3 pulses per minute and 50 pulses per second. The objective of this report was to study the response of a TRIGA reactor in this range, and to investigate the feasibility of operating a standard TRIGA reactor in the frequency range considered by Whittemore.

The response of thermal reactors to small sinusoidal reactivities at very low power levels is adequately described by the zero-power reactor transfer function (18,24,26,31,34). For large reactivity oscillations at low power levels the zero-power reactor transfer function gives large errors in predicting the reactor response (3). At high power levels thermal feedback effects must be considered.

The reactor considered in this study is The Pennsylvania State University TRIGA Reactor (PSTR) with a modification to include a rotating reactivity pulsing mechanism rather than a pneumatically operated transient reactivity control rod. The reactivity pulsing mechanism was proposed by Kenney (19), independent of the similar design proposed by Whittemore (37). The reactivity pulse is produced by absorber material mounted on a circular disk rotating in the centerline plane of

the reactor. The disk is assumed to have a radius equal to the radius of the reactor core and to extend into the reactor so that the disk edge is tangent to the reactor centerline.

In order to study the pulsed behavior of the PSTR, a digital computer program was developed which may have practical use beyond this report. The program was specifically designed to simulate the pulsed response of a reactor to either a single pulse or a series of pulses. The objective in writing the program was to provide a simple program with generality and flexibility. The program is written completely in elementary FORTRAN IV and requires no knowledge of computer job control language (JCL). The extent of the programming experience required to thoroughly understand the program is equivalent to a first course in digital computer programming. The program description in Appendix A is sufficient for practical application of the program without previous programmming experience. Possible applications of the program are in simulating the pulsed response of the PSTR while planning, or in conjunction with, laboratory courses.

II. REACTOR MODEL

A. Point Reactor Kinetics Model

In the analysis of the kinetic behavior of nuclear reactors the point reactor kinetics model is often used. The spatial effects of the reactor are assumed to be negligible, or at least small enough to not effect the transient behavior of the reactor.

As might be expected, the point reactor kinetics model provides an accurate simulation of the kinetic behavior of small reactors. The model is also used for intermediate and large reactors in which the error introduced by spatial effects is minor relative to the desired results.

The spatial effects of the PSTR have been examined in two previous studies. Pallette (24) studied the space dependence of the zero-power reactor transfer function. He found that the spatial effects were not detectable at frequencies below 100 cycles per second. The theoretical results of that study predicted significant spatial effects beginning at about 50 cycles per second. Sears (29) studied the spatial behavior of the PSTR during prompt critical pulse transients. He determined that there was measurable differences in the neutron density profiles for steady state operation and for pulse operation. This effect is discussed in more detail in Section II-C.

The results of these previous studies indicate that the validity of the point reactor kinetics model is limited, although the results would be approximately correct over a wide range of reactor power and temperature.

B. Reactor Kinetics Equations

The equations describing the point reactor kinetics model appear in several forms in the literature. The form of the equations used here follows Keepin (18). In the one group space averaged approximation,

$$\frac{dn}{dt} = \frac{k(1-\bar{\gamma}\beta)-1}{\ell^*} n + \sum_{i=1}^{6} \lambda_i \gamma_i C_i'$$

$$\frac{dC_i'}{dt} = \frac{\beta k}{\ell^*} n - \lambda_i C_i'$$

where

n = neutron density (n/cm^3)

k = neutron reproduction factor

ℓ^* = prompt neutron lifetime (μsec)

γ_i = effectiveness (in producing fission) of ith group delayed neutrons relative to prompt neutrons

$\bar{\gamma}$ = average delayed neutron effectiveness

β_i = ith group delayed neutron yield fraction

β = total delayed neutron yield fraction

λ_i = ith delayed neutron group decay constant (sec^{-1})

C_i' = ith delayed neutron group precursor density ($atoms/cm^3$)

and where the source is neglected. This is the "neutron destruction formulation" in terms of the prompt neutron lifetime ℓ^*. If the prompt neutron generation time Λ is defined as

$$\Lambda = \frac{\ell^*}{k}$$

the kinetics equations can be expressed in the "neutron production formulation",

$$\frac{dn}{dt} = \frac{\rho - \bar{\gamma}\beta}{\Lambda} n + \sum_{i=1}^{6} \lambda_i \gamma_i C_i'$$

$$\frac{dC_i'}{dt} = \frac{\beta_i n}{\Lambda} - \lambda_i C_i'$$

where

$$\rho = \frac{k-1}{k} = \text{reactivity}$$

Although the two formulations are equivalent one may be more practical than the other for a particular problem. According to Keepin (18), in reactors controlled by "black" absorbers, neutron absorption and, hence, ℓ^* varies, while the production and, hence, Λ may be considered relatively constant. Conversely, in fast metal systems control may be by fuel displacement with little change in absorption; in this case, the production and, hence, Λ is varied, and ℓ^* is more nearly a constant parameter. For this reason, the "neutron production formulation", with constant Λ, was chosen for this study. Also, in this form of the kinetics equations, the reactivity ρ appears explicitly and, therefore, this form of the equations is more convenient for describing reactivity feedback effects.

Certain manipulations will be performed on these equations to simplify their solution. Define the effective delayed neutron fraction $\bar{\beta}$ as

$$\bar{\beta} = \bar{\gamma}\beta = \sum_{i=1}^{6} \gamma_i \beta_i$$

For the PSTR, $\bar{\beta} = 0.00700$ and

$$\bar{\gamma} = \frac{\bar{\beta}}{\beta} = 1.0766$$

In general, the delayed group effectiveness values γ_i are difficult to obtain. Keepin (18) suggests that in the absence of reliable data for the γ_i values the approximation be made

$$\gamma_i \approx \bar{\gamma}$$

Define the effective delayed neutron group precursor density[1] C_i as

$$C_i = \gamma_i C_i' = \bar{\gamma} C_i'$$

Then the "neutron production formulation" of the kinetics equations is

$$\frac{dn}{dt} = \frac{\rho - \bar{\beta}}{\Lambda} n + \sum_{i=1}^{6} \lambda_i C_i$$

$$\frac{dC_i}{dt} = \frac{\bar{\gamma}\beta_i}{\Lambda} n - \lambda_i C_i$$

---

[1] A common notation for the effective delayed neutron group precursor density in the literature is $C_{i_{eff}}$.

C. Point Reactor Temperature Model

The thermal effects in a nuclear reactor can be studied with either of two heat transfer models; a multi-region model in which the heat balance is established among the regions, or a point model in which the heat transfer effects are described by gross averaged parameters. Both of these models have been used in previous kinetic studies of TRIGA type reactors.

One difficulty in developing a multi-region heat transfer model is the extreme temperature gradient across a fuel element during pulsing. As a result, temperature measurements made by thermocouples at any point in the reactor core are strongly dependent on the location of the thermocouple within a fuel element. In addition, Wyman (38) determined that the temperature profile within a fuel element changes dramatically during a pulse. In that study, he used an instrumented TRIGA fuel element containing seven small thermocouples positioned at various radii in the fuel but all in the central plane of the fuel element. Temperature data from that study for a $2.00 reactivity insertion are shown in Figures 3 and 4. Similar results obtained by West (36) using the RAT heat transfer digital computer code are shown in Figure 5.

Another difficulty in developing a multi-region heat transfer model is that the neutron density profile across the reactor core also changes during a pulse. Sears (29) studied the spatial behavior of the PSTR during pulsing. He found differences between the profiles for steady state operation and for pulse operation. He attributed these differences to the heating of the water during steady state operation resulting in profile flattening.

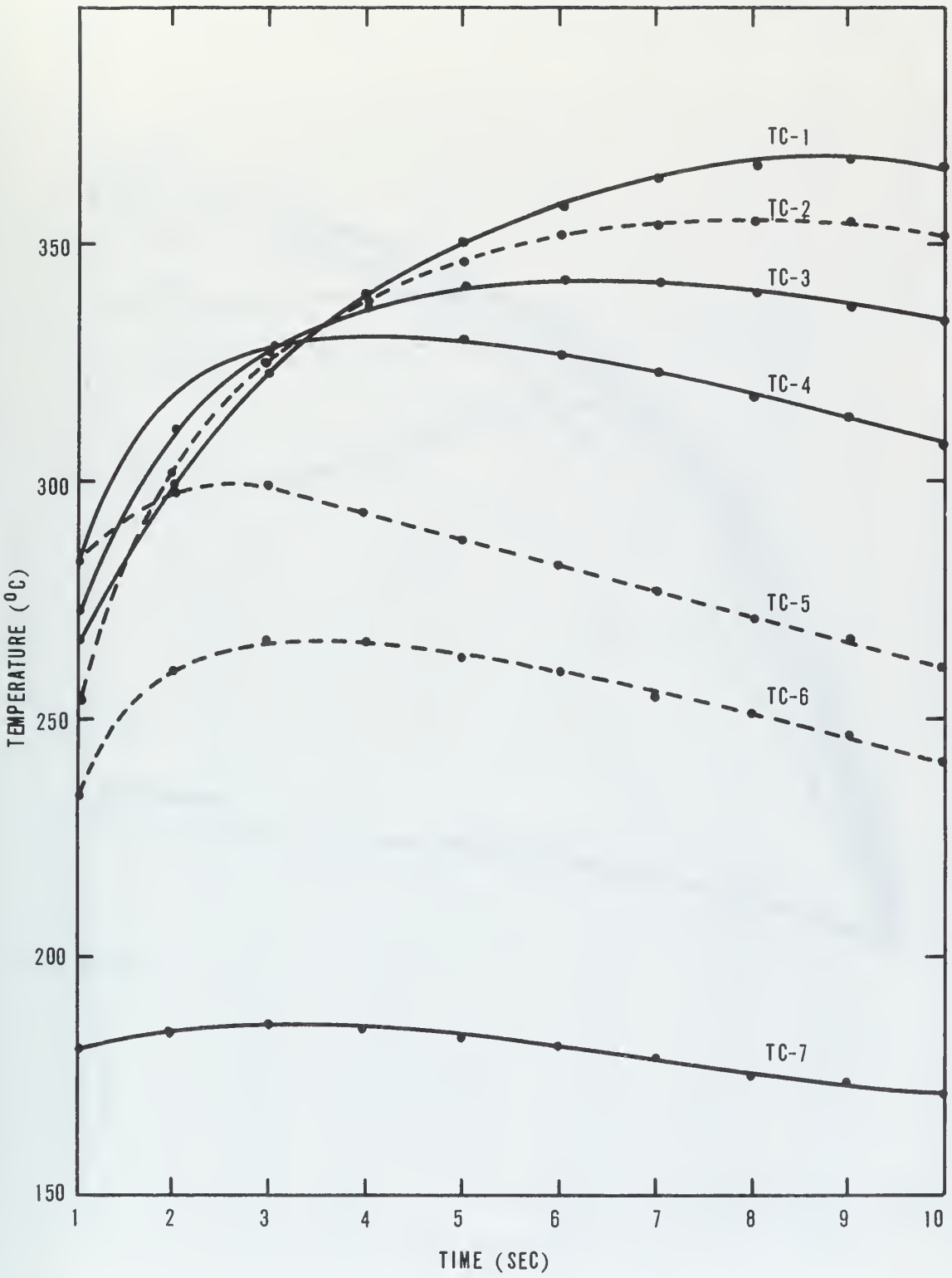

Fig. 3 Variations of Fuel Temperature at given Locations with Time

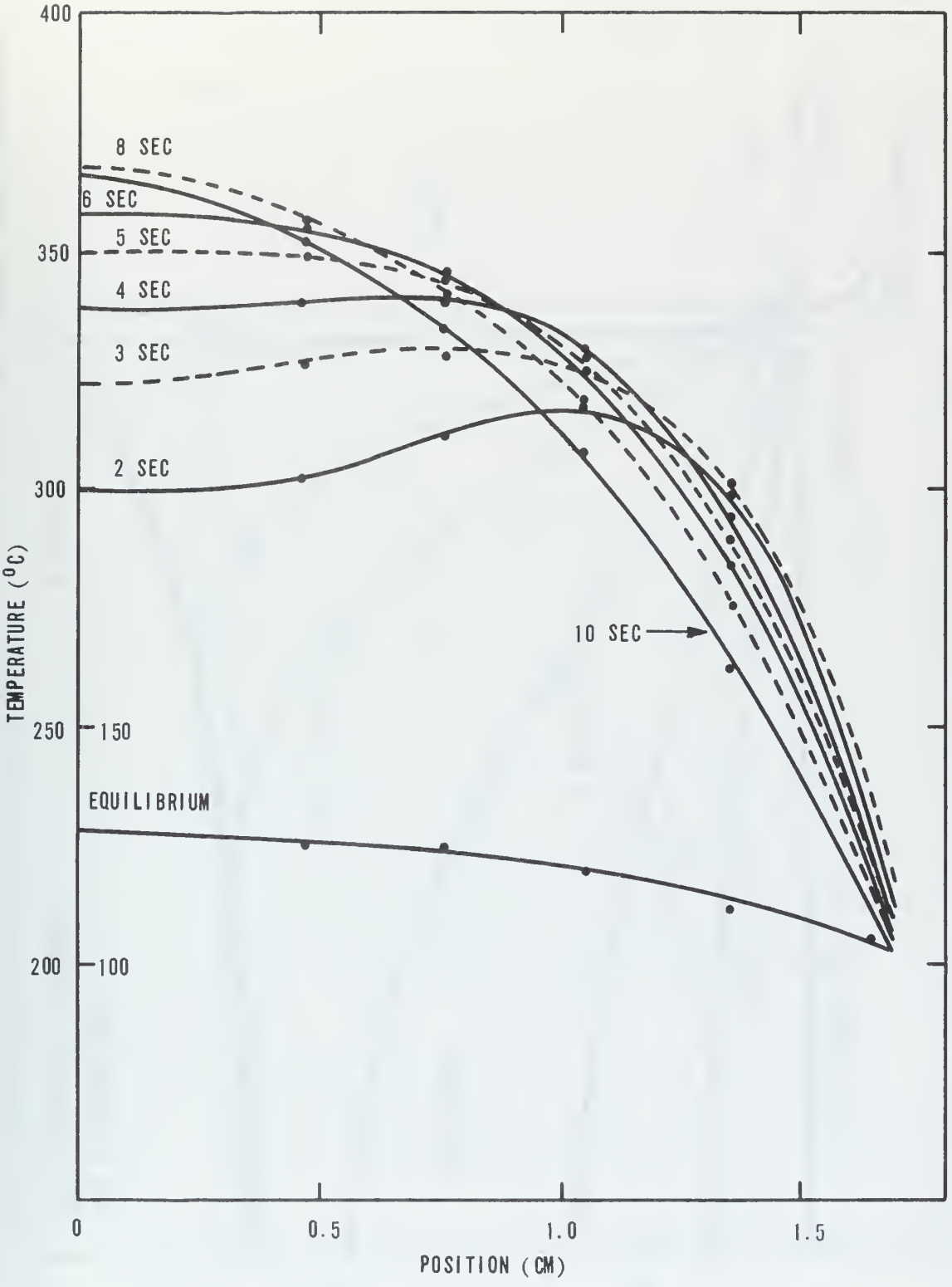

Fig. 4 Spatial Variation of Fuel Temperature at Different Times

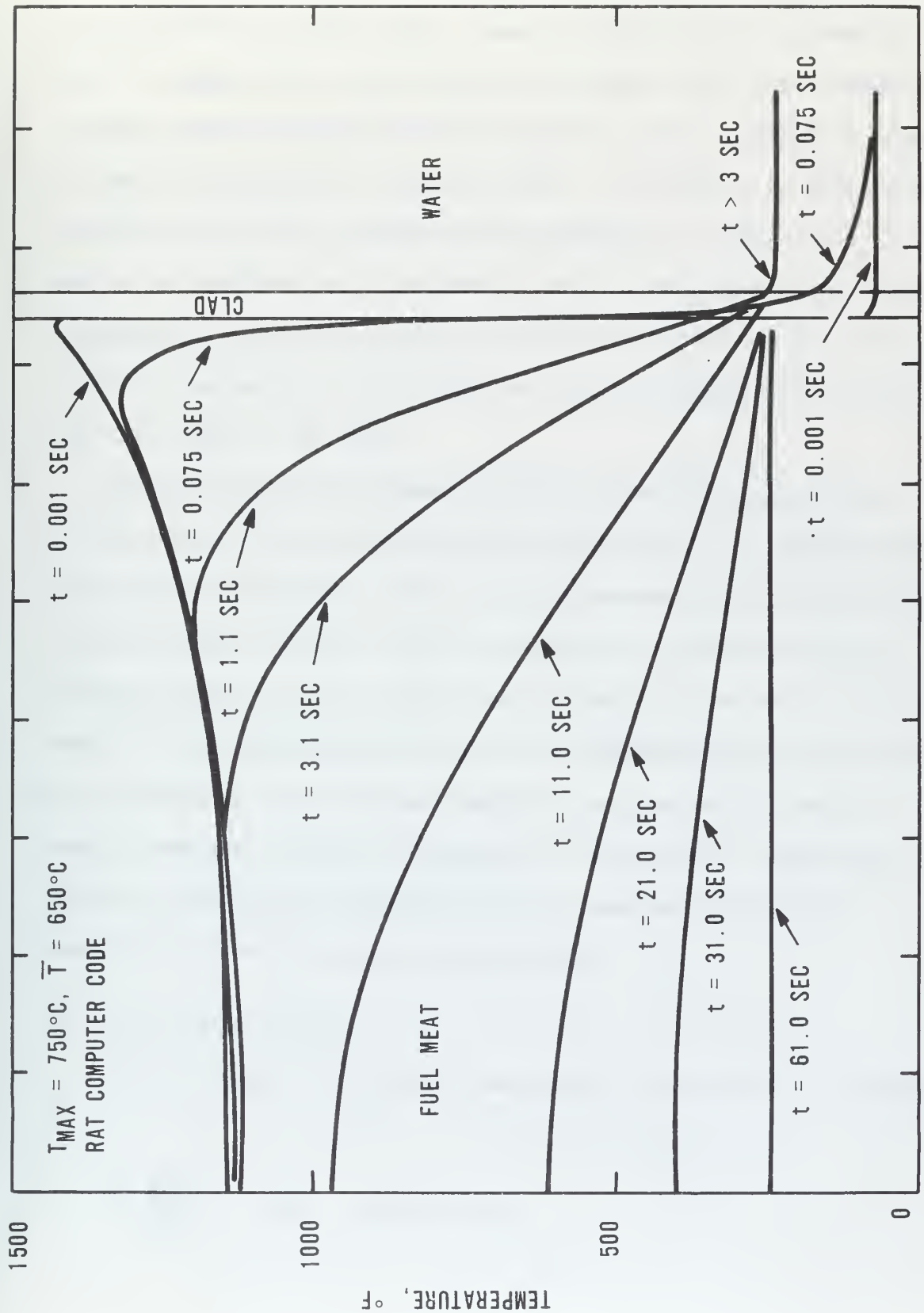

Fig. 5 Temperature Distribution in Fuel Meat following Pulse

A third difficulty in developing a multi-region heat transfer model was identified by Hornyik (16). Using the earlier fuel temperature results of Wyman (38), he combined a multi-region heat transfer model with a point reactor kinetics model in a digital computer simulation program to study the dynamics of a TRIGA reactor. In addition, he made extensive coolant flow rate and temperature measurements in a TRIGA core. By comparison of simulated and experimental results, he proposed that a radial component of coolant flow occurs in a TRIGA type core. This flow is a result of the pressure differences set up by the higher flow velocities near the center of the core.

The multi-region heat transfer model is complicated by several factors which are not easily described analytically. In contrast, the point reactor temperature model is easily represented by a simple heat transfer equation and all reactor parameters are treated as gross averaged quantities. The temperature indicated by the point reactor temperature model does not correspond to a physically measurable quantity. Instead, it is an average temperature related to the reactor kinetic behavior through the temperature coefficient of reactivity. Similarly, the average temperature can be measured indirectly by measuring the reactivity (see Section VII).

D. Heat Balance Equation

The heat balance in the point temperature reactor model is described by

$$C\frac{dT}{dt} = P - h(T - T_o)$$

where

C = heat capacity of the reactor (watt-sec/°C)

T = reactor temperature (°C)

T_0 = coolant temperature (°C)

P = reactor power (watts)

h = heat transfer coefficient (watt/°C)

Then

$$\frac{dT}{dt} = \eta P - \gamma (T - T_0)$$

where

η = reciprocal heat capacity (°C/watt-sec)

γ = heat transfer coefficient (ratio of h to C) (sec^{-1}).

