Doctoral Dissertations   Graduate School

12-1984

Dynamic Behavior of Boiling Water Reactors

Jose March-Leuba

University of Tennessee - Knoxville

To the Graduate Council:

I am submitting herewith a dissertation written by Jose March-Leuba entitled "Dynamic Behavior of Boiling Water Reactors." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Nuclear Engineering.

<div style="text-align:right">R. B. Perez, Major Professor</div>

We have read this dissertation and recommend its acceptance:

P. F. Pasqua, B. R. Upadhyaya, D. G. Cacuci, R. C. Gonzalez

<div style="text-align:right">Accepted for the Council:
Carolyn R. Hodges</div>

<div style="text-align:right">Vice Provost and Dean of the Graduate School</div>

(Original signatures are on file with official student records.)
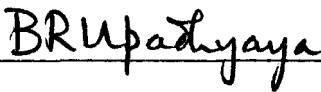
To the Graduate Council

I am submitting herewith a dissertation written by
José March-Leuba entitled "Dynamic Behavior of Boiling Water Reactors."
I have examined the final copy of this dissertation for form and
content and recommend that it be accepted in partial fulfillment of the
requirements for the degree of Doctor of Philosophy, with a major in
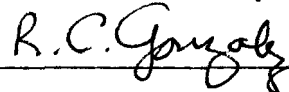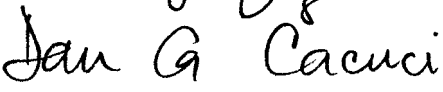Nuclear Engineering.

R. B. Perez, Major Professor

We have read this dissertation
and recommend its acceptance:

Accepted for the Council

The Graduate School

DYNAMIC BEHAVIOR OF BOILING WATER REACTORS

A Dissertation

Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

.

José March-Leuba

December 1984

to Emi

ACKNOWLEDGMENTS

ABSTRACT

A study of the basic processes involved in boiling water nuclear reactor dynamics is presented. The main emphasis of this research has been placed on the physical interpretation of these processes. It is shown that this type of reactors have two regimes of operation: linear, during normal operation, and nonlinear, if they become unstable due to the thermohydraulic feedback. Both of these regimes are studied using low-order physical models.

The main result obtained from the linear study is the pole-zero configuration of the reactivity-to-power transfer function. It is determined that three zeros and four poles are needed to properly represent this transfer function. Physical processes are identified with these transfer function features. Based on the understanding of these processes, an automated algorithm to estimate boiling water reactor stability from neutron noise measurements is developed and implemented as a computer code.

The causative mechanism leading to the appearance of the limit cycle in boiling water reactors is identified from the nonlinear study. The relationship between the different process variables during limit cycle oscillations is studied. It is shown that these oscillations could reach large amplitudes.

The stability of the limit cycle is also studied. It is shown that the amplitude of the limti cycle can become unstable and produce period-doubling pitchfork bifurcations which scale according

to Feigenbaum's universality theory. As a consequence of the

bifurcation process, aperiodic solutions of the deterministic reactor

equations are found to be possible.

Finally, nonlinear noise propagation is studied. A nonper-

turbative technique is developed for detecting the onset of linear

instability and the transition to the nonlinear regime.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

There are presently 72 commercial boiling water reactors (BWRs) either in operation or under construction in the western world; 37 of them are located in the United States.[1] Consequently a large effort has been devoted to the study of the BWR dynamic behavior under varied plant operating conditions. The work presented in this dissertation contributes to this ongoing effort. It deals with the study of the basic dynamic processes taking place in these reactors, with special emphasis being placed on the physical interpretation of these processes.

In view of the negative reactivity feedback from neutron moderation and heat transfer in a two-phase flow environment, BWRs are essentially stable machines in the sense that they regulate their own power without the need of external control systems. In some cases, however, instances of unstable behavior that lead into a nonlinear regime of operation have been observed. Thus, BWRs have two different regimes of operation: the linear or stable regime and the nonlinear or unstable regime. Both of them have been studied during the course of this research.

1.1 General BWR Dynamics

Nuclear power plants are in essence devices in which a fluid is caused to flow through a volume in which heat is generated by a nuclear fission chain reaction. In BWRs this fluid is water that enters the reactor core at temperatures close to saturation and partially vaporizes while flowing inside channel boxes containing uranium oxide fuel rods. In this type of reactors, the water has the dual role of coolant and moderator; it removes heat from the fission process, and at the same time helps maintain the fission reaction by moderating the energy of the fast fission-neutrons. This dual role is the cause for the strong moderator-density reactivity feedback which is characteristic of BWR operation.

Two distinct dynamic loops can be considered in BWRs: a neutronic loop, which controls the way in which heat is produced in the core, and a thermal-hydraulics loop, which in a sense controls the way in which heat is removed from the core by means of fluctuations in heat transfer rate, coolant density, and flow. The coupling between these two loops is through the Doppler and moderator-density reactivity coefficients. BWRs, thus, form a closed-loop system with negative feedback, which allows for self-regulation of the power level without the need for control systems during normal operation.