The coolant temperature T_0 is assumed constant at 25°C. This is the initial value of T, and at high power levels, T_0 can be neglected relative to T.

The heat transfer coefficient γ is temperature dependent and can be expressed as

$$\gamma(T) = \frac{h(T)}{C(T)} = \frac{h_s(T) A_s(T)}{c_p(T)\, \delta(T)\, V(T)}$$

where

$h_s(T)$ = fuel element surface coefficient of heat transfer (watt-sec/sec-°C-cm^2)

$A_s(T)$ = total fuel element surface area (cm^2)

$c_p(T)$ = fuel element specific heat (watt-sec/gram-°C)

$\delta(T)$ = fuel density (gram/cm^3)

$V(T)$ = total fuel element volume (cm^3)

The quantities $h_s(T)$, $c_\rho(T)$ and $\delta(T)$ are characteristic of the fuel element materials and are, therefore, independent of the actual number of fuel elements. The quantities $A_s(T)$ and $V(T)$ could be expressed as

$$A_s(T) = a_s(T)\, N$$

and

$$V(T) = v(T)\, N$$

where

$a_s(T)$ = individual fuel element surface area

$v(T)$ = individual fuel element volume

N = total number of fuel elements

Then

$$\gamma(T) = \frac{h_s(T)\, a_s(T)\, N}{c_\rho(T)\, \delta(T)\, v(T)\, N} = \frac{h_s(T)\, a_s(T)}{c_\rho(T)\, \delta(T)\, v(T)}$$

and, thus, $\gamma(T)$ is independent of the number of fuel elements in a particular fuel loading.

The heat capacity per fuel element for stainless steel clad TRIGA fuel elements has been reported by West (36) as

$$\frac{C}{N} = 857 + 1.60(T - 25°C) \text{ (watt-sec/°C-element)}$$

E. Transient Heat Transfer Modes

The heat transfer from a pool type pulse reactor can be described by several heat transfer modes which occur successively following a pulse transient. Rivard (27) has studied these modes and a summary of his results follows.

Consider an extremely simplified thermal-hydraulic model in which the reactor fuel element is idealized as a vertical fuel cylinder concentric within its stainless steel cladding. Between the fuel and cladding is a thin, air-filled annulus. Each fuel element is assumed to be surrounded by a uniform channel of quiescent, subcooled water at the time of a reactor pulse. Following a pulse, a large temperature gradient is thus impressed from the fuel surface, across the thin gas gap, to the cladding. Large radial heat transfer rates result, by combined radiation and conduction, with the magnitude of the total heat flux depending primarily upon the instantaneous value of the thermal conductance of the gap.

If the gap conductance is initially large, subcooled nucleate boiling or subcooled film boiling will ensue after enough heat has been transferred to raise the cladding above the coolant saturation temperature. This might be called a flash boiling mode.

If the gap conductance is smaller, less severe convection or boiling heat transfer will follow the pulse, and the heated water surrounding the fuel elements will gradually accelerate due to its buoyancy. If the instantaneous acceleration is assumed to be proportional to the water temperature rise, an assumed constant heat flux into the initial core water will produce acceleration linearly increasing with time. Double

integration of this acceleration yields an upward displacement of this core water proportional to the cube of time. The early displacement is therefore small relative to the displacement at later times; thus the water initially in the reactor core is heated considerably before being totally displaced by cooler water from below. This decrease in the subcooling of the initial core water can result in a local, severe, delayed boiling transient due to a reduced threshold for film boiling. This is an inertia-dominated heat transfer mode.

Finally, at later times after a pulse (several seconds), a quasi-steady condition is achieved in which the fluid acceleration is negligible because the heat-flux-induced buoyant forces through the channel just balance the frictional losses.

The Rivard study contains sufficient material to develop a simple model describing the heat transfer from a pool type pulse reactor. This model could be combined with the point temperature model to more accurately describe the thermal-hydrodynamics of reactors of this type. In this report, the reactor heat transfer characteristics are lumped into the emperical heat transfer coefficient determined by Kurstedt (20) (see Section III-A). This is a possible area for future study.

III. REACTOR PARAMETERS

A. Heat Transfer Coefficient

There are two temperature dependent effects that determine the heat transfer from the reactor fuel to the water coolant. First, the heat transfer coefficient has been determined to be a linear function of temperature over a wide temperature range. Second, thermal expansion of the fuel changes the size of the gap between the fuel and the cladding.

Kurstedt (20) determined the temperature dependence of the heat transfer coefficient by an emperical fit to experimental data. He determined the reactor average temperature by measuring the temperature dependent reactivity by the technique discussed in Section VII. He found the form of the temperature dependence of the heat transfer coefficient γ to be

$$\gamma(T) = \gamma_1 + \gamma_2 T$$

where γ_1 and γ_2 are constants and T is the reactor average temperature.

The reactor used by Kurstedt was the University of Illinois TRIGA Mark II reactor. The major differences between the TRIGA Mark II reactor and the PSTR TRIGA Mark III are that the Mark II fuel elements are clad in aluminum rather than stainless steel, and the core is radially and vertically graphite reflected, while the PSTR is reflected radially with water. Thus, the values of the heat transfer coefficient constants γ_1 and γ_2 might be expected to be different in the Mark II and Mark III reactors, although the form of the temperature dependence is

assumed to be the same. In this study, the TRIGA Mark II values were used:

$$\gamma(T) = \gamma_1 + \gamma_2 T$$

where

γ_1 = 0.02 sec^{-1}

γ_2 = 0.000033 $sec^{-1} \cdot C^{-1}$

T = average reactor temperature (°C)

The fuel element expansion in the PSTR fuel elements is examined theoretically by Hoover (15). The thermal expansion effects in TRIGA type fuel elements was examined experimentally by Coffer (8). During pulsing operation the fuel reaches high temperatures, and the resulting expansion causes the fuel cladding to be expanded beyond its elastic limit. Upon cooling the fuel then contracts more than the cladding producing a gap between the fuel and the cladding. This gap reduces heat transfer and causes higher steady state fuel temperatures. The size of the gap is determined entirely by the largest reactivity pulse and, hence, the highest temperature in the fuel element life history.

Coffer (8) determined the effect of gap size on the steady state operating temperature of the fuel elements. Virgin fuel elements were first operated over the steady state power range in a TRIGA reactor, and the measured fuel temperatures were recorded. The reactor was then pulsed with successively larger reactivities causing correspondingly higher peak fuel temperatures. After each pulse the steady state operating temperatures were again measured. The results of the study are shown in Figure 6.

The gap between the fuel and cladding is largest at low temperatures

and decreases as the fuel expands at higher temperatures. At some temperature, the gap width is reduced to zero. At that temperature, the heat transfer from the fuel to the cladding, and then to the water, would be expected to increase significantly. The magnitude of this effect was examined by setting the heat transfer coefficient equal to zero (simulating zero heat transfer) until the temperature reached some value during the temperature rise following a pulse. The result was a difference in peak power of 0.1 percent and in the corresponding temperature of 0.4 percent (see Section III-C).

B. Ratio of Reactor Power to Thermal Neutron Density

The ratio P/N is used as a conversion factor to determine reactor power from neutron density in the point reactor model. Two independent calculations of this ratio yield the same value within a factor of two.

From Lamarsh (21), the power generated in the reactor is

$$P = \varepsilon \int_V \int_0^\infty \Sigma_f(E)\, \varphi(\underline{r}, E)\, dE\, dV$$

where

ε = recoverable energy per fission (200 mev)

Since

$$\Sigma_f(E) = N\sigma_f(E) = \frac{m_f N_0}{A V}\sigma_f(E)$$

where

m_f = fissionable fuel mass (3.5 kg)

N_0 = Avagadro's number

A = fissionable fuel mass number (235)

V = fuel volume

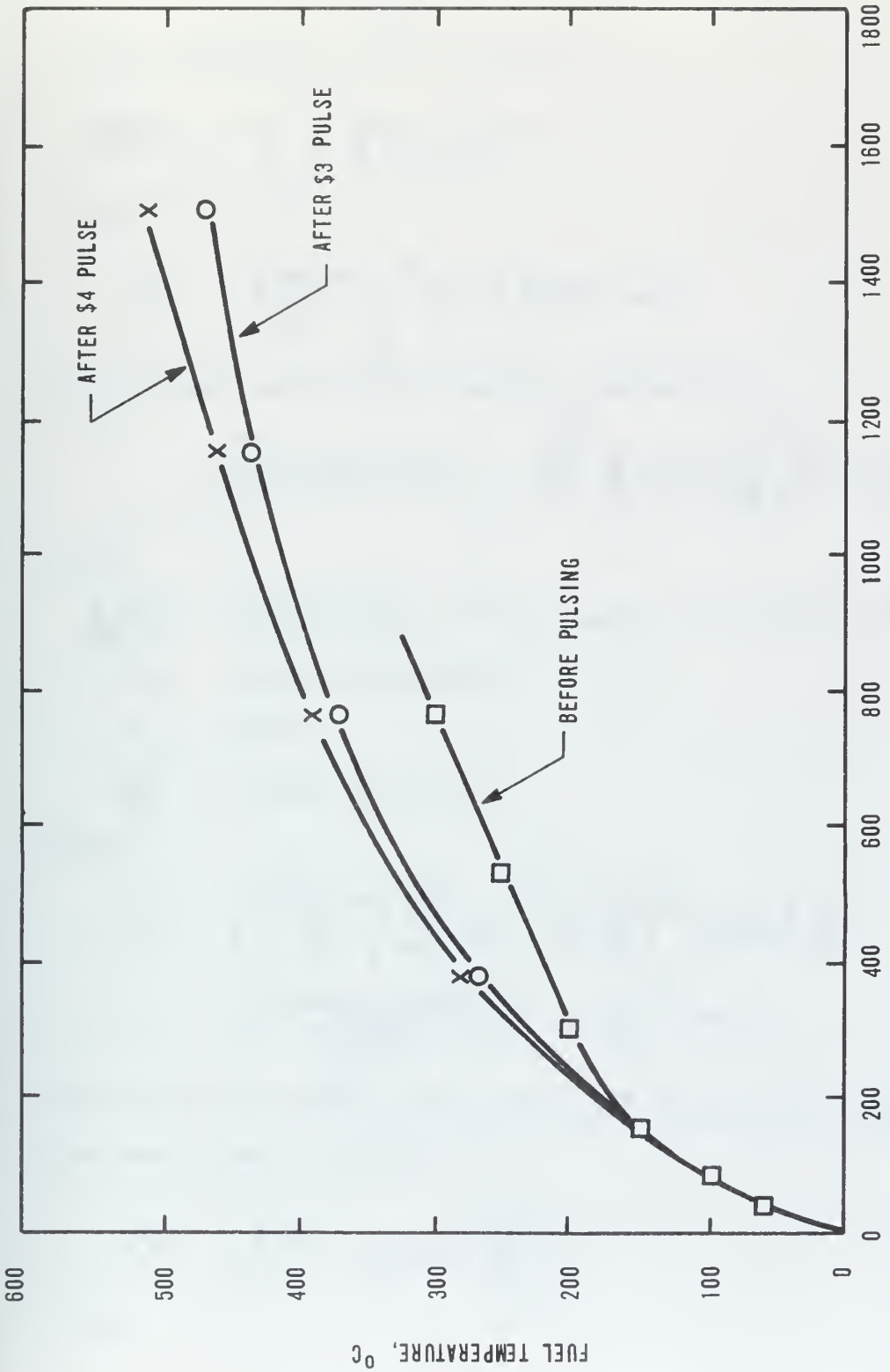

Fig. 6 Fuel Temperature Required for a Given Steady-State Reactor Power as a Function of Transient Size Performed on Fuel

then

$$P = \frac{\varepsilon m_f N_0}{AV} \int_V \int_0^\infty \sigma_f(E)\, \varphi(\underline{r}, E)\, dE\, dV$$

If the average flux $\bar{\varphi}$ is defined as

$$\bar{\varphi}(E) = \frac{1}{V} \int_V \varphi(\underline{r}, E)\, dV$$

then

$$P = \frac{\varepsilon m_f N_0}{A} \int_0^\infty \sigma_f(E)\, \bar{\varphi}(E)\, dE$$

In a thermal reactor with Maxwellian thermal flux,

$$\int_0^\infty \sigma_f(E)\, \bar{\varphi}(E)\, dE = \frac{\sqrt{\pi}}{2}\, g_f(T_n) \left(\frac{T_0}{T_n}\right)^{\frac{1}{2}} \sigma_f(E_0)\, \bar{\varphi}_T$$

where

$g_f(T_n)$ = non-1/v factor for fissionable fuel isotope (≈ 0.95)

T_n = neutron temperature

T_0 = 293°K

$\bar{\varphi}_T$ = average thermal flux

Then

$$P = \left(\frac{\varepsilon m_f N_0}{A}\right) \frac{\sqrt{\pi}}{2}\, g_f(T_n) \left(\frac{T_0}{T_n}\right)^{\frac{1}{2}} \sigma_f(E_0)\, \bar{\varphi}_T$$

$$= \frac{7.19\, m_f\, g_f(T_n)\, \bar{\varphi}_T}{\sqrt{T_n}} \times 10^{-13}\ m\omega$$

In a Maxwellian thermal neutron flux φ at a neutron temperature T_n, the most probable energy is

$$E_T = k T_n = \frac{1}{2} m v_T^2$$

where

k = Boltzmann's constant

m = neutron mass

v_T = neutron velocity

Then

$$\frac{v_T}{2200\ m/sec} = \sqrt{\frac{T_n}{T_o}}$$

where

T_o = 293°K,

and the neutron velocity in thermal equilibrium at T_o is 2200 m/sec.

Then

$$v_T = 2.2 \times 10^5 \left(\frac{T_n}{T_o}\right)^{\frac{1}{2}}$$

$$= \frac{2.2 \times 10^5}{\sqrt{293}} \sqrt{T_n}$$

$$= 1.285 \times 10^4 \sqrt{T_n}\ cm/sec$$

Then the ratio of reactor power to thermal neutron density is

$$\frac{P}{n} = \frac{P}{n v_T} v_T = \frac{P}{\bar{\varphi}_T} v_T$$

or

$$\frac{P}{n} = \left(\frac{7.19\ m_f\ g_f(T_n)}{\sqrt{T_n}} \times 10^{-13}\right) v_T$$

$$= \frac{7.19(3.5)(0.95) \times 10^{-13}}{\sqrt{T_n}} \times 1.285 \times 10^4 \sqrt{T_n}$$

$$= 3.07 \times 10^{-8}\ \frac{mw}{n/cm^3}$$

$$= 3.07 \times 10^{-2}\ \frac{watts}{n/cm^3}$$

An independent calculation of the ratio P/n uses experimentally measured thermal fluxes. Since the reactor power is determined by the fission rate, the ratio of power to neutron flux is a constant, neglecting the dependence on neutron temperature. However, the neutron flux across the reactor is a function of position, with a maximum at the core center and a minimum at the edges. For a fixed power level P, the corresponding neutron flux φ in the constant ratio P/φ would be somewhere between the minimum and the maximum. That value of the flux is also the average flux $\bar{\varphi}$ defined as

$$\bar{\varphi} = \frac{1}{V} \int_V \varphi(\underline{r})\, dV$$

The neutron flux in a vertical cylindrical reactor is described approximately by the expression (21)

$$\varphi(r,z) = A' \cos\left(\frac{\pi z}{H}\right) J_0\left(\frac{2.405\, r}{R}\right)$$

where R and H are the extrapolated radius and height of the reactor core and r and z are measured from the core center.

Across the horizontal plane passing through the core center,

$$\varphi(r) = A\, J_0\left(\frac{2.405\, r}{R}\right)$$

and the average value of the neutron flux is

$$\bar{\varphi} = \frac{1}{R} \int_0^R \varphi(r)\, dr$$

$$= \frac{1}{R} \int_0^R A\, J_0\left(\frac{2.405 r}{R}\right) dr$$

Define $\rho = 2.405\, r/R$. Then

$$d\rho = \frac{2.405\, dr}{R}$$

or

$$dr = \frac{R\, d\rho}{2.405}$$

and

$$\bar{\varphi} = \frac{A}{R}\int_0^{2.405} J_0(\rho)\left(\frac{R}{2.405}\right) d\rho$$

$$= \frac{A}{2.405}\int_0^{2.405} J_0(\rho)\, d\rho$$

From reference (1),

$$\int_0^{2.405} J_0(\rho)\, d\rho = 1.47$$

Then

$$\bar{\varphi} = \frac{A}{2.405}(1.47) = 0.612\, A = 0.612\, \varphi_{max}$$

where φ_{max} is the maximum value of the neutron flux and is attained at the core center.

Substituting in the expression for $\varphi(r)$,

$$\varphi(r) = 0.612\, A = A\, J_0(\rho)$$

or

$$J_0(\rho) = 0.612$$

Again from reference (1),

$$\rho = 1.31 = 2.405\, r/R$$

and

$$r = 1.31R/2.405 = 0.545\, R$$

Thus, at the point $r = 0.545R$, the neutron flux equals the average flux $\bar{\varphi}$. Fortuitously, the thermal neutron flux at exactly this position is given by reference (12) for a standard TRIGA Mark III reactor at a reactor power of 1 megawatt and at 2000 megawatts. These values are listed in Table 3. Reference (12) also gives the peak neutron flux at the same power levels and these flux levels are the same as those for the PSTR given by Geisler, et. al. (11). These peak neutron flux values are also listed in Table 3.

Table 3

Thermal Neutron Flux (12)

	1 mw steady state	2000 mw pulse
Average thermal flux	2.14×10^{13} nv	4.28×10^{16} nv
Peak thermal flux	3.26×10^{13} nv	6.52×10^{16} nv

Using the average thermal flux $\bar{\varphi}_T$ given in Table 3, the ratio $\frac{P}{\bar{\varphi}_T}$ is

$$\frac{P}{\bar{\varphi}_T} = \frac{10^6}{2.14 \times 10^{13}} = \frac{2000 \times 10^6}{4.28 \times 10^{16}} = 4.67 \times 10^{-8} \ \frac{\text{watts}}{\text{nv}}$$

Recall that

$$v_T = 1.285 \times 10^4 \sqrt{T_n}$$

Then the ratio of power to neutron density is

$$\frac{p}{n} = \frac{p}{n v_T} v_T = \frac{p}{\phi_T} v_T$$
$$= 4.67 \times 10^{-8} \times 1.285 \times 10^4 \sqrt{T_n}$$
$$= 6.00 \times 10^{-4} \sqrt{T_n} \ \frac{watts}{n/cm^3} (^\circ C)^{\frac{1}{2}}$$

For a neutron temperature T_n of 100°C = 393°K

$$\frac{p}{n} = 6.00 \times 10^{-4} \sqrt{393}$$
$$= 1.19 \times 10^{-2} \text{watts}/(\text{n}/\text{cm}^3)$$

For a neutron temperature T_n of 400°C = 693°K

$$\frac{p}{n} = 6.00 \times 10^{-4} \sqrt{693}$$
$$= 1.58 \times 10^{-2} \text{watts}/(\text{n}/\text{cm}^3)$$

These values for the ratio $\frac{p}{n}$ differ from the previous value

$$\frac{p}{n} = 3.07 \times 10^{-2} \text{watts}/(\text{n}/\text{cm}^3)$$

by roughly a factor of two. The flux data from reference (12) are considered less reliable than the relationship between power and flux derived from Lamarsh (21). Therefore, the value used for the ratio $\frac{p}{n}$ is 3.07×10^{-2}.

C. Prompt Negative Temperature Coefficient

The prompt negative temperature coefficient of reactivity is defined by the logarithmic temperature derivative of the reproduction factor k, where, following the notation of Hoover (15),

$$k = \eta f p \epsilon P_{nl}$$

and

k = neutron reproduction factor

η = fission neutrons emitted per thermal neutron absorbed in fuel

f = thermal utilization, thermal neutrons absorbed in fuel per thermal neutron absorbed

p = resonance escape probability, thermal neutrons absorbed per neutron absorbed

ϵ = fast fission factor, fission neutrons per neutron from thermal fission

P_{nl} = non-leakage probability

Then the prompt negative temperature coefficient α is expressed as

$$\alpha = \frac{1}{k}\frac{dk}{dT} = \frac{1}{\eta f}\frac{d(\eta f)}{dT} + \frac{1}{p\epsilon}\frac{d(p\epsilon)}{dT} + \frac{1}{P_{nl}}\frac{d(P_{nl})}{dT}$$

and there are three terms contributing to α ,

$\frac{1}{\eta f}\frac{d(\eta f)}{dT}$ = thermal cell coefficient

$\frac{1}{p\epsilon}\frac{d(p\epsilon)}{dT}$ = fast cell coefficient

$\frac{1}{P_{nl}}\frac{d(P_{nl})}{dT}$ = leakage coefficient

West (36) has computed the total TRIGA prompt negative temperature coefficient as a function of temperature. The results of that study are shown in Figure 7. In addition, West (36) formulated the theory of the TRIGA prompt negative temperature coefficient. More recently Hoover (15) has studied the thermal cell coefficient for the PSTR in more detail. His values for the PSTR thermal cell coefficient (TCC) as a function of

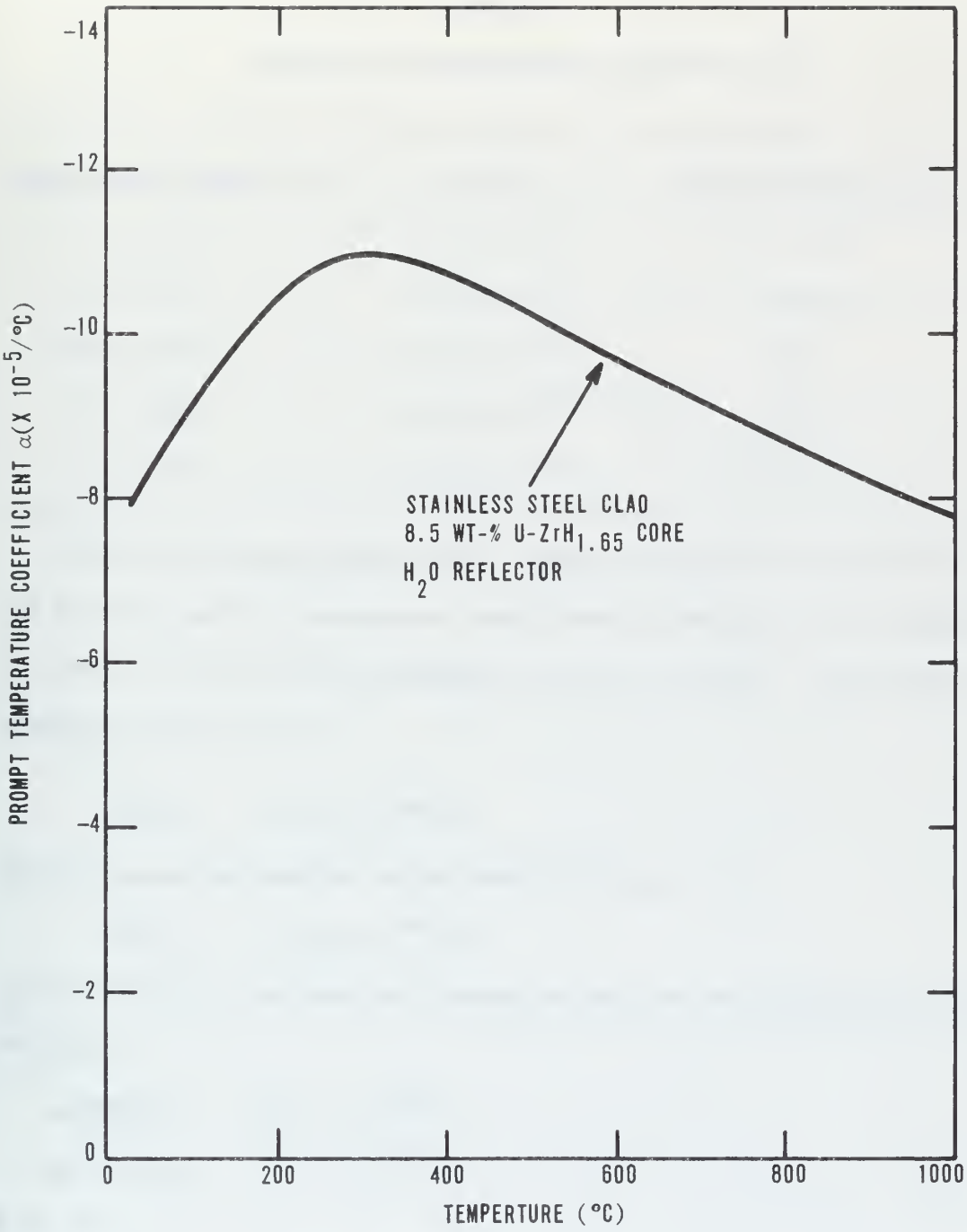

Fig. 7 Prompt Negative Temperature Coefficient Versus Average Fuel Temperature (West-36)

temperature are given in Table 4.

Table 4

Thermal Cell Coefficient (Hoover - 15)

Temperature Range (°C)	(TCC)	Median Temperature
23 - 127	-7.6×10^{-5}	75
127 - 227	-8.6×10^{-5}	177
227 - 327	-8.0×10^{-5}	277
327 - 427	-8.0×10^{-5}	377
23 - 427	-8.0×10^{-5}	--

Using the Fuchs-Nordheim model, Hoover obtained an average value for the prompt negative temperature coefficient by analysis of 34 pulses of the PSTR with reactivity insertions of \$2.00 to \$2.68. The average value obtained by Hoover was

$$\bar{\alpha}_H = -1.24 \times 10^{-4}/°C.$$

The corresponding average value obtained by West was

$$\bar{\alpha}_W = -1.01 \times 10^{-4}/°C.$$

Hoover and West also computed average values for the thermal cell coefficient;

$$\alpha(TCC)_H = -8.0 \times 10^{-5}/°C$$

$$\alpha(TCC)_W = -5.7 \times 10^{-5}/°C.$$

Note that

$$\bar{\alpha}_H - \bar{\alpha}_W = -0.23 \times 10^{-4}/°C$$

and that

$$\alpha(TCC)_H - \alpha(TCC)_W = -0.23 \times 10^{-4}/°C.$$

It is assumed that the deviation between the two prompt negative temperature coefficient values is due essentially to the deviation in the thermal coefficient values.

The Hoover data is not extensive enough to determine temperature dependence. Therefore, the data due to West was normalized with the data due to Hoover. According to both West and Hoover, the thermal cell coefficient contributes about 57 percent to the prompt negative temperature coefficient. Then, in general, the temperature dependence of the Hoover data will be approximated by

$$\alpha_H(T) = 0.43\,\alpha_\omega(T) + \left(\alpha(TCC)_H / \alpha(TCC)_\omega\right) 0.57\,\alpha_\omega(T)$$

$$= 0.43\,\alpha_\omega(T) + \left(-8.0/-5.7\right) 0.57\,\alpha_\omega(T)$$

$$= 0.43\,\alpha_\omega(T) + 0.8\,\alpha_\omega(T)$$

$$= 1.23\,\alpha_\omega(T)$$

Figure 8 gives $\alpha_H(T)$. Note also that, in particular,

$$\bar{\alpha}_H = 1.23\,\bar{\alpha}_\omega$$

$$= 1.23\,(-1.01 \times 10^{-4})$$

$$= -1.24 \times 10^{-4}$$

The "x" points in Figure 7 were obtained by normalizing the Hoover thermal cell coefficient data from Table 4 with the formula

$$\alpha_H^*(T) = 0.43\,\alpha_\omega(T) + \alpha(TCC)_H(T)$$

More recently than Hoover and West, Sears (29) determined an average prompt negative temperature coefficient using the Fuchs-Nordheim model for each of four reactivity insertions. These values are given in

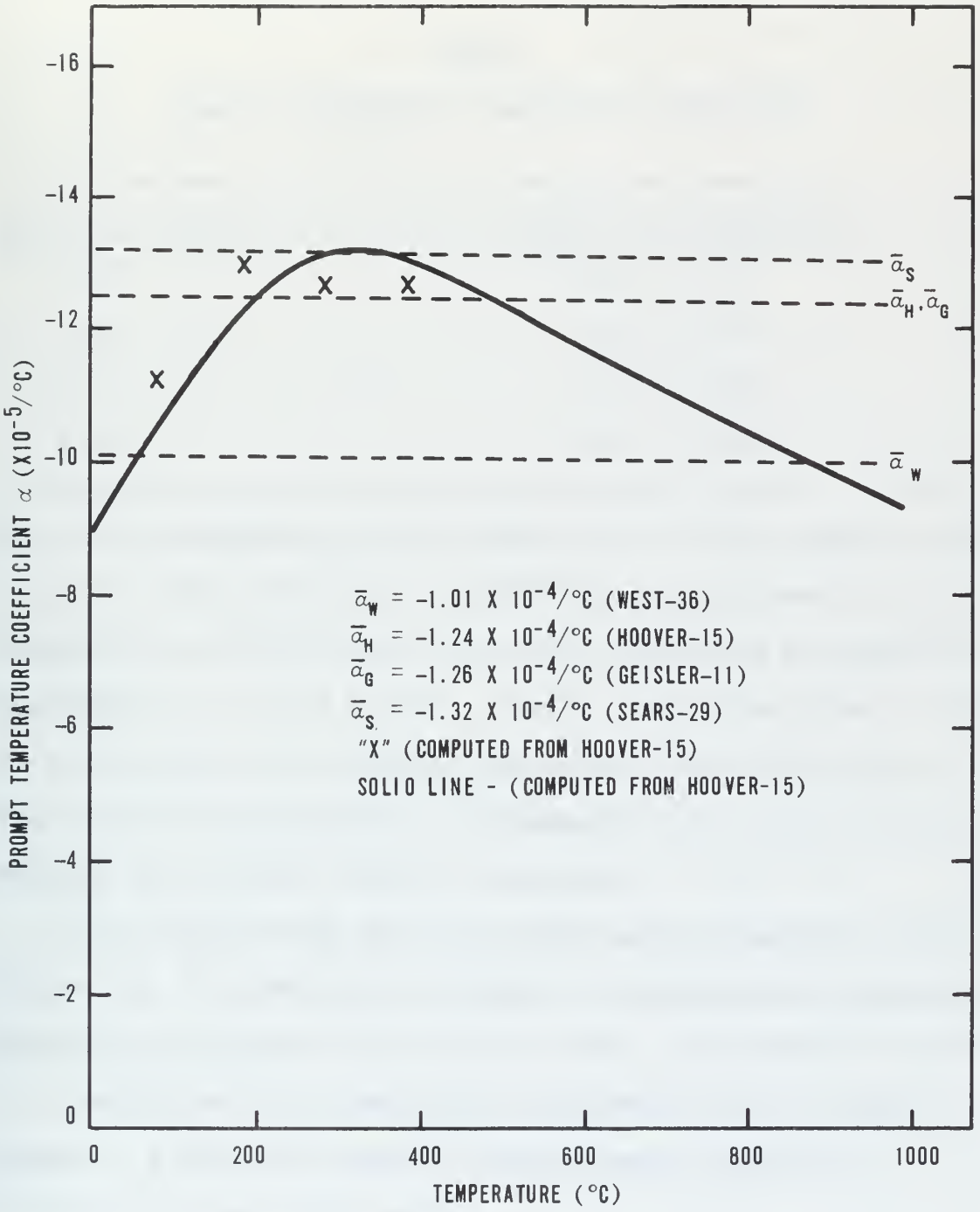

Fig. 8 Prompt Negative Temperature Coefficient Versus Temperature (Hoover-15)

Table 5. The value $\overline{\alpha}_s = 1.32 \times 10^{-4}/°C$ is shown in Figure 7 with the Hoover and West data.

Table 5

Computed Temperature Coefficients (Sears-29)

Reactivity inserted	Temperature coefficient
\$2.68	$1.31 \times 10^{-4}/°C$
\$2.41	$1.33 \times 10^{-4}/°C$
\$2.00	$1.33 \times 10^{-4}/°C$
\$1.71	$1.14 \times 10^{-4}/°C$

The value corresponding to \$1.71 differs from the other values by about 14 percent. Sears attributes this difference to heat transfer in the comparatively wide \$1.71 pulse, which would violate the Fuchs-Nordheim approximation of no heat transfer. Another contributing factor to this low value could be the temperature dependence of the coefficient α and the lower fuel temperatures corresponding to the lower peak power attained from a smaller reactivity insertion.

In the Fuchs-Nordheim model the prompt negative temperature coefficient α is assumed to be a constant. Although several important results can be obtained easily with this model, the assumption that α is a constant may not be justified in more general kinetic studies. Therefore, a temperature dependent prompt negative temperature coefficient is used in this study.

To determine the extent of the temperature dependence of the prompt negative temperature coefficient, it was necessary to compare simulated

responses with experimentally observed responses. Sears (29) reported peak powers resulting from four different reactivity insertions: \$1.71, \$2.00, \$2.41, and \$2.68. The simulated temperature dependence of α was varied until the simulated peak powers were approximately equal to the experimental peak powers.

The temperature at the peak of each power pulse varied with the size of the pulse.[1] First, a temperature dependence of α was determined for the temperature range up to the peak of the smallest pulse (\$1.71). Then a temperature dependence of α was determined for the range up to the next pulse (\$2.00), with the requirement that the temperature dependence be a smooth continuation of the previous result. Successively, a relatively smooth temperature dependence of α was determined over the range of all pulses. This temperature dependency of α is shown in Figure 9. The experimental and simulated peak powers are given in Table 11.

The effect of heat transfer during a small pulse as proposed by Sears would produce the same effect as the decrease in the size of α at low temperatures as shown in Figure 9. Heat transfer would result in lower temperatures with a resulting smaller negative reactivity due to temperature. A small α would similarly result in a smaller negative reactivity due to temperature.

To determine the extent of heat transfer during a pulse, simulated pulsed responses were compared for different heat transfer rates. For one set of pulsed responses the heat transfer coefficient described in

---

[1] The peak temperature always occurred much later than the peak of power pulse.

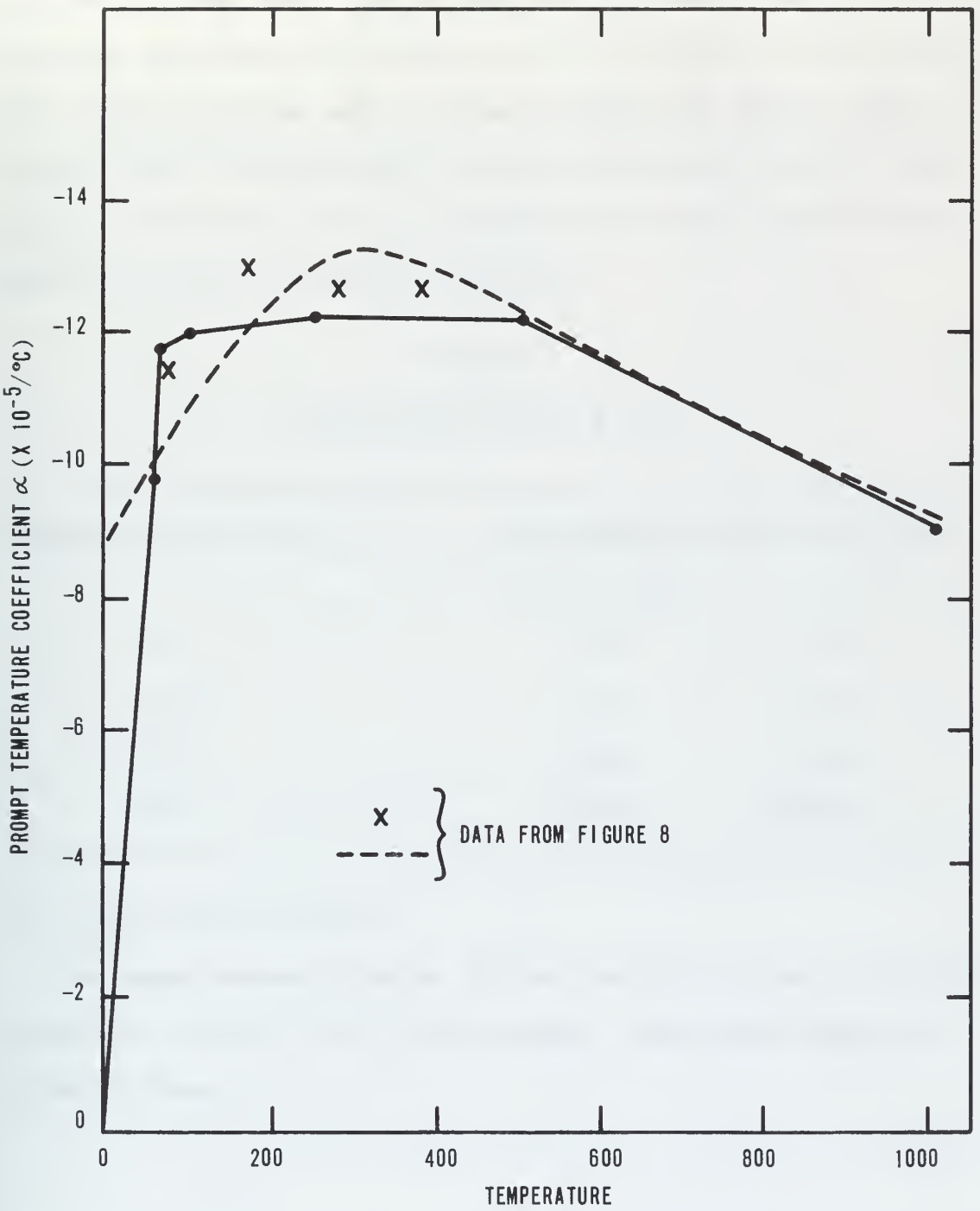

Fig. 9 Prompt Negative Temperature Coefficient (Simulated)

Section III-A was used. For the next set of pulsed responses the heat transfer coefficient was set equal to zero simulating no heat transfer. The results of the two sets of pulsed responses are given in Table 6. In every case, the difference in temperature rise was less than 0.25 degrees. Therefore, the heat transferred up to the time of the peak power in a pulse is considered negligible.

Table 6

Heat Transfer During a Pulse

Reactivity Inserted ($)	Temperature at Peak Power (°C)	
	$\gamma = \gamma_1 + \gamma_2 T$	$\gamma = 0$
1.71	61.52	61.27
2.00	69.29	69.29
2.41	93.36	93.36
2.68	108.68	108.68

D. Prompt Neutron Lifetime

The prompt neutron lifetime ℓ^* for the PSTR is given in different references as either 38 or 39 microseconds. Some quoted values are listed in Table 7.

Table 7

Prompt Neutron Lifetime

(microseconds)	Source	Reference
38	Hoover (15)	Geisler (11)
38	Pallette (24)	West (35)
38 ± 3	Pallette (24)	(Own result)
39	Sears (29)	West (36)

The value selected for this study is

$$\ell^* = 38.0 \text{ microseconds.}$$

In the simulation equations, the neutron generation time Λ is considered a constant rather than ℓ^*, as explained in Section II-B, where

$$\Lambda = \frac{\ell^*}{k}$$

The two parameters are strictly equal only in a critical reactor ($k = 1.0$). It is the critical value of ℓ^* which has been determined. Therefore, the neutron generation time is considered constant with the value

$$\Lambda = 38.0 \text{ microseconds.}$$

E. Delayed Neutron Data

The effective delayed neutron fraction for the PSTR is (11,35,36)

$$\bar{\beta} = 0.0070.$$

The remainder of the delayed neutron data is from Keepin (18) as summarized by Lamarsh (21) in Table 8.

Table 8

Delayed Neutron Data

Group	Half-life (sec)	Decay Constant (sec^{-1})	Yield Fraction
1	55.72	0.0124	0.000215
2	22.72	0.0305	0.001424
3	6.22	0.111	0.001274
4	2.30	0.301	0.002568
5	0.610	1.14	0.000748
6	0.230	3.01	0.000273
		Total	0.006502

IV. PULSED REACTIVITY SIMULATION

A. Transient Rod Reactivity Pulse

The reactivity insertion in a pulsing reactor is often described loosely as a step insertion. In reality, the reactivity in a TRIGA reactor is inserted over a period of several tenths of a second, while the resulting power pulse width is about 20 milliseconds.

Sears (29) made a high speed motion picture study at 1000 frames per second of the motion of the transient rod in the PSTR. He determined the full stroke withdrawal time to be about 210 milliseconds. This writer verified the withdrawal time to be about 213 milliseconds using secondary contacts on the pulsing switch at the reactor console and a contact microphone at the base mounting of the transient rod shock absorber cylinder.

The motion of the transient rod with respect to time as determined by Sears is shown in Table 9 and in Figure 10. An integrated rod worth curve for the transient rod is shown in Figure 11. The data in these two figures were combined to show the reactivity inserted by the transient rod as a function of time in Figure 12.

The elapsed time during the motion of the transient rod is short; about 210 milliseconds. However, this time is long when compared with the reactor power pulse width; about 20 milliseconds at half peak power, depending on the reactivity inserted. Thus, the time for the insertion is in fact 10 times longer than the effect it produces. For this reason, accurate simulation of the PSTR kinetic response requires that the reactivity insertion be simulated as shown in Figure 12 rather than as a step insertion.

Table 9

Transient Rod Reactivity (Sears-29)

Time (msec)	Rod Position (ins)	Rod Position (units)	Reactivity[1] ($)
0	0.00	42	0.00
40	0.00	42	0.00
45	0.05	55	0.00
50	0.05	55	0.00
55	0.11	49	0.00
60	0.23	56	0.01
65	0.40	66	0.02
70	0.56	75	0.03
75	0.86	93	0.04
80	1.08	106	0.05
85	1.42	127	0.08
90	1.77	148	0.11
95	2.28	178	0.16
100	2.63	199	0.20
105	3.01	221	0.26
110	3.54	253	0.35
115	4.09	286	0.46
120	4.60	316	0.58
125	5.28	357	0.75
130	5.74	384	0.86
135	6.37	421	1.04

[1] From Figure 12

Table 9 (Continued)

Transient Rod Reactivity (Sears-29)

Time (msec)	Rod Position (ins)	Rod Position (units)	Reactivity[1] ($)
140	6.93	456	1.20
145	7.68	500	1.41
150	8.42	544	1.60
155	9.20	590	1.81
160	9.90	632	1.97
165	10.64	677	2.12
170	11.48	726	2.25
175	12.33	777	2.36
180	13.30	835	2.46
185	14.16	886	2.52
190	14.97	934	2.57
195	15.75	982	2.62
200	16.00	996	2.64
205	16.30	1014	2.65
210	16.50	1026	2.66
215	16.70	1037	2.67
220	16.80	1044	2.68

[1] From Figure 12

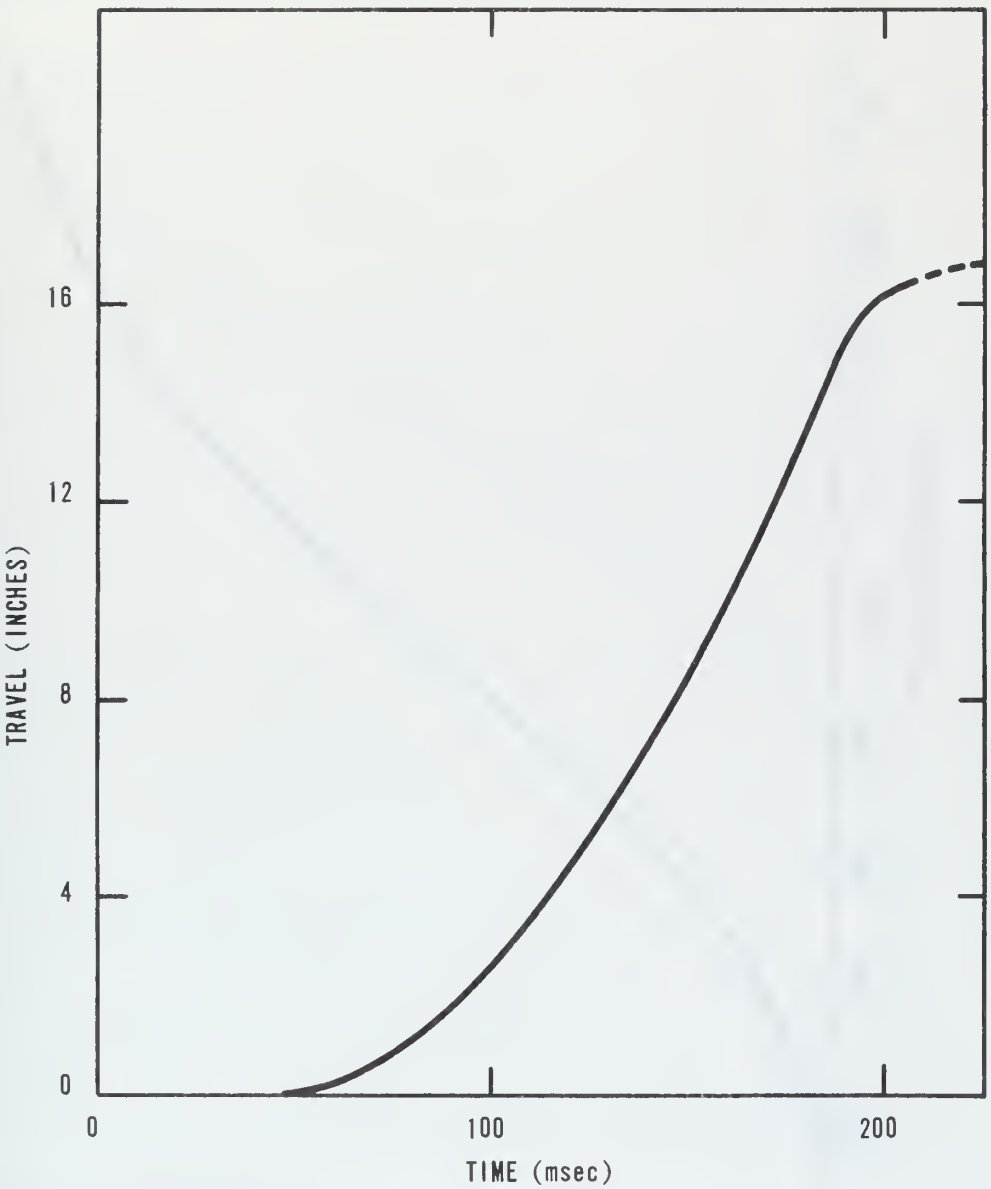

Fig. 10 Transient Rod Motion (Sears-29)

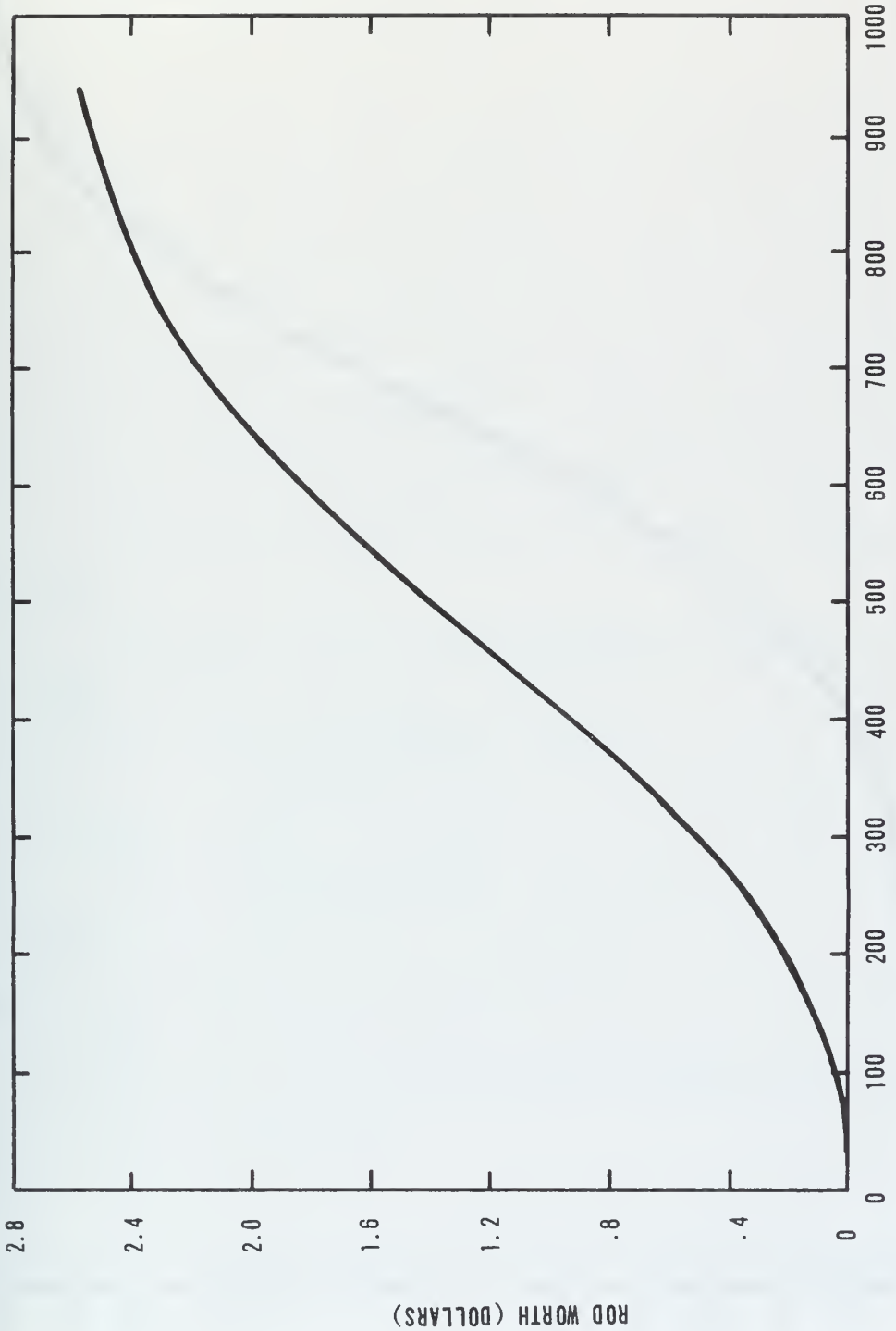

Fig. 11 Transient Rod Integrated Rod Worth Curve

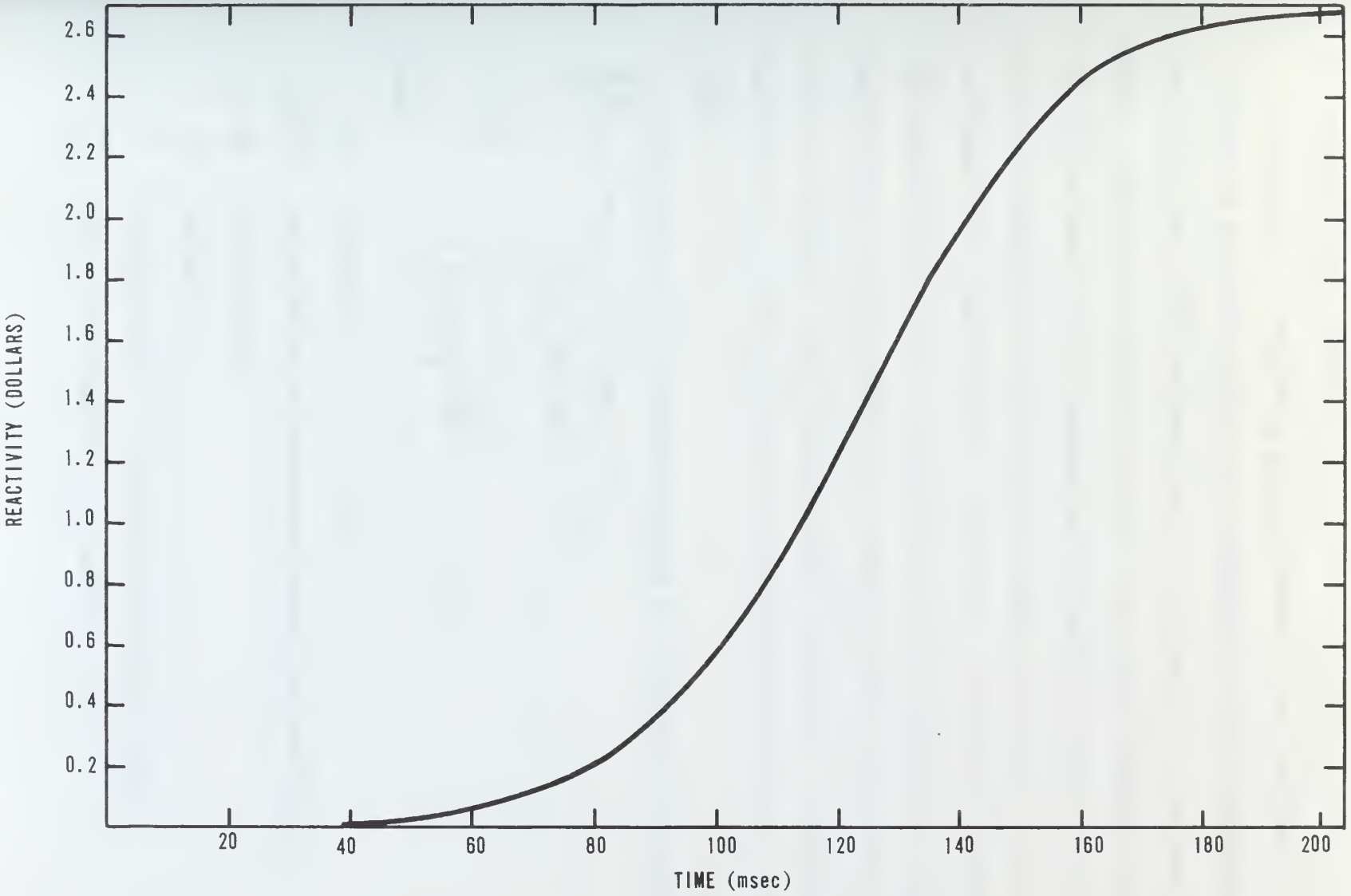

FIG. 12 Transient Rod Reactivity Insertion Rate

B. Reactivity Wheel Pulse

In Section I, a reactivity pulsing mechanism was described which consisted of absorber material on a rotating disk which intersected the reactor core. This mechanism will be referred to as the reactivity wheel. Depending on the pulse desired, the absorber material could be distributed over all but a segment of the reactivity wheel, or concentrated on a small area. To compute the shape of the pulse, the absorber is assumed to be concentrated on a small area on the rim of the wheel. The computation will use a one group first order perturbation theory approximation. By this method, the reactivity pulse from either absorber distribution can be analyzed, since the absorber perturbation in the one case is just the negative of the absorber perturbation in the other.

From Lamarsh (21), the reactivity due to an absorber perturbation $\delta\Sigma_a$ in the reactor volume V is

$$\rho = \frac{-\int_V \delta\Sigma_a \varphi^2 dV}{\nu \int_V \Sigma_f \varphi^2 dV}$$

where

ρ = reactivity

$\delta\Sigma_a$ = perturbation to macroscopic absorption coefficient

φ = neutron flux

V = volume

ν = average number of neutrons produced per fission

Σ_f = macroscopic fission cross section.

If the absorber perturbation occupies only a small area at the position $\underline{r}_o$,

$$\delta\Sigma_a \approx \Sigma_{a_p} V_p \, \delta(\underline{r} - r_o)$$

where

Σ_{a_p} = macroscopic absorption coefficient of absorber material

V_p = volume of absorber

$\delta(\underline{r})$ = Dirac delta function

Then

$$\rho = \frac{-\Sigma_{a_p} V_p \, \varphi^2(\underline{r}_o)}{\int_V \Sigma_f \, \varphi^2 \, dV} = K \varphi^2(\underline{r}_o)$$

where K is a constant.

In a cylindrical reactor, the flux is given approximately by

$$\varphi(r,z) = A \cos\left(\frac{\pi z}{H}\right) J_0\left(\frac{2.405 r}{R}\right)$$

where

A = constant dependent on power level

R = reactor extrapolated radius

H = reactor extrapolated height

and where the coordinates r and z are measured from the center of the core. For the PSTR,

$R \approx$ 8.5 inches

$H \approx$ 15 inches

In units of the reactor radius, which equals the wheel radius,

$R = 1.000$

$H = 1.765.$

The angular position θ of the reactivity wheel is measured in radians with the center of the core corresponding to $\theta = 0$. Then, for any θ, the position of a point on the wheel rim is described by the coordinates (r, z) where

$$r = 1 - \cos\theta$$

and

$$z = \sin\theta$$

Then, if the maximum reactivity of the reactivity wheel at the position $\theta = 0$ is $\hat{\rho}$, the reactivity for any position θ is

$$\rho = \hat{\rho}\,\varphi^2(r, z)$$

where the constant A is contained in $\hat{\rho}$.

For example, if $\theta = 1.0$, then

$$\rho = \hat{\rho}\left\{\cos\left[\frac{\pi \sin 1.0}{1.765}\right] J_0\left[\frac{2.405(1-\cos 1.0)}{1}\right]\right\}^2$$

$$= \hat{\rho}\left\{\cos(1.497)\, J_0(1.106)\right\}^2$$

$$= \hat{\rho}\left\{(0.71)(0.073)\right\}^2$$

$$= \hat{\rho}\,(0.052)^2$$

$$= 0.0027\,\hat{\rho}$$

By this method, the reactivity wheel pulse shape was computed. The computed pulse shape is shown in Figure 13. From that figure, the pulse width is 2.0 radians, or the ratio of the pulse width to the interval between pulse beginnings is $\frac{2}{2\pi} = \frac{1}{\pi}$.

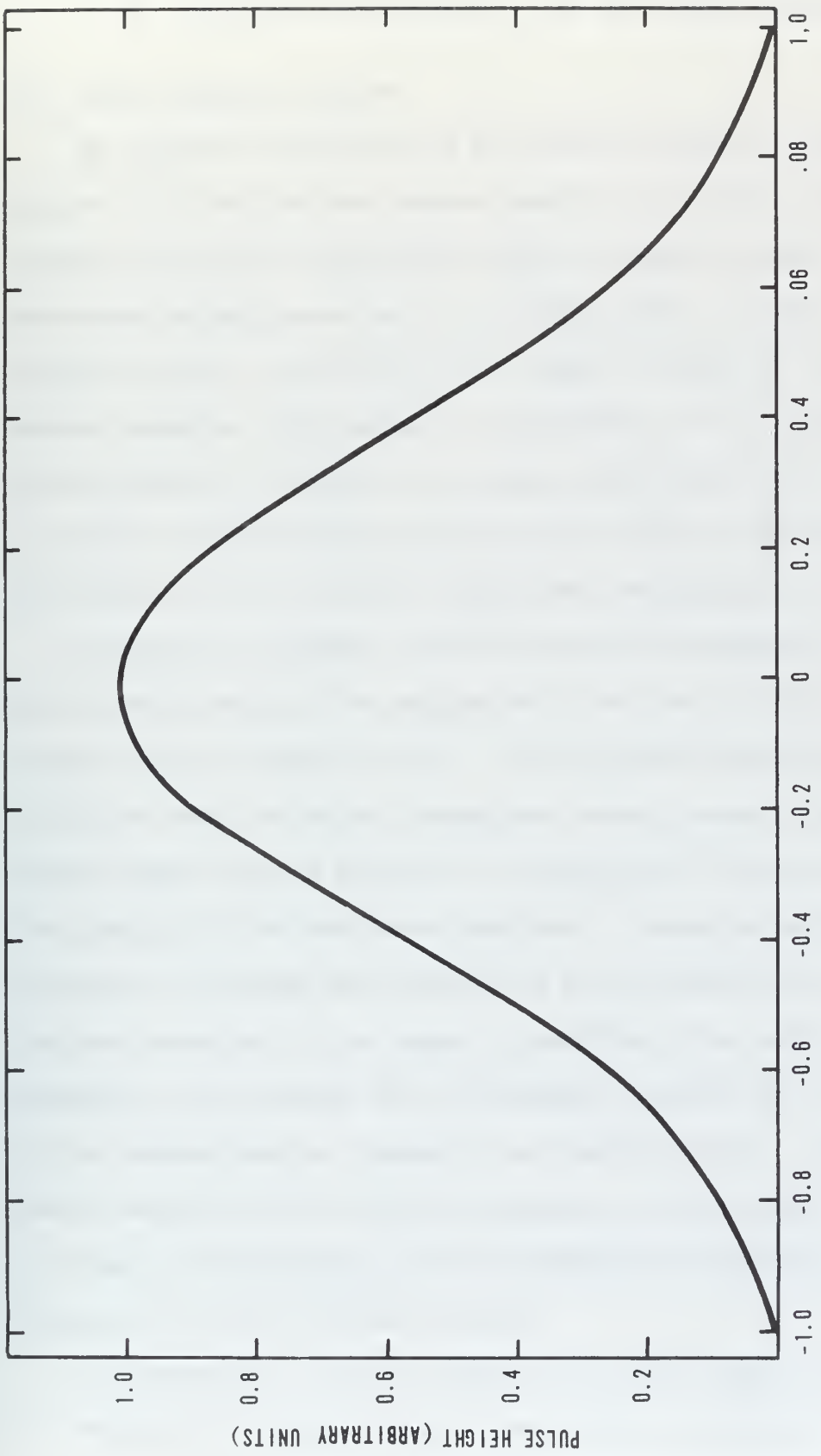

Fig. 13 Reactivity Wheel Pulse Height Versus Angular Position

V. TECHNIQUE FOR SOLUTION OF THE SIMULATION EQUATIONS

A. Analog Computer Solution

The simulation equations are the reactor kinetics equations (see Section II-B) and the heat balance equation (see Section II-D). Analog computer solutions to the reactor kinetics equations appear in several references in the literature (4,5,9,13,28,31,34). A typical analog computer program from Weaver (34) is shown in Figure 14. The heat balance equation could easily be incorporated in this program by adding a potentiometer to compute reactor power from neutron density and adding a feedback loop with function generators to compute temperature dependent reactivity, heat transfer coefficient, and reciprocal heat capacity.

Although the simulation equations could be programmed for solution on an analog computer, the program would have only limited use in the study of pulsed reactor kinetics. During pulsing operations, the variables and their derivatives change over several orders of magnitude. These changes would be difficult or impossible to account for by magnitude scaling of the simulation equations. A solution to this problem was developed to simulate the startup of a nuclear reactor from source levels to power operation. By a change of variables in the reactor kinetics equations, the equations can be programmed to solve for the logarithm of the neutron density, instead of the neutron density. Examples of analog computer reactor startup simulation programs appear in several references (4,10,28,31,34). These programs could also be modified to incorporate the heat balance equation.

The logarithmic solution programs for analog computers can simulate the response of a reactor over several orders of magnitude. However,

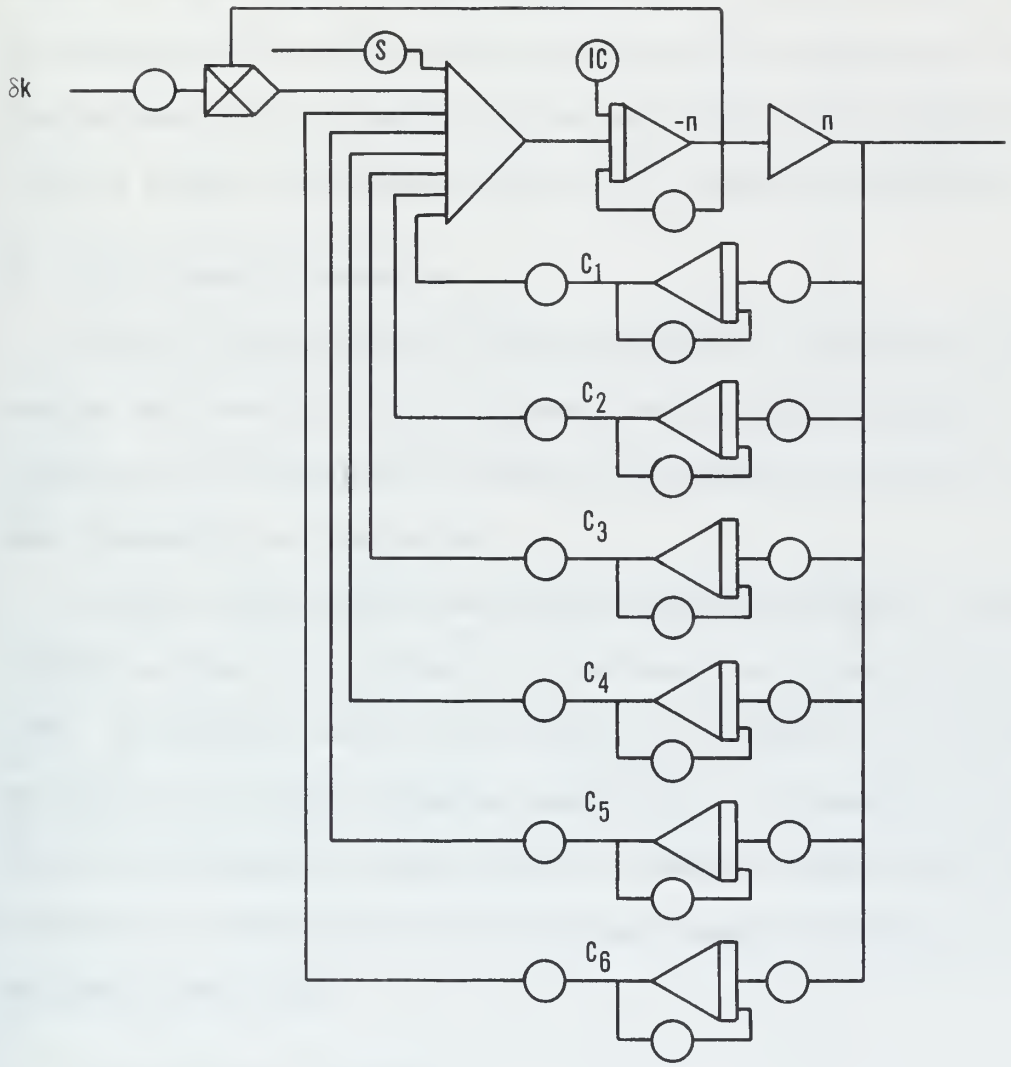

Fig. 14 Analog Computer Program for Solution of the Reactor Kinetic Equations

there would be little precision at low temperatures and power levels. Another method of solution is to program the simulation equations in their original form and to rescale the equations several times during the simulation, as necessary to maintain the variables within the analog computer limitations. With analog computer circuitry, this is difficult to accomplish, but it is a fairly simple matter using digital logic. The combination of an analog computer and digital logic suggest use of a hybrid computer for the solution of the simulation equations.

B. Hybrid Computer Solution

A hybrid computer reactor kinetic simulation program has been developed by Patterson (25). This program uses a hybrid automatic rescaling technique and automatically rescales the simulation variables as necessary throughout the simulation.

A simple hybrid simulation program utilizing automatic rescaling techniques could be developed for simulating the pulsed response of the PSTR. The program could utilize the remote terminal at the PSU Reactor Facility for the Hybrid Computer Laboratory. An elementary version of this hybrid simulation program might be possible using digital logic circuitry in combination with the analog computer installation at the PSU Reactor Facility.

C. Digital Computer Solution

Numerous digital computer reactor kinetics codes have been evaluated by the Argonne Code Center (7). These codes are typically designed to simulate the general kinetic behavior of nuclear reactors, and often, the development of the codes involved considerable reactor theory and mathe-

matical techniques.

Another method for solving the simulation equations with a digital computer is to use a digital simulation language, e.g., DSL/90 in the IBM Share Library Program (33). A digital simulation language has elements that can perform each of the functions of the components in an analog computer program. The simulation language elements are arranged in a simulation program corresponding to an analog computer program. The variables are limited in magnitude only by the limitations of the digital computer.

In a digital computer simulation language, the process of integration is incremental rather than continuous as in an analog computer program. A numerical integration approximation such as the trapezoidal rule is used.

In this study, a digital computer simulation program was developed in which the simulation equations are solved by direct numerical intergration. Although some of the programming techniques are similar to the digital simulation language DSL/90 techniques, no knowledge of DSL/90 is required for complete understanding of the simulation program.

VI. APPLICATION OF THE SIMULATION PROGRAM

The simulation program was specifically developed to simulate the kinetic behavior of a pulsed reactor. The program is described in Appendix A. In this section, some of the techniques and possible applications of the program are discussed.

In order to accurately simulate a reactor pulse, the shape of the reactivity pulse must be determined. The value of the pulsed reactivity at specific intervals of time are used as input to the program in the form of punched cards. A typical pulse can be accurately simulated by a stepwise approximation of the reactivity pulse, using about one hundred step intervals. If the reactivity pulse shape can be described by an analytic expression, the punched card data can be easily computed. If only a polygonal approximation is known, the punched card data can be compiled by linear interpolation.

After the reactivity pulse has been accurately simulated, the actual pulse repetition frequency, pulse width, and other reactor parameters are read in as punched card data. The required format for input data is described in Appendix A.

A typical application of the simulation program would be to determine the mean power produced in a repetitive pulse mode of operation. From the heat balance equation in Section II-D,

$$\frac{dT}{dt} = \eta\rho - \gamma(T-25)$$

During steady state operation,

$$\frac{dT}{dt} = 0$$

and solving for the steady state power,

$$P = \frac{\gamma}{\eta}(T-25)$$

The steady state power, as a function of temperature, is shown in Figure 15.

During high frequency repetitive pulse operation, after an initial transient, the reactor temperature remains essentially constant. The temperature change during a single pulse is less than one degree. The mean power in this mode of operation equals the steady state temperature corresponding to this constant temperature.

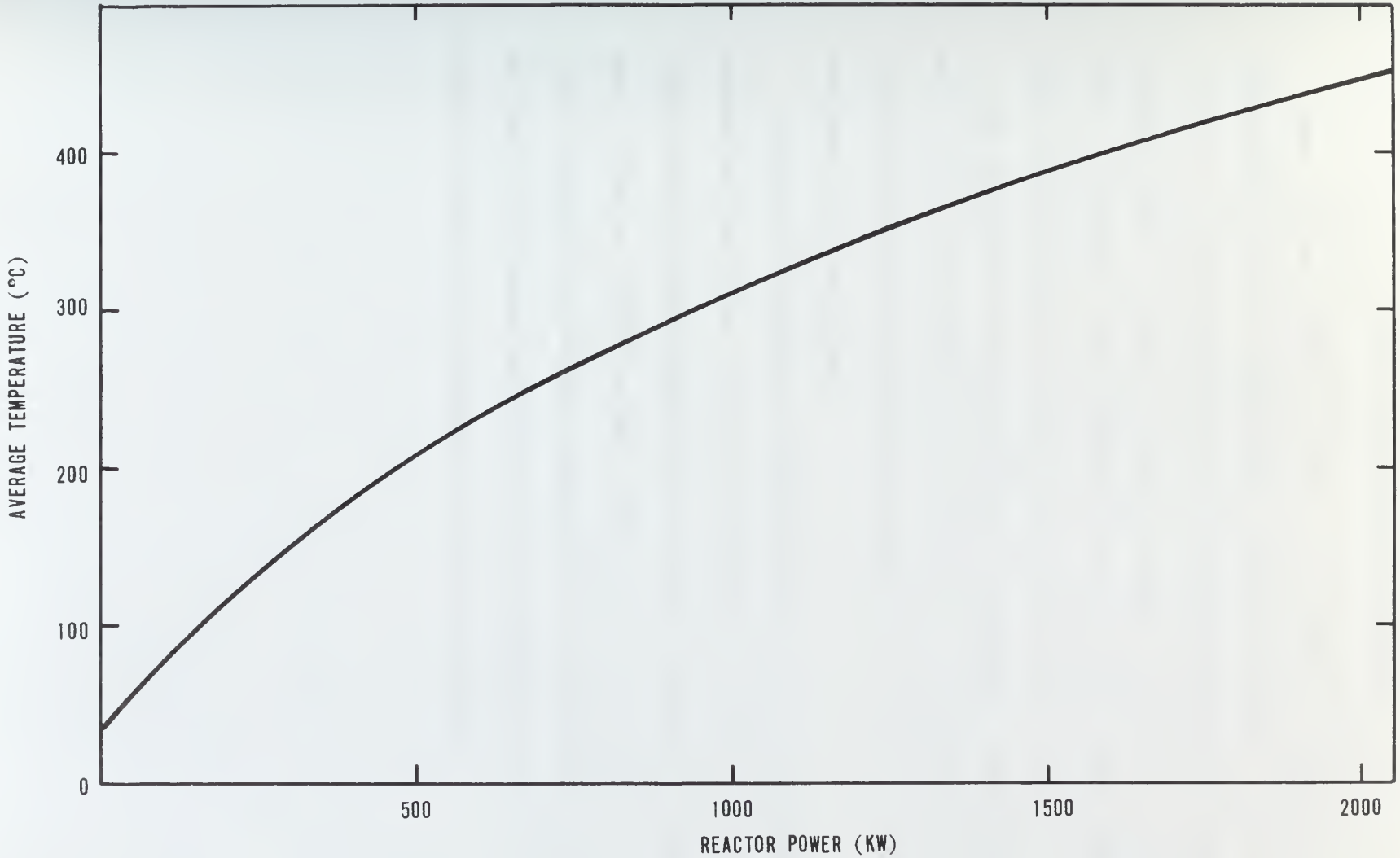

FIG. 15 Steady State Average Reactor Temperature Versus Reactor Power

VII. ERROR AND UNCERTAINTY

There are two sources of error in the numerical computation. The trapezoidal rule is an approximate integration process. The error is decreased as time intervals in the integrating process are made smaller. A second source of error is computer roundoff. The error due to roundoff is increased as time intervals are made smaller, since more computations must be performed to cover a specific length of time.

The magnitude of these errors can be estimated. Simulation results were compared for different time intervals in the integration process. The difference in the results are a measure of the error involved in the numerical integration.

In Table 10, two sets of simulation data for four transient rod simulations are given. The first simulation set used a value for the time interval between computations of 0.0001 seconds. The second set used a value of 0.00005 seconds. Therefore, the second set of simulations required twice as many computations to simulate the same response. In nearly every case, the differences are negligible, and in all cases, the differences are no greater than about one percent.

Table 10

Numerical Integration Time Interval Comparison

	Reactivity Inserted (dollars) 1.71	2.00	2.41	2.68	Time Interval (seconds)
Peak power (megawatts)	215	381	723	1018	0.0001
	214	378	716	1006	0.00005
Time of peak power (milliseconds)	267	238	222	218	0.0001
	266	237	221	218	0.00005
Temperature at peak power (°C)	61.52	69.29	93.36	108.68	0.0001
	61.17	69.17	92.52	108.15	0.00005

To estimate the error in the numerical computation due to computer roundoff, simulation results were compared for which different numbers of significant figures were retained throughout the simulation. The simulation program was run on an IBM 360 model 67 computer in which two variable storage sizes are used. In "single precision" computation each variable occupied 4 bytes of computer storage with a capacity of about 7.2 significant decimal digits. In "double precision" computation each variable occupies 8 bytes of computer storage with a capacity of about 16.8 significant decimal digits. Two simulations were run for comparison, one in single precision and the other in double precision. The differences in the values of peak power, time of peak power, and temperature at peak power were less than one percent. All other simulations used double precision computation.

An uncertainty in the program is the actual value of the heat transfer coefficient for the PSTR. For this study, the value found for the University of Illinois TRIGA Mark II reactor by Kurstedt (20) was used. He used a double pulse reactivity measuring technique, and this technique

could be used to determine the heat transfer coefficient for the PSTR.

In the double pulse reactivity measuring technique, the reactor is initially pulsed with a known reactivity from a cold condition at a low power level (e.g., 10 watts). At a known time later, the reactor is again pulsed with the same rod worth, but at a higher temperature, due to the heat generated by the previous pulse. By comparing the two peak powers in the successive pulses, the negative reactivity due to thermal feedback can be determined from a plot of peak power versus reactivity inserted. From this known reactivity, and from the known negative temperature coefficient, the temperature at the time of initiation of the second pulse can be determined.

By repeating this process for various time intervals between the first and second pulse, a plot of temperature versus time can be determined which describes the heat transfer from the reactor fuel to the coolant. The heat transfer coefficient is then determined as the value which gives the best fit of the simulated temperature following a pulse to the experimentally determined plot of temperature versus time.

VIII. COMPARISON TO EXPERIMENTAL RESULTS

As explained in Section III-C the temperature dependence of the prompt negative coefficient of reactivity was determined by comparing simulated power pulses with experimentally determined power pulses measured by Sears (29). The peak power values in these power pulses are given in Table 11.

Table 11

Transient Rod Response of the PSTR

Reactivity Inserted ($)	Simulated Peak Power (megawatts)	Experimental Peak Power (Sears-29)(megawatts)
1.71	215	216
2.00	381	370
2.41	723	745
2.68	1018	1081

IX. SIMULATED REPETITIVE PULSE RESPONSE OF THE PSTR

Using the reactivity wheel pulse shape described in Section IV-B the response of the PSTR was simulated for reactivity pulse heights of three dollars and pulse repetition rates of 2 to 400 pulses per second. The simulation data are summarized in Table 12. Each simulation was initiated from an initial power of one megawatt and the corresponding steady state temperature of 310 degrees centigrade.

For each pulse repetition rate the initial reactivity was varied to determine the value required for a mean thermal power of one megawatt. With these parameters the repetitive pulse mode of operation could be continued indefinitely.

For simulations with mean thermal power values other than one megawatt the data indicate the parameter values after eight to ten pulses. Due to power and temperature transients these simulations do not indicate possible continuous modes of operation.

From the data in Table 12 it is evident that a large negative reactivity is required between pulses to maintain a mean thermal power of one megawatt. The effect of the negative reactivity is to reduce the background power between pulses to less than one megawatt by minimizing the multiplication of delayed neutrons. The initial reactivity level could be controlled by adjustment of the reactor control rods.

The necessary balance between control rod reactivity and the reactivity wheel pulse height suggests several possible startup procedures from an initial shutdown condition.

(1) With the reactivity wheel removed from the reactor core the reactor could be brought to an initial power of one megawatt.

Table 12

Simulated Reactivity Wheel Response of the PSTR[1]

	Pulse Repetition Rate (sec^{-1})								Initial Reactivity
	2	4	10	20	40	100	200	400	(dollars)
Mean thermal power (megawatts)			3.35[2]	1.81[2]	1.24	1.08	1.04	1.03	- 0.40
			2.40[2]	1.34[2]	0.975	0.886	0.873	0.860	- 0.60
			1.79[2]	1.08					- 0.80
			1.28	0.884					- 1.00
			0.951						- 1.20
			0.760						- 1.40
	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	-- [3]
Peak power (megawatts)			21.8	6.70	2.72	1.46	1.21	1.11	- 0.40
			14.9	4.87	2.15	1.22	1.01	0.932	- 0.60
			11.2	3.68					- 0.80
			7.28	2.88					- 1.00
			5.21						- 1.20
			3.70						- 1.40
	10.1	8.01	5.30	3.42	2.20	1.37	1.16	1.08	-- [3]
Ratio of peak to background power			24.2	8.82	3.58	1.78	1.33	1.16	- 0.40
			19.8	7.50	3.36	1.74	1.30	1.14	- 0.60
			18.6	6.58					- 0.80
			14.0	5.81					- 1.00
			11.45						- 1.20
			9.25						- 1.40
	28.1	20.5	11.52	6.34	3.40	1.76	1.32	1.15	-- [3]

Table 12 (Continued)

Simulated Reactivity Wheel Response of the PSTR[1]

	Pulse Repetition Rate (sec^{-1})								Initial Reactivity (dollars)
	2	4	10	20	40	100	200	400	
Width of power pulse at one half pulse height above background (milliseconds)			11	7	5	4	2	1	- 0.40
			10	7	5	3	2	1	- 0.60
			10	7					- 0.80
			10	7					- 1.00
			10						- 1.20
			10						- 1.40
	28	17	10	7	5	3	2	1	--[3]
Initial reactivity for equilibrium mean thermal power of one megawatt (dollars)	-1.74	-1.56	-1.18	-0.86	-0.58	-0.48	-0.44	-0.43	

[1]All simulations used a $3.00 reactivity pulse with the pulse shape shown in Figure 13.

[2]Temperature increases from the high mean thermal powers would limit operation in this mode to a few seconds because of residual negative reactivity due to temperature.

[3]Data is for an equilibrium mean thermal power of one megawatt which requires different initial reactivity for each pulse repetition rate.

Then with the wheel rotating at the desired speed, it could be moved into the reactor core while control rods were adjusted to maintain a mean thermal power of one megawatt.

(2) With the reactivity wheel initially rotating in the reactor core at the desired speed the reactor could be brought directly to a mean thermal power of one megawatt.

(3) With the reactivity wheel stationary in the reactor core the reactor could be brought to an initial power of one megawatt. Then, simultaneously, the wheel could be accelerated to the desired speed and the required initial negative reactivity could be established by control rod insertion.

In the opinion of the author, of the three startup procedures the first involves the least deviation from normal operating procedures. In that procedure the reactor startup is standard. The rotating wheel could then be moved into the core as slowly as desired so that the required control rod adjustment rate is minimized. The movement of the wheel into the core and the adjustment of the control rods would both be stepwise with the controlling parameter being a mean thermal power of one megawatt as indicated by the reactor power and temperature instrumentation. The power instrumentation would indicate the range of the oscillation caused by the rotating reactivity wheel. The temperature instrumentation would be stable since the temperature oscillation is less than one degree per cycle.

The second startup procedure would require bringing the reactor power from shutdown levels to the megawatt range with a reactivity oscillation of several dollars. The power instrumentation would sense the oscillation

and at low power levels the temperature instrumentation would respond slowly due to the small amount of heat generated. Therefore this procedure would require changing reactor power over many orders of magnitude in a condition where the instrumentation is neither stable nor precise.

The third startup procedure would require accurate timing of the control rod insertion with the start of the wheel rotation to prevent large temperature or power transients. Also the wheel must be rapidly brought to speed to prevent a large power excursion. At very low speeds the response of the reactor would be similar to the response caused by the ejection of the transient rod. This would require an asymptotic approach to an equilibrium condition that would require several minutes.

The first startup procedure would require a mechanism for moving the reactivity wheel in or out of the reactor core while the wheel was rotating and while the reactor was at a power level near one megawatt. One possible mechanism would be to mount the reactivity wheel supporting equipment on rails. The entire assembly could then be moved along the rails with a travelling nut and a lead screw driven either electrically or manually with a flexible shaft.

In each of the simulations summarized in Table 12 a reactivity wheel pulse height of three dollars was used. This value is the maximum reactivity of an experiment in the PSTR by the limit established in the PSTR technical specifications. For comparison the reactor response was simulated for a reactivity wheel pulse height of six dollars at a pulse repetition rate of 20 pulses per second with an initial power level of

one megawatt. The resulting response compared with the response to a three dollar pulse was an increase in the peak power of 25 percent, an increase in the ratio of peak to background power by a factor of three, and a decrease in the power pulse width of 25 percent. For larger pulse heights comparable changes would be expected.

In all simulations discussed previously the reactivity pulse was positive. For comparison a negative pulse of three dollars was simulated at a pulse repetition rate of 20 pulses per second with an initial power level of one megawatt. The result was a very wide power pulse of 40 milliseconds with a period of 50 milliseconds and a ratio of maximum to minimum power of 3.0.

X. CONCLUSIONS

The results of the simulated reactor responses described in the previous section indicated that the PSTR could be pulsed at frequencies of 2 to 400 pulses per second, with a reactivity pulsing mechanism capable of introducing reactivity pulses of three dollars at the desired frequencies, to produce power pulses of several megawatts. The background power level between pulses would be a factor of up to 30 less than the peak power, or about several hundred kilowatts. For larger reactivity pulses the ratio of peak power to background power could be increased.

There are limits to the size of reactivity pulses that can be considered. The maximum allowable total reactivity in an experiment in the PSTR is limited to three dollars by technical specifications. If this number could be extended there would be a practical limit to the positive reactivity at any time, due to safety considerations. If the reactivity pulsing mechanism were to fail in the position of maximum positive reactivity, the effect on the reactor would be identical to a reactivity pulse from the transient rod. Transient rod pulse sizes in TRIGA reactors are limited by the maximum fuel temperatures following a pulse. For the stainless steel clad TRIGA fuel elements installed in the PSTR, the maximum safe pulse size is about five dollars (8). This is the limit on the positive reactivity during a pulse. A pulse size of ten dollars would meet this restriction if the initial reactivity in the reactor were negative five dollars.

The reactor simulation program developed in this study is a simple

means for determining the reactor response to pulsed reactivity insertions. The results obtained for single pulses compare favorably with experiment and with results computed using the Fuchs-Nordheim model. In addition, the program can simulate the response to a series of reactivity pulses. At the same time, the program is simple enough for student use without previous experience beyond an introductory course in digital computer programming and a very elementary knowledge of reactor kinetics.

XI. BIBLIOGRAPHY

1. Abramowitz, M. and I. A. Stegun, Handbook of Mathematical Functions, National Bureau of Standards Applied Mathematics Series No. 55, U. S. Government Printing Office, Washington, D. C., May 1968.

2. Anan'ev, V. D., et al., "The Operational Experience and Development of Periodically Pulsed Reactors at Dubna," in Fast Burst Reactors, CONF-690102, December 1969.

3. Barrett, L. G., "The Effect of Reactivity Modulation Amplitude on Measurement of the Reactor Transfer Function," KAPL-M-LGB-10, 1956.

4. Bryant, L. T., et al., "Engineering Applications of Analog Computers," ANL-6319, February 1961.

5. Bryant, L. T., and L. C. Just, "Introduction to Electronic Analog Computing," ANL-6187, July 1961.

6. Bunin, B. N., et al., "Operating Experience with IBR Reactor, Its Use for Neutron Investigations and Its Characteristics on Neutron Injection from a Microtron," Proceedings of the Third United Nations International Conference on the Peaceful Uses of Atomic Energy, Geneva, 1964, Paper A/CONF. 28/P/324.

7. Butler, M. K., et al., "Argonne Code Center Compilation of Program Abstracts," ANL-7411 plus supplements 1, 2, and 3, January 1968.

8. Coffer, C. O., "Characteristics of Large Reactivity Insertions in a High-performance TRIGA Uranium-Zirconium Hydride Core," in Neutron Dynamics and Control, CONF-650413, May 1966.

9. Corran, E. R., "Analog Computer Model of a Research Reactor," AAEC/TM-498, 1969.

10. Franz, J. P., and N. F. Simcic, "Nuclear Reactor Startup Simulation," Trans. IRE, NS-4, p. 11, March 1957.

11. Geisler, G. C., "Design Manual and Safety Analysis for the Penn State TRIGA Reactor," March 1965.

12. General Atomic Division of General Dynamics, "TRIGA Mark III Reactor Description," GA-4339, December 1963.

13. Glower, D. G., Experimental Reactor Analysis and Radiation Measurement, McGraw-Hill, New York, 1965.

14. Hendrie, J. M., "Brookhaven Pulsed Fast Research Reactor," in Fast Burst Reactors, CONF-690102, December 1969.

15. Hoover, L. J., "The Thermal Cell Contribution to the Prompt Negative Temperature Coefficient of Reactivity of the Penn State TRIGA Reactor," Ph.D. Thesis, The Pennsylvania State University, 1968.

16. Hornyik, K., and M. E. Wyman, "Study of the Dynamics of a TRIGA Type Reactor," in Neutron Dynamics and Control, CONF-650413, May 1966.

17. IAEA Symposium on Pulsed Neutron Research at Karlsruhe, STI/PUB/104, 1965.

18. Keepin, G. R., Physics of Nuclear Kinetics, Addison-Wesley, Reading, 1965.

19. Kenney, E. S., Personal Communication, June 1970.

20. Kurstedt, H. A., "Applications of the Kinetics Equations Solved in Temperature to the Short Interval Series Pulsing of a TRIGA Reactor," Ph.D. Thesis, University of Illinois, 1968.

21. Lamarsh, J. R., Introduction to Nuclear Reactor Theory, Addison-Wesley, Reading, 1966.

22. Larrimore, J. A., "SORA", in Fast Burst Reactors, CONF-690102, December 1969.

23. Miley, G. H., "Utilization of a Pulsed Reactor for Research and Irradiation," CNM-R2, Vol. 2, p. 1645, 1967.

24. Pallette, J. R., "Space-Dependent Transfer Function Characteristics of the Pennsylvania State University TRIGA Reactor," Master's Thesis, The Pennsylvania State University, 1966.

25. Patterson, J. R., "A Multidecade Nuclear Reactor Simulation Using Hybrid Computer Automatic Rescaling Techniques," BNWL-1099, August 1969.

26. Qazi, M. N., "An Oscillator Measurement of the Prompt Neutron Lifetime in The Pennsylvania State University Reactor," Master's Thesis, The Pennsylvania State University, 1963.

27. Rivard, J. B., "Fuel Rod Transient Heat Transfer: Experiments and Analysis," SC-RR-67-19, June 1967.

28. Rogers, A. E. and T. W. Connelly, "Analog Computation in Engineering Design," McGraw-Hill, New York, 1960.

29. Sears, C. F., "Spatial Behavior of the Pennsylvania State University TRIGA Reactor (PSTR) During Pulsing," Ph.D. Thesis, The Pennsylvania State University, 1969.

30. Seminar on Intense Neutron Sources at Sante Fe, CONF-660925, September 1966.

31. Shultz, M. A., Control of Nuclear Reactors and Power Plants, McGraw-Hill, New York, 1961.

32. Symposium on Pulsed High Intensity Fission Neutron Sources at Washington, CONF-650217, February 1965.

33. Syn, W. M., and D. G. Wyman, "Digital Simulation Language DSL/90," IBM Share Program Library, November 11, 1966.

34. Weaver, L. E., System Analysis of Nuclear Reactor Dynamics, Rowman and Littlefield, New York, 1963.

35. West, G. B., "Calculated Fluxes and Cross Sections for TRIGA Reactors," GA-4361, August 1963.

36. West, G. B., "Kinetic Behavior of TRIGA Reactors," GA-7882, March 1967.

37. Whittemore, W. L., "A Multiple Pulsed TRIGA-Type Reactor for Neutron Beam Research," in Intense Neutron Sources, CONF-660925, September 1966.

38. Wyman, M. E., "Transient Temperature Measurements in Reactor Fuels," in Reactor Kinetics and Control, TID-7662, April 1964.

APPENDICES

APPENDIX A. SIMULATION PROGRAM DESCRIPTION

This appendix contains a description of the simulation program used in this report. The program description follows the format for library program descriptions at the PSU Computation Center as outlined in the specifications issued in January 1969. The name assigned to the program is KESDI for Kinetic Equations Solved by Direct Integration.

The program description is intended to be independent of the remainder of the report. Therefore, there is some repetition although this is limited to characteristics or use of the program or to typical results.

SOURCE LANGUAGE: FORTRAN PROGRAM CALLING NAME: KESDI

PROGRAM: September, 1970 WRITE-UP: December, 1970

Kinetic Equations Solved by Direct Integration

Main Program

PURPOSE

The purpose of the KESDI Program is to provide a simple model for simulating the dynamic behavior of a pulsed nuclear reactor. The general objective in writing the program was to provide a program with generality and flexibility and wide user availability. This required several specific objectives.

1. Provide complete generality of reactivity pulse input and reactor response output.
2. Limit the program language to elementary FORTRAN IV.
3. Require no knowledge of computer job control language (JCL).
4. Limit the reactor theory to the point reactor kinetics equations and simple heat transfer equations.

USE

The designed use of the KESDI Program is to simulate the response of a reactor to a pulsed reactivity insertion. Numerous codes exist for simulating general reactor dynamic behavior. These codes are described and evaluated in reference 1.

The KESDI Program differs from more general reactor dynamic codes since it is specifically designed to simulate pulsed reactors. The response of a pulsed reactor is in general a short transient followed by an asymptotic behavior until the initiation of a subsequent pulse. In the limit of high pulse frequencies the length of the asymptotic response interval approaches zero.

In the KESDI Program the time dependent reactivity pulse is generated in a segment of the program. The reactivity values at equal intervals of time are stored in an array. The reactivity insertion of the transient rod of the Pennsylvania State University TRIGA Reactor (PSTR) has been accurately determined from a high speed motion picture film of the transient rod motion and an integrated rod worth curve. Data for a transient rod pulse are supplied in punched card form. Any size pulse can be examined by limiting the simulated rod ejection time corresponding to the desired total reactivity insertion.

Pulse data is also supplied in punched card form for a reactivity wheel on which a reactor poison is mounted. The wheel is assumed to rotate in the centerline plane of a cylindrical reactor. The wheel radius is equal to the reactor radius, and it intersects the reactor core a distance equal to the radius. The time dependence of the wheel reactivity was calculated using a one group first order perturbation theory approximation. The reactivity wheel pulse data is normalized to unity at maximum reactivity. Any size pulse can be simulated by multiplying the normalized time dependent reactivity by the maximum reactivity in the pulse.

The reactor parameters used in the program are for the PSTR TRIGA

Mark III reactor. One exception is the heat transfer coefficient which was determined for the TRIGA Mark II reactor at the University of Illinois. The computed results would be approximately characteristic of any TRIGA type reactor and qualitatively characteristic of thermal reactors in general.

RESULTS

The computed results of the KESDI Program can be supplied in two forms of output: a tabulation of data (TAB mode), or a graphical representation (PLOT mode). Output can be selected in either or both forms in any single program execution.

At specified intervals of time values are computed for reactivity, reactor power level, neutron density, and reactor average temperature. In the TAB mode the values for each parameter are tabulated at specified recording intervals. The tabulated values have a precision of several significant digits. At the same time intervals the computed parameter values are stored for use in the PLOT mode. The PLOT mode consists of superimposed scaled plots of reactivity, power level, and temperature versus time. These parameters are also tabulated alongside the plot for convenience, but to only a few significant figures.

The data recording intervals are preset for periods of transient response and then reset to less frequent intervals during the asymptotic response. At subsequent pulses the data recording intervals are again reset to their original value. Because of the storage limitation of the parameter arrays for use in the PLOT mode, the number of data recording intervals is limited to 1000. If this number is reached before the simulation time limit is exceeded the simulation is terminated at that point

and an output message explains the reason for the termination. In that case, if the PLOT mode is selected, the plot will consist of 1000 points.

Following any termination of the simulation an output message gives the maximum value of the power level, with the corresponding time and temperature, and the maximum temperature with the corresponding time.

CAPABILITIES AND LIMITATIONS

(I) GENERAL

The KESDI Program is capable of simulating the pulsed reactor response to arbitrary reactivity insertions with certain limitations. The time dependent pulsed reactivity is described by an array of the reactivity values for successive increments of time. The array has dimension 221 which is the number of values in the transient rod reactivity pulse simulation. Any portion of this array may be used to simulate an arbitrary reactivity pulse. During the interval between the end of one reactivity pulse and the start of the next the pulsed reactivity is set equal to zero. At any time the total reactivity of the reactor is the algebraic sum of the reactivities due to the (1) oscillator or pulsing mechanism, (2) control rods, and (3) temperature. The temperature dependent reactivity is computed each time temperature is computed. The pulse reactivity can be updated either at each increment of time or, to reduce computer time, at each data recording interval. Therefore, the selections of computing time increments, data recording intervals, and pulsed reactivity updating intervals are related. The numerical integration accuracy is inversely proportional to the size of the time increments between computations. Therefore, the optimum computing time increment is determined by the accuracy desired and a reasonable limit

on computer execution time. Data recording intervals are selected as some convenient multiple of (e.g., 2 to 10 times) the computing time increment. The total number of data recording intervals is limited to 1000 as described above. The pulsed reactivity can be updated at each data recording interval. Therefore, the data recording interval must be less than the difference between the times corresponding to successive array elements. Otherwise the pulse simulation would be a coarse stepwise approximation. Typical values are given in Table 1.

Table 1

Typical Time Variable Assignments

Simulation	DELTAT (seconds)	RECORD (seconds)
Single pulse	0.0005	0.002
Repetitive pulses at 1 pulse per second	0.0001	0.001
Repetitive pulses at 50 pulses per second	0.00001	0.0001

The only other limitations on input parameters is that each of the initial values must be within the range of the plotting field if the PLOT mode is selected. Otherwise the program will execute normally, but the parameters will be off scale. The plotting scales are shown in Figure 1.

The only limitations during a simulation execution are that all parameters and their derivatives maintain reasonable values. This is ensured in most cases by a proper selection of the computing time increment as described above. Also, the reactor temperature must not exceed

POWER (MEGAWATTS)(*)

0	100	200	300	400	500	600	700	800	900	1000

TEMPERATURE (DEGEEES C)(C)

C	50	100	150	200	250	300	350	400	450	500

REACTIVITY (DOLLARS)(+)

-5.0	-4.0	-3.0	-2.0	-1.0	0.0	+1.0	+2.0	+3.0	+4.0	+5.0

FIGURE 1. PLOTTING SCALES

500°C. An example of when this limitation might be exceeded is the case of large repetitive pulses from criticality at high frequencies. The limitation is imposed by the array size of the tabulated temperature dependent temperature coefficient of reactivity and heat capacity. By using array sizes allowing temperatures up to 1000°C the computer storage requirements are increased considerably. In particular, this requires the use of the "large core region" at the PSU IBM 360/67 computing facility, which requires a lower priority run category. The limitation could be eliminated if necessary by extending the associated array dimensions from 500 to 1000, extrapolating the array data from 500° to 1000°, and altering the associated program segments.

A simulation is not limited by the maximum simulated time for low pulse frequencies since the data recording intervals are increased during asymptotic response. A simulation of a single pulse and the asymptotic response up to a simulated 30 minutes requires less than the maximum of 1000 recording intervals. At very high pulse frequencies the maximum simulated time is reduced since the response never becomes asymptotic and time intervals do not reset.

(II) STORAGE REQUIREMENTS

The total approximate storage requirement of the program is about 1,250,000 hexadecimal characters or about 90,000 bytes on the IBM System 360/67. No effort has been made to optimize storage requirements.

(III) TIMING CONSIDERATIONS

Execution time requirements are strongly dependent on the type of simulation program. During an asymptotic response period considerably less computation time is required than during a transient period. A

single pulse followed by an asymptotic response period of 30 minutes simulated time requires an execution time of 90 seconds. A sequence of 20 pulses during a period of one second requires an execution time of 60 seconds. Simulation runs using the WATFOR compiler require considerably more execution time. However, because of the good diagnostics of the WATFOR compiler, this is recommended for preliminary runs.

(IV) OUTPUT RECORDS

If only the TAB mode is selected the number of output records will be less than 1500 including the program listing of about 450 records. If the PLOT mode is selected to follow a tabulation of 1000 records about 4000 additional records will be produced since 4 output records are required for each recorded interval.

(V) COMPUTER SYSTEM

This program currently runs on the IBM 360, model 67 under OS/360. It has been compiled and tested using the FORTRAN G-level compiler.

METHOD AND DISCUSSION

The simulation equations are:

$$\frac{dn}{dt} = \frac{\rho - \bar{\beta}}{\Lambda} n + \sum_{i=1}^{6} \lambda_i C_i$$

$$\frac{dC_i}{dt} = \frac{\bar{\gamma} \beta_i}{\Lambda} n - \lambda_i C_i$$

$$\frac{dT}{dt} = \eta P - \gamma (T - T_0)$$

where

n = neutron density (n/cm^3)

C_i = ith delayed neutron group effective precursor density ($atoms/cm^3$)

T = reactor temperature (°C)

T_o = coolant temperature (25°C)

P = reactor power (watts)

ρ = reactivity (Δ k/k)

$\bar{\beta}$ = effective total delayed neutron yield fraction

β_i = ith delayed neutron group yield fraction

λ_i = ith delayed neutron group decay constant (sec^{-1})

Λ = prompt neutron generation time (sec)

η = reciprocal heat capacity (°C/watt)

$\bar{\gamma}$ = average delayed neutron effectiveness

γ = heat transfer coefficient (sec^{-1}).

Expressing reactivity in units of Λ,

$$\dot{n} = (K - \bar{B})n + \sum_{i=1}^{6} \lambda_i C_i$$

$$\dot{C}_i = B_i n - \lambda_i C_i$$

$$\dot{T} = \eta P - \gamma(T - 25)$$

where

$$K = \frac{\rho}{\Lambda}$$

$$\bar{B} = \frac{\bar{\beta}}{\Lambda} = \frac{\bar{\gamma}\beta}{\Lambda} = \sum_{i=1}^{6} \frac{\bar{\gamma}\beta_i}{\Lambda} = \sum_{i=1}^{6} B_i$$

$$B_i = \frac{\bar{\gamma}\beta_i}{\Lambda}$$

$$\dot{n} = \frac{dn}{dt}$$

$$\dot{C}_i = \frac{dC_i}{dt}$$

$$\dot{T} = \frac{dT}{dt}$$

The reactor power level is proportional to the neutron density.

In FORTRAN notation the simulation equations are

NDOT = (K - SUBM)*N + LC(1) + LC(2) + LC(3) + LC(4) + LC(5) + LC(6)

CDOT(I) = B(I)*N - LC(I)

TDOT = ETA * POWER - GAMMA * (TEMP - 25)

POWER = POVERN*N

The simulation equations are numerically integrated using the trapezoidal rule.

The program definitions are listed in Appendix A. In the program description below, refer to the program listing in Appendix B. The data card formats are listed in Appendix C.

a. Read Input Parameters

The first data read is the reactivity pulse data. The array KPULSE then contains the reactivity values in dollars. Each array element is then converted to Δ k/k in units of the prompt neutron generation time GENTIM. The time limit of the pulse input is RLIMIT. For a transient rod pulse this is the time that the transient rod is dropped. For a reactivity wheel pulse RLIMIT is set equal to PLIMIT. The value of RLIMIT is used later in the program to compute the time at which time intervals can be reset during asymptotic response. PLIMIT is the period of time required to insert the reactivity pulse. For a transient rod pulse this is the transient rod ejection time. The ejection time versus reactivity pulse size is tabulated in Appendix D. For a reactivity wheel

pulse PLIMIT is the pulse width. If the reactivity wheel radius equals the reactor radius and the wheel intersects the reactor core a distance equal to the radius, then the pulse width PLIMIT equals the interval between pulse beginnings PLSINT divided by π, where $\pi = 3.1416$. The maximum value of the reactivity pulse is PULSE. Expressed as $\Delta k/k$ in units of GENTIM the maximum pulse reactivity is KPLS.

The next data card contains time and mode information. The three integer variables RESET, PLOTQ and TABQ are either 0 to 1, where NO = 0 and YES = 1. If RESET = YES, time intervals will reset when the time in a particular pulse sequence, SEQTIM, exceeds TSET. If TABQ = YES, the TAB output mode is selected. If PLOTQ = YES, the PLOT mode is selected. Otherwise both modes are ignored and the only output data are maximum power and temperature.

The next data card contains the initial values of reactivity in dollars, DELTAK, power in watts, POWER, and temperature in degrees centigrade, TEMP.

b. Initialize Variables

c. Print Initial Parameters

d. Determine Change Rates

The values of NDOT, CDOT(I), and TDOT are computed from the simulation equations.

e. Advance Time

TIME is advanced by an amount DELTAT and N, C(I), and TEMP are computed by the trapezoidal rule. Reactivity is computed to determine the effect of a temperature change. POWER is determined from N.

f. Check Recording Interval Time

If the time since the previous data recording interval INTTIM exceeds the length of a data recording interval RECORD then the parameters are recorded in the PLOT mode arrays. The variable PSTOP is set equal to INT to store the maximum value of INT for the PLOT mode.

g. Record Parameters

If TABQ = YES the parameters are tabulated. If the simulated time TIME exceeds the limit TLIMIT or if the recording interval counter INT equals 1000 the simulation is terminated. Otherwise, the pulsed reactivity is updated and the simulation continues.

h. Check Reset Time

If RESET = NO the reset feature is ignored. Otherwise, if SEQTIM exceeds TSET the time intervals DELTAT, RECORD and TSET are increased by a factor of 10. The array RESETS(J) stores the intervals at which resets occur. If SEQTIM exceeds PLSINT the time intervals are reset to their original values and a new pulse is begun. In the TAB mode a dashed line indicates a reset of time intervals.

i. Terminate the Simulation

Output messages state the reason for a normal termination and the maximum power, temperature, and corresponding times.

j. Begin Plotting Process

If PLOTQ = NO the program execution is terminated. Otherwise, scaled values of power, temperature, and reactivity are computed in the integer array PLOT(INT, INDEX). These values determine the position of the plotting symbols. The plotting process continues until INT = either PSTOP or 1000.

REFERENCES

1. Butler, M. K., et al., Argonne Code Center: Compilation of Program Abstracts, ANL-7411 and supplements 1, 2 and 3, January 1968.

```
C      DYNAMICS OF PSTR INCLUDING TEMPERATURE FEEDBACK
C
C      DATA CARD FORMATS
C      0-IPULSE-10-KPULSE-20
C      0-PLIMIT-10-PULSE-20-PLSINT-30
C      0-DELTAT-10-RECORD-20-ILIMIT-30-RESET-40-TSET-50-PLOTQ-60-TABQ-70
C      0-DELTAK-10-POWER-20-TEMP-30
C
C      SIMULATION EQUATIONS
C      NDOT  = (K-SUMB)*N + LC(1) + LC(2) + LC(3) + LC(4) + LC(5) + LC(6)
C      CDOT(I) = B(I)*N - LC(I)
C      POWER = POWERN*N
C      TDOT = ETA*POWER - GAMMA*(TEMP - 25)
C
C      DEFINITIONS
C      ATTAB(TEMP)  REACTIVITY TEMPERATURE COEFFICIENT
C      BETA(I)      I-TH DELAYED NEUTRON GROUP YIELD FRACTION
C      BETAEF       EFFECTIVE TOTAL DELAYED NEUTRON YIELD FRACTION
C      BETAT        TOTAL DELAYED NEUTRON YIELD FRACTION
C      B(I)         EFFECTIVE BETA(I) IN UNITS OF GENTIM
C      C(I)         I-TH DELAYED GROUP PRECURSOR CONCENTRATIONS
C      CDOT(I)      TIME DERIVATIVE OF C(I)
C      CDOTO(I)     PREVIOUS CDOT(I)
C      DELTAK       REACTIVITY IN UNITS OF BETA (DOLLARS)
C      DELTAT       TIME INTERVAL BETWEEN COMPUTATIONS
C      DELTO        ORIGINAL VALUE OF DELTAT BEFORE A TIME INTERVAL RESET
C      EFFECT       AVERAGE DELAYED NEUTRON EFFECTIVENESS
C      ETA          RECIPROCAL HEAT CAPACITY
C      ETATAB(TEMP) TABULATED RECIPROCAL HEAT CAPICITY
C      FRACT        FACTOR IN TABULATING TEMPERATURE COEFFICIENT
C      FULELS       NUMBER OF FUEL ELEMENTS IN REACTOR FUEL LOADING
C      GAMMA        HEAT TRANSFER COEFFICIENT
C      GAMMA1       CONSTANT TERM IN GAMMA
C      GAMMA2       TEMPERATURE DEPENDENCE OF GAMMA
C      GENTIM       PROMPT NEUTRON GENERATION TIME
C      H            DELTAT
C      HALFH        H/2
C      HTTEMP       TEMPERATURE AT WHICH HEAT TRANSFER INCREASES
C      I            COUNTER FOR DELAYED GROUPS
C      INDEX        PLOTTING INDEX FOR POWER, TEMPERATURE AND REACTIVITY
C      INT          COUNTER FOR RECORDED INTERVALS
C      INTTIM       TIME WITHIN A RECORDED INTERVAL
C      IPULSE       COUNTER FOR REACTIVITY PULSE DATA
C      IT           INTEGER TEMPERATURE FOR TABULATION INDEX
C      ITEMP(INT)   PLOTTING ARRAY FOR TEMPERATURE AS AN INTEGER
C      J            COUNTER FOR TIME INTERVAL RESETS
C      K            REACTIVITY IN UNITS OF GENTIM
C      KC           CONTROL REACTIVITY IN UNITS OF GENTIM
C      KO           OSCILLATOR REACTIVITY IN UNITS OF GENTIM
C      KPLS         TOTAL PULSE REACTIVITY IN UNITS OF GENTIM
C      KPULSE       TABULATED PULSE REACTIVITY IN UNITS OF GENTIM
C      KT           TEMPERATURE DEPENDENT REACTIVITY IN UNITS OF GENTIM
C      KTO          INITIAL VALUE OF KT
C      KTTAB(TEMP)  TEMPERATURE DEPENDENT REACTIVITY IN UNITS OF GENTIM
C      LAMBDA(I)    I-TH DELAYED NEUTROM GROUP DECAY CONSTANT
C      LC(I)        PRODUCT OF C(I) AND LAMBDA(I)
```

```
C      LSTAR          PROMPT NEUTRON LIFETIME
C      MAXPOW         MAXIMUM POWER
C      MAXTEM         MAXIMUM TEMPERATURE
C      MPTEMP         TEMPERATURE AT MAXIMUM POWER
C      MPTIME         TIME OF MAXIMUM POWER
C      MTTIME         TIME OF MAXIMUM TEMPERATURE
C      N              REACTOR NEUTRON DENSITY (THERMAL)
C      NDOT           TIME DERIVATIVE OF NEUTRON DENSITY
C      NDOTO          PREVIOUS NDOT
C      NO             NC
C      PDELK(INT)     PLOTTING ARRAY FOR DELTA K
C      PLIMIT         TRANSIENT ROD EJECTION TIME OR WHEEL PULSE WIDTH
C      PLOT(INT,INDEX) PLOTTING ARRAY
C      PLOTQ          INDICATOR TO SELECT PLOTTING MODE
C      PLSINT         TIME INTERVAL BETWEEN SEQUENTIAL PULSES
C      POVERN         RATIO OF POWER TO NEUTRON DENSITY
C      POWER          REACTOR POWER
C      PPOWER(INT)    PLOTTING ARRAY FOR POWER
C      PPOWMW(INT)    PLOTTING ARRAY FOR POWER IN MEGAWATTS
C      PSTOP          NUMBER OF DATA RECORDING INTERVALS
C      PTEMP(INT)     PLOTTING ARRAY FOR TEMPERATURE
C      PTIME(INT)     PLOTTING ARRAY FOR TIME
C      PULSE          TOTAL PULSE REACTIVITY
C      RECO           ORIGINAL VALUE OF RECORD BEFORE A TIME INTERVAL RESET
C      RECORD         TIME INTERVAL BETWEEN RECORDED DATA
C      RESET          INDICATOR TO SELECT RESET MODE
C      RESETS(J)      INDICATOR FOR RESET INTERVALS
C      RLIMIT         LIMIT OF TIME THAT TRANSIENT ROD IS WITHDRAWN
C      SEQTIM         TIME MEASURED BETWEEN SEQUENTIAL PULSES
C      SUMB           SUM OF THE B(I)
C      SYMBOL(INDEX) PLOTTING SYMBOL
C      TABQ           INDICATOR TO SELECT DATA TABULATION MODE
C      TDOT           TIME DERIVATIVE OF TEMPERATURE
C      TDOTO          PREVIOUS TDOT
C      TEMP           REACTOR TEMPERATURE
C      TEMPT          TEMPERATURE USED IN TABULATIONS
C      TIME           TIME
C      TLIMIT         MAXIMUM SIMULATED TIME
C      TSET           TIME THAT INTERVALS RESET DURING ASYMPTOTIC RESPONSE
C      TSETO          ORIGINAL VALUE OF TSET
C      YES            YES
C
```

APPENDIX B. PROGRAM LISTING

```
      REAL*8 ATTAB(500),BETA(6),BETAEF,BETAT,B(6),C(6),CDCT(6),CDOTO(6),
     1 DELTAK,DELTAT,EFFECT,ETATAB(500),FRACT,FULELS,GAMMA,GAMMA1,
     2 GAMMA2,GENTIM,H,HALFH,HTTEMP,INTTIM,K,KC,KO,KT,KTC,KTTAB(500),
     3 LAMBDA(6),LC(6),LSTAR,MAXPOW,MAXTEM,MPTEMP,MPTIME,MTTIME,N,NDOT,
     4 NDOTO,POVERN,POWER,RECORD,SUMB,TEMP,TEMPT,TDOT,TDOTC,TIME,TLIMIT,
     5 TSET
      REAL*8 KPLS,KPULSE(221),PLIMIT,PLSINT,PULSE,RLIMIT,SEQTIM
      REAL*8 PDELK(1000),PPOWER(1000),PPOWMW(1000),PTEMP(1000),
     1 PTIME(1000)
      INTEGER I,INT,IPULSE,IT,J,NC,RESET,RESETS(20)/20*0/,TABQ,YES
      INTEGER INDEX,ITEMP(1000),PLOT(1000,3),PLOTQ,PSTOP
      LOGICAL SYMBOL(3)/'*','O','+'/
C
C     REACTOR PARAMETERS
      LSTAR = 0.000038
      GENTIM = LSTAR
      LAMBDA(1) = 0.0124
      LAMBDA(2) = 0.0305
      LAMBDA(3) = 0.111
      LAMBDA(4) = 0.301
      LAMBDA(5) = 1.14
      LAMBDA(6) = 3.01
      BETA(1) = 0.000215
      BETA(2) = 0.001424
      BETA(3) = 0.001274
      BETA(4) = 0.002568
      BETA(5) = 0.000748
      BETA(6) = 0.000273
      BETAT = 0.006502
      BETAEF = 0.00700
      EFFECT = BETAEF/BETAT
      GAMMA1 = 0.02
      GAMMA2 = 0.000033
      FULELS = 88.4
      HTTEMP = 0.0
      POVERN = 0.0307
      NO = 0
      YES = 1
C
```

```
C       TABULATE RECIPROCAL HEAT CAPICITY
        WRITE (6,1001)
 1001 FORMAT ('1','  I     TEMP           ETA(TEMP)')
        DO 1 I = 1,500
        TEMPT = I
    1 ETATAB(I) = 1.0/((857.0 + 1.60*(TEMPT - 25.0))*FULELS)
C
C       TABULATE TEMPERATURE COEFFICIENT
        WRITE (6,1002)
 1002 FORMAT ('1','  I     TEMP         ALPHA T(TEMP)         KT(TEMP)')
        DO 2 I = 1,60
        FRACT = I/60.0
    2 ATTAB(I) = -(0.0000000 + FRACT*0.0000975)
        DO 3 I = 61,65
        FRACT = (I - 60)/5.0
    3 ATTAB(I) = -(0.0000975 + FRACT*0.0002000)
        DO 4 I = 66,100
        FRACT = (I - 65)/35.0
    4 ATTAB(I) = -(0.0001175 + FRACT*0.0000015)
        DO 5 I = 101,250
        FRACT = (I - 100)/150.0
    5 ATTAB(I) = -(0.0001190 + FRACT*0.0000030)
        DO 6 I = 251,500
    6 ATTAB(I) = -0.0001220
        DO 12 I = 1,500
        TEMPT = I
       KTTAB(I) = ATTAB(I)*TEMPT/GENTIM
   12 CONTINUE
C
```

```
C      READ INPUT PARAMETERS
C
C      ************************************
C      TRANSIENT ROD PULSE DATA
       DO 1000 IPULSE = 1,221
 1000 READ (5,2000) JPULSE, KPULSE(JPULSE)
 2000 FORMAT (I1C,F1C.2)
       DO 2001 IPULSE = 1,221
 2001 KPULSE(IPULSE) = KPULSE(IPULSE)*BETAEF/GENTIM
C
       RLIMIT = 1.8C
       READ (5,20C2) PLIMIT,PULSE,PLSINT
 2002 FORMAT (F10.2,F1C.2,F1C.2)
       KPLS = PULSE*BETAEF/GENTIM
       WRITE (6,2003) PLIMIT,RLIMIT
 2003 FORMAT ('1','TRANSIENT ROD EJECTION TIME LIMIT IS ',F1C.6,
     1 ' AND ROD DROP BEGINS WHEN TIME IS ',F1C.6)
       WRITE (6,2004) PULSE,PLSINT
 2004 FORMAT ('0','PULSED REACTIVITY IS ',F1C.6,' DOLLARS AT INTERVALS O
     1F ',F12.6,' SECONDS')
C      ************************************
C
       READ (5,1003) DELTAT,RECORD,TLIMIT,RESET,TSET,PLOTQ,TABQ
 1003 FORMAT (2F10.8,F10.4,I1C,F10.4,I10,I1C)
       WRITE (6,1004) DELTAT,RECORD,TLIMIT
 1004 FORMAT ('P','DELTA T IS  ',F10.6,', THE LENGTH OF A RECORDED INTER
     1VAL IS  ',F1C.6,', AND THE TIME LIMIT IS  ',F10.4)
       IF (RESET.EQ.NO) GO TO 13
       WRITE (6,1005) TSET
 1005 FORMAT ('0','TIME INTERVALS WILL RESET WHEN TIME = ',F8.4,
     1 ' UNLESS A SUBSEQUENT PULSE OCCURS FIRST.')
       GO TO 14
   13 WRITE (6,1006)
 1006 FORMAT ('0','TIME INTERVALS WILL NOT RESET')
   14 WRITE (6,1007) HTTEMP
 1007 FORMAT ('0','HEAT TRANSFER INCREASES WHEN TEMPERATURE EXCEEDS ',
     1 F6.2)
       READ (5,1008) DELTAK,POWER,TEMP
 1008 FORMAT (3F10.8)
       WRITE (6,1C09)
 1009 FORMAT ('P','INTERVAL',8X,'TIME',8X,'DELTA K',5X,'NEUTRON LEVEL',
     1 3X,'POWER LEVEL',2X,'TEMPERATURE')
       WRITE (6,101C)
 1010 FORMAT (1X,14X,'(SECONDS)    (DOLLARS)    (N/CUBIC CM)      (WATTS
     1)    (DEGREES C)')
C
```

```
C       INITIALIZE VARIABLES
        INT = 0
        TIME = 0.0
        SEQTIM = 0.0
        INTTIM = 0.0
        H = DELTAT
        HALFH = H/2.0
        DELTO = DELTAT
        RECO = RECORC
        TSETO = TSET
        J = 0
        DO 15 I = 1,6
   15 B(I) = EFFECT*BETA(I)/GENTIM
        SUMB = B(1) + B(2) + B(3) + B(4) + B(5) + B(6)
        TDOT = 0.0
        TDOTO = C.0
        MAXTEM = 0.0
        MAXPOW = 0.0
        N = POWER/POVERN
        DO 16 I = 1,6
        LC(I) = B(I)*N
   16 C(I) = LC(I)/LAMBDA(I)
        DO 17 I = 1,6
        CDOT(I) = 0.C
   17 CDOTO(I) = B(I)*N - LC(I)
        IT = TEMP + C.CC1
        GAMMA = GAMMA1 + GAMMA2*TEMP
        IF (TEMP.LT.HTTEMP) GAMMA = C.0
        ETA = ETATAB(IT)
        TDOTO = ETA*POWER - GAMMA*(TEMP - 25.0)
        KT = KTTAB(IT)
        KTO = KT
        KC = DELTAK*BETAFF/GENTIM
        KO = 0.0
        K = KT + KC + KO - KTC
        NDOTO = (K-SUMB)*N + LC(1) + LC(2) + LC(3) + LC(4) + LC(5) + LC(6)
C
C       PRINT INITIAL PARAMETERS
        GO TO 23
C
```

```
C       DETERMINE CHANGE RATES
   18 NDOT  = (K-SUMB)*N + LC(1) + LC(2) + LC(3) + LC(4) + LC(5) + LC(6)
      DO 19 I = 1,6
   19 CDOT(I) = B(I)*N - LC(I)
      GAMMA = GAMMA1 + GAMMA2*TEMP
      IF (TEMP.LT.HTTEMP) GAMMA = 0.0
      ETA = ETATAB(IT)
      TDOT = ETA*POWER - GAMMA*(TEMP - 25.0)
C
C       ADVANCE TIME
      TIME = TIME + DELTAT
      SEQTIM = SEQTIM + DELTAT
      INTTIM = INTTIM + DELTAT
C       NEUTRON DENSITY
      N = HALFH*(NDOTO + NDOT) + N
      NDOTO = NDOT
C       DELAYED NEUTRON PRECURSOR LEVELS
      DO 20 I = 1,6
      C(I) = HALFH*(CDOTO(I) + CDOT(I)) + C(I)
      LC(I) = LAMBDA(I)*C(I)
   20 CDOTO(I) = CDOT(I)
C       TEMPERATURE
      TEMP = HALFH*(TDOTO + TDOT) + TEMP
      TDOTO = TDOT
      IF (TEMP.LE.MAXTEM) GO TO 21
      MAXTEM = TEMP
      MTTIME = TIME
   21 CONTINUE
      IT = TEMP + 0.001
      IF (IT.GT.500) GO TO 27
C       REACTIVITY
      KT = KTTAB(IT)
      K = KT + KC + KO - KTO
C       POWER LEVEL
      POWER = POWERN*N
      IF (POWER.LE.MAXPOW) GO TO 22
      MAXPOW = POWER
      MPTIME = TIME
      MPTEMP = TEMP
   22 CONTINUE
C
C       CHECK RECORDING INTERVAL TIME
      IF (INTTIM.LT.RECORD) GO TO 18
      INT = INT + 1
      INTTIM = 0.0
      DELTAK = K*GENTIM/BETAEF
      PSTOP = INT
      PTIME(INT) = TIME
      PPOWER(INT) = POWER
      PTEMP(INT) = TEMP
      PDELK(INT) = DELTAK
      ITEMP(INT) = IT
C
C       RECORD PARAMETERS
   23 IF (TABQ.EQ.YES) WRITE (6,1011) INT,TIME,DELTAK,N,POWER,TEMP
 1011 FORMAT (4X,I5,F14.6,F14.8,D16.6,D16.6,F10.4)
      IF (TIME.GT.TLIMIT) GO TO 25
      IF (INT.EQ.1000) GO TO 26
C
```

```
C     ******************************************
C     SIMULATE EJECTION MOTION OF TRANSIENT ROD
      IF (SEQTIM.LE.PLIMIT) KC = KPULSE((SEQTIM + 0.0011)/0.001)
      IF (SEQTIM.GT.PLIMIT.AND.SEQTIM.LE.RLIMIT) KC = KPLS
      IF (SEQTIM.LE.RLIMIT) GO TO 200
C     ******************************************
C     SIMULATE DROP MOTION OF TRANSIENT ROD
      IF (SEQTIM - RLIMIT.LT.0.5) KC = KPLS*(1.0 - (SEQTIM-RLIMIT)/0.5)
      IF (SEQTIM - RLIMIT.GE.0.5) KC = 0.00
  200 CONTINUE
C     ******************************************
C
C     CHECK RESET TIME
      IF (RESET.EQ.NC) GO TO 24
      IF (SEQTIM.LT.TSET) GO TO 24
C     DELTAT SHOULD NOT RESET UNTIL AT LEAST 2 SECONDS FOLLOWING A PULSE
      IF (SEQTIM.GT.RLIMIT + 2.0) DELTAT = 10.0*DELTAT
      H = DELTAT
      HALFH = 0.5*H
      RECORD = 10.0*RECORD
      TSET = 10.0*TSET
      J = J + 1
      RESETS(J) = INT
      WRITE (6,1012)
 1012 FORMAT ('+',8X,'_______________',/)
   24 IF (SEQTIM.LT.PLSINT) GO TO 18
      SEQTIM = 0.0
      DELTAT = DELTO
      RECORD = RECO
      TSET = TSETO
      J = J + 1
      RESETS(J) = INT
      WRITE (6,1012)
      GO TO 18
C
C     TERMINATE THE SIMULATION
   25 WRITE (6,1013) TIME,TLIMIT
 1013 FORMAT ('1','TIME = ',F10.4,' AND EXCEEDS TIME LIMIT = ',F10.4,
     1 '.  SIMULATION WAS TERMINATED.')
      GO TO 28
   26 WRITE (6,1017)
 1017 FORMAT ('1','NUMBER OF RECORDED INTERVALS EQUALS 1000.  SIMULATION
     1 WAS TERMINATED DUE TO ARRAY LIMITATION.')
      GO TO 28
   27 WRITE (6,1018)
 1018 FORMAT ('1','TEMPERATURE EXCEEDS PROGRAM CAPACITY.  SIMULATION WAS
     1 TERMINATED.')
   28 WRITE (6,1014) MAXPOW,MPTIME,MPTEMP
 1014 FORMAT ('0','THE MAXIMUM POWER WAS ',D16.6,' AT TIME = ',F10.6,
     1 ' AND AT A TEMPERATURE OF ',F8.4)
      WRITE (6,1015) MAXTEM,MTTIME
 1015 FORMAT('0','THE MAXIMUM TEMPERATURE WAS ',F8.4,' AT TIME =',F10.6)
C
```

```
C      BEGIN PLOTTING PROCESS
  100 IF (PLOTQ.EQ.NO) GO TO 105
C
      DO 101 INT = 1,PSTOP
      PLOT(INT,1) = PPOWER(INT)*100.0/10000000.0 + 0.001
      PLOT(INT,2) = PTEMP(INT)*100.0/500.0 + 0.001
      PLOT(INT,3) = PDELK(INT)*100.0/10.0 + 50.001
      PPOWMW(INT) = PPOWER(INT)/1000000.0
  101 CONTINUE
C
      WRITE (6,3000)
 3000 FORMAT ('P',40X,'POWER (MEGAWATTS)(*)')
      WRITE (6,3001)
 3001 FORMAT (1X,'0',8X,'100',7X,'200',7X,'300',7X,'400',7X,'500',
     1 7X,'600',7X,'700',7X,'800',7X,'900',6X,'1000')
      WRITE (6,3003)
 3003 FORMAT ('0',38X,'TEMPERATURE (DEGEEES C)(O)')
      WRITE (6,3004)
 3004 FORMAT (1X,'0',9X,'50',7X,'100',7X,'150',7X,'200',7X,'250',7X,
     1 '300',7X,'350',7X,'400',7X,'450',7X,'500')
      WRITE (6,3005)
 3005 FORMAT ('0',39X,'REACTIVITY (DOLLARS)(+)')
      WRITE (6,3006)
 3006 FORMAT (1X,'-5.0',4X,'-4.0',6X,'-3.0',6X,'-2.0',6X,'-1.0',7X,
     1'0.0',6X,'+1.0',6X,'+2.0',6X,'+3.0',6X,'+4.0',6X,'+5.0')
      WRITE (6,3007)
 3007 FORMAT ('0','.',9X,'.',9X,'.',9X,'.',9X,'.',9X,'.',9X,'.',9X,'.',
     1 9X,'.',9X,'.',9X,'.',' I  TIME  DELTA K POWER TEMP')
      J = 1
      DO 104 INT = 1,PSTOP
      WRITE (6,3008) INT,PTIME(INT),PDELK(INT),PPOWMW(INT),ITEMP(INT)
 3008 FORMAT ('S','.',9X,'.',9X,'.',9X,'.',9X,'.',9X,'.',9X,'.',9X,'.',
     1 9X,'.',9X,'.',9X,'.',I4,F9.3,F6.2,F7.1,I5)
      IF (INT.EQ.RESETS(J)) GO TO 102
      GO TO 103
  102 WRITE (6,3009)
 3009 FORMAT ('+','.....................................................
     1.....................................')
      J = J + 1
  103 CONTINUE
      DO 104 INDEX = 1,3
      IF (PLOT(INT,INDEX).EQ.  0) WRITE (6,4000) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  1) WRITE (6,4001) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  2) WRITE (6,4002) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  3) WRITE (6,4003) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  4) WRITE (6,4004) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  5) WRITE (6,4005) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  6) WRITE (6,4006) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  7) WRITE (6,4007) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  8) WRITE (6,4008) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.  9) WRITE (6,4009) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 10) WRITE (6,4010) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 11) WRITE (6,4011) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 12) WRITE (6,4012) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 13) WRITE (6,4013) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 14) WRITE (6,4014) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 15) WRITE (6,4015) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 16) WRITE (6,4016) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 17) WRITE (6,4017) SYMBOL(INDEX)
```

```
      IF (PLOT(INT,INDEX).EQ. 18) WRITE (6,4C18) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 19) WRITE (6,4C19) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 20) WRITE (6,4C20) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 21) WRITE (6,4C21) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 22) WRITE (6,4C22) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 23) WRITE (6,4C23) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 24) WRITE (6,4C24) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 25) WRITE (6,4C25) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 26) WRITE (6,4C26) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 27) WRITE (6,4C27) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 28) WRITE (6,4C28) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 29) WRITE (6,4C29) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 30) WRITE (6,4C30) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 31) WRITE (6,4C31) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 32) WRITE (6,4C32) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 33) WRITE (6,4C33) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 34) WRITE (6,4C34) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 35) WRITE (6,4C35) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 36) WRITE (6,4C36) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 37) WRITE (6,4C37) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 38) WRITE (6,4C38) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 39) WRITE (6,4C39) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 40) WRITE (6,4C40) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 41) WRITE (6,4C41) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 42) WRITE (6,4C42) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 43) WRITE (6,4C43) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 44) WRITE (6,4C44) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 45) WRITE (6,4C45) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 46) WRITE (6,4C46) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 47) WRITE (6,4C47) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 48) WRITE (6,4C48) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 49) WRITE (6,4C49) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 50) WRITE (6,4C50) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 51) WRITE (6,4C51) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 52) WRITE (6,4C52) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 53) WRITE (6,4C53) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 54) WRITE (6,4C54) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 55) WRITE (6,4C55) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 56) WRITE (6,4C56) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 57) WRITE (6,4C57) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 58) WRITE (6,4C58) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 59) WRITE (6,4C59) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 60) WRITE (6,4C60) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 61) WRITE (6,4C61) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 62) WRITE (6,4C62) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 63) WRITE (6,4C63) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 64) WRITE (6,4C64) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 65) WRITE (6,4C65) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 66) WRITE (6,4C66) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 67) WRITE (6,4C67) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 68) WRITE (6,4C68) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 69) WRITE (6,4C69) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 70) WRITE (6,4C70) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 71) WRITE (6,4C71) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 72) WRITE (6,4C72) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 73) WRITE (6,4C73) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 74) WRITE (6,4C74) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 75) WRITE (6,4C75) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 76) WRITE (6,4C76) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 77) WRITE (6,4C77) SYMBOL(INDEX)
```

```
      IF (PLOT(INT,INDEX).EQ. 78) WRITE (6,4078) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 79) WRITE (6,4079) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 80) WRITE (6,4080) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 81) WRITE (6,4081) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 82) WRITE (6,4082) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 83) WRITE (6,4083) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 84) WRITE (6,4084) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 85) WRITE (6,4085) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 86) WRITE (6,4086) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 87) WRITE (6,4087) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 88) WRITE (6,4088) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 89) WRITE (6,4089) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 90) WRITE (6,4090) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 91) WRITE (6,4091) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 92) WRITE (6,4092) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 93) WRITE (6,4093) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 94) WRITE (6,4094) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 95) WRITE (6,4095) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 96) WRITE (6,4096) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 97) WRITE (6,4097) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 98) WRITE (6,4098) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ. 99) WRITE (6,4099) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.100) WRITE (6,4100) SYMBOL(INDEX)
      IF (PLOT(INT,INDEX).EQ.101) WRITE (6,4101) SYMBOL(INDEX)
 4000 FORMAT ('+',     A1)
 4001 FORMAT ('+',  1X,A1)
 4002 FORMAT ('+',  2X,A1)
 4003 FORMAT ('+',  3X,A1)
 4004 FORMAT ('+',  4X,A1)
 4005 FORMAT ('+',  5X,A1)
 4006 FORMAT ('+',  6X,A1)
 4007 FORMAT ('+',  7X,A1)
 4008 FORMAT ('+',  8X,A1)
 4009 FORMAT ('+',  9X,A1)
 4010 FORMAT ('+', 10X,A1)
 4011 FORMAT ('+', 11X,A1)
 4012 FORMAT ('+', 12X,A1)
 4013 FORMAT ('+', 13X,A1)
 4014 FORMAT ('+', 14X,A1)
 4015 FORMAT ('+', 15X,A1)
 4016 FORMAT ('+', 16X,A1)
 4017 FORMAT ('+', 17X,A1)
 4018 FORMAT ('+', 18X,A1)
 4019 FORMAT ('+', 19X,A1)
 4020 FORMAT ('+', 20X,A1)
 4021 FORMAT ('+', 21X,A1)
 4022 FORMAT ('+', 22X,A1)
 4023 FORMAT ('+', 23X,A1)
 4024 FORMAT ('+', 24X,A1)
 4025 FORMAT ('+', 25X,A1)
 4026 FORMAT ('+', 26X,A1)
 4027 FORMAT ('+', 27X,A1)
 4028 FORMAT ('+', 28X,A1)
 4029 FORMAT ('+', 29X,A1)
 4030 FORMAT ('+', 30X,A1)
 4031 FORMAT ('+', 31X,A1)
 4032 FORMAT ('+', 32X,A1)
 4033 FORMAT ('+', 33X,A1)
 4034 FORMAT ('+', 34X,A1)
 4035 FORMAT ('+', 35X,A1)
```

```
4036 FORMAT ('+', 36X,A1)
4037 FORMAT ('+', 37X,A1)
4038 FORMAT ('+', 38X,A1)
4039 FORMAT ('+', 39X,A1)
4040 FORMAT ('+', 40X,A1)
4041 FORMAT ('+', 41X,A1)
4042 FORMAT ('+', 42X,A1)
4043 FORMAT ('+', 43X,A1)
4044 FORMAT ('+', 44X,A1)
4045 FORMAT ('+', 45X,A1)
4046 FORMAT ('+', 46X,A1)
4047 FORMAT ('+', 47X,A1)
4048 FORMAT ('+', 48X,A1)
4049 FORMAT ('+', 49X,A1)
4050 FORMAT ('+', 50X,A1)
4051 FORMAT ('+', 51X,A1)
4052 FORMAT ('+', 52X,A1)
4053 FORMAT ('+', 53X,A1)
4054 FORMAT ('+', 54X,A1)
4055 FORMAT ('+', 55X,A1)
4056 FORMAT ('+', 56X,A1)
4057 FORMAT ('+', 57X,A1)
4058 FORMAT ('+', 58X,A1)
4059 FORMAT ('+', 59X,A1)
4060 FORMAT ('+', 60X,A1)
4061 FORMAT ('+', 61X,A1)
4062 FORMAT ('+', 62X,A1)
4063 FORMAT ('+', 63X,A1)
4064 FORMAT ('+', 64X,A1)
4065 FORMAT ('+', 65X,A1)
4066 FORMAT ('+', 66X,A1)
4067 FORMAT ('+', 67X,A1)
4068 FORMAT ('+', 68X,A1)
4069 FORMAT ('+', 69X,A1)
4070 FORMAT ('+', 70X,A1)
4071 FORMAT ('+', 71X,A1)
4072 FORMAT ('+', 72X,A1)
4073 FORMAT ('+', 73X,A1)
4074 FORMAT ('+', 74X,A1)
4075 FORMAT ('+', 75X,A1)
4076 FORMAT ('+', 76X,A1)
4077 FORMAT ('+', 77X,A1)
4078 FORMAT ('+', 78X,A1)
4079 FORMAT ('+', 79X,A1)
4080 FORMAT ('+', 80X,A1)
4081 FORMAT ('+', 81X,A1)
4082 FORMAT ('+', 82X,A1)
4083 FORMAT ('+', 83X,A1)
4084 FORMAT ('+', 84X,A1)
4085 FORMAT ('+', 85X,A1)
4086 FORMAT ('+', 86X,A1)
4087 FORMAT ('+', 87X,A1)
4088 FORMAT ('+', 88X,A1)
4089 FORMAT ('+', 89X,A1)
4090 FORMAT ('+', 90X,A1)
4091 FORMAT ('+', 91X,A1)
4092 FORMAT ('+', 92X,A1)
4093 FORMAT ('+', 93X,A1)
4094 FORMAT ('+', 94X,A1)
4095 FORMAT ('+', 95X,A1)
```

```
 4096 FORMAT ('+', 96X,A1)
 4097 FORMAT ('+', 97X,A1)
 4098 FORMAT ('+', 98X,A1)
 4099 FORMAT ('+', 99X,A1)
 4100 FORMAT ('+',100X,A1)
 4101 FORMAT ('+',101X,A1)
  104 CONTINUE
  105 CONTINUE
      WRITE (6,1016)
 1016 FORMAT ('1')
      STOP
      END
********** THE FOLLOWING IS SAMPLE DATA **********
         1       0.00
         2       0.00
         .         .
         .         .
         .         .
         .         .
         .         .
       220       2.68
       221       2.68
0.161      2.00       2000.0
0.0005     0.003999991800.0              10.39999999          1          1
0.00       10.0       25.0
********** END OF SAMPLE DATA **********
```

TO SIMULATE THE REACTOR RESPONSE TO A REACTIVITY WHEEL RATHER THAN A TRANSIENT ROD, THE FOLLOWING PROGRAM SEGMENTS ARE EXCHANGED WITH THE CORRESPONDING SEGMENTS SHOWN IN THE PROGRAM LISTING.

```
C     ************************************
C     REACTIVITY WHEEL PULSE DATA
      DO 1000 IPULSE = 1,101
 1000 READ (5,2000) JPULSE, KPULSE(JPULSE)
 2000 FORMAT (I10,F10.2)
      READ (5,2001) PLIMIT,PULSE,PLSINT
 2001 FORMAT (3F10.2)
      DO 2002 IPULSE = 1,101
 2002 KPULSE(IPULSE) = KPULSE(IPULSE)*PULSE*BETAEF/GENTIM
      RLIMIT = PLIMIT
      WRITE (6,2003) PLIMIT
 2003 FORMAT ('1','REACTIVITY WHEEL PULSE WIDTH IS ',F10.6)
      WRITE (6,2004) PULSE,PLSINT
 2004 FORMAT ('0','PULSED REACTIVITY IS ',F10.6,' DOLLARS AT INTERVALS O
     1F ',F12.6,' SECONDS')
C     ************************************


C     ************************************
C     SIMULATE MOTION OF REACTIVITY WHEEL
      IF (SEQTIM.LT.PLSINT/3.1416) KO=KPULSE(314.16*SEQTIM/PLSINT + 1.0)
      IF (SEQTIM.GE.PLSINT/3.1416) KO = 0.0
C     ************************************
```

APPENDIX C. DATA CARD FORMAT

Card	Variable	Column	Type	Units	Example
Pulse data	IPULSE	1 - 10	Integer*	--	65
Pulse data	KPULSE	11 - 20	Real	$	1.20
1	PLIMIT	1 - 10	Real	sec.	0.220
1	PULSE	11 - 20	Real	$	3.00
1	PLSINT	21 - 30	Real	sec.	$5.9999 \approx 6.00$
2	DELTAT	1 - 10	Real	sec.	0.0001
2	RECORD	11 - 20	Real	sec.	$0.0009999 \approx 0.001$
2	TLIMIT	21 - 30	Real	sec.	20.0
2	RESET	40	Integer	--	0
2	TSET	41 - 50	Real	sec.	$0.39999 \approx 0.400$
2	PLOTQ	60	Integer	--	1
2	TABQ	70	Integer	--	1
3	DELTAK	1 - 10	Real	$	0.0
3	POWER	11 - 20	Real	watts	10.00
3	TEMP	21 - 30	Real	°C	25.0

*right justified

APPENDIX D. TRANSIENT ROD EJECTION TIME

Reactivity Pulse Size (dollars)	Ejection Time (milliseconds)
0.01	60
0.02	65
0.03	70
0.04	75
0.05	80
0.08	85
0.11	90
0.16	95
0.20	100
0.26	105
0.35	110
0.46	115
0.58	120
0.75	125
0.86	130
1.04	135
1.20	140
1.41	145
1.60	150
1.81	155
1.97	160
2.12	165
2.25	170

APPENDIX D. TRANSIENT ROD EJECTION TIME (Continued)

Reactivity Pulse Size (dollars)	Ejection Time (milliseconds)
2.36	175
2.46	180
2.52	185
2.57	190
2.62	195
2.64	200
2.65	205
2.66	210
2.67	215
2.68	220

Thesis 122699
C879 Cross
Modulation of reactor power in the Pennsylvania State TRIGA (PSTR) as a research tool.