The relatively large magnitude of the density reactivity feedback causes the reactor power to oscillate around equilibrium following reactivity perturbations. At low flow and high power

conditions the magnitude of the reactivity feedback can become so large that the power oscillations following a small perturbation do not converge to the equilibrium point; that is, the reactor becomes unstable. The stability problem was recognized in the early experimetal BWR designs in the 1950's, but commercial BWRs were thought not to be susceptible to instabilities in view of their high operating pressure, which reduces the moderator density reactivity coefficient. Recent design changes, though, have increased the reactor power density and fuel heat transfer coefficient to a level in which instabilities become possible.

1.2 Survey of Previous Work in the Field of BWR Dynamics

There have been many studies performed on the dynamic behavior of BWRs. One of the first works reported is that of Dietrich and Layman[2-4] relative to the Borax experiment in 1953. During the 1950's and 1960's many studies were conducted while designing the present day BWRs; among them are those of Skinner,[5] Iriarte,[6] Beckjord,[7,8] Thie,[9] Akcasu,[10,11] Christensen,[12] Suda,[13] Garlid,[14] Zivi,[15] Margolis,[16] Jones,[17-22] Fleck,[23,24] Niemi,[25] and Charmichael.[26]

In recent times, there has been a continuing effort toward the study and modeling of BWR dynamic behavior;[27-37] however, the main contemporary studies on BWR dynamics have been directed towards the development of detailed computer codes to simulate the physical processes in the reactor. Two main categories of codes exist:

(a) transient codes to simulate large transients such as pipe breaks or turbine trips, and (b) stability codes, which are specifically tailored to the calculation of the stability margin of reload cores. Among the first type are RAMONA,[38] RELAP,[39] COBRA,[40] and RETRAN.[41] Among the second type are FABLE,[28] ODYSY,[42] and TOSDYN[43] used by the Geneal Electric Company, COTRAN[44] used by the Exxon Nuclear Company, PARADYN[45] used by Hitachi Ltd., NUFREQ[46,47] used in the Rensselaer Polytechnic Institute, and LAPUR[48,49] which was developed in the Oak Ridge National Laboratory and is presently used also by TVA. All the stability codes are based on linear frequency domain analysis except for TOSDYN, COTRAN, and PARYDYN which model the reactor non-linearities and solve the equations in the time domain.

The original experiments to study the reactor dynamics were rod-oscillator type tests,[50-52] for which a special control rod is made to oscillate at a particular frequency in a sinusoidal fashion. More recently, dynamic tests are performed by perturbing the reactor pressure using pseudorandom binary signals (PRBS).[53-59] The first series of this new type of tests was performed in the Peach Bottom reactor[53-56] to determine the reactor stability. These tests showed that this particular reactor was stable with a decay ratio of 0.5 at the most unstable operating condition (the decay ratio is a measure of the system stability; if the decay ratio is less than 1.0 the system is stable). The relatively high decay ratio obtained cast doubts about the stability of other reactors with higher power density. As a consequence, two more tests were performed, one of them in the

ASEA-ATOM Barseback reactor,[57] and the other in the Vermont Yankee reactor.[58] In these last two tests the reactor became unstable when operated at high power levels and natural circulation flow. An important result of these tests was the appearance of a limit cycle at the onset of unstable behavior, which limited the amplitude of the oscillations to about ±15% of the steady-state value.

The tests in the Peach Bottom and Vermont Yankee reactors have been extensively modeled by almost all existing codes in a benchmarking effort.[60-64] Most of the codes give satisfactory agreement with the test results in the linear (stable) range, showing that there is a good understanding about which processes are involved in linear BWR dynamics and about how to model them numerically. Although these numerical solutions do not provide a clear picture of the physical processes. However, the nonlinear range of BWR operation corresponding to linear unstability has been scarcely modeled or studied.

Another area of interest related to the work in this dissertation is the analysis of noise (stochastic) signals in BWRs. There are many publications in this field.[65-79] Among the uses of BWR neutron noise analysis that have been reported, the most important are: vibration monitoring,[69] bypass boiling detection,[69,75] in core void velocity measurements,[60,70,75] two phase flow parameters measurements,[69,70,72] and stability monitoring.[70,71,75,76] The latter one being the most relevant to this work; however, due to an apparent confusion between different researchers about the definition of decay ratio, this methodology yielded inconsistent results.

There are many publications in the area of nonlinear dynamics,[80-99] but the author was unable to find any references about studies of nonlinear stochastic phenomena in BWRs.

1.3 Motivation and Objectives

The current approach to BWR dynamic modeling has been based on the development of detailed and complex models of the reactor neutronics and thermal-hydraulics which are then implemented in large computer codes. The result of calculations performed with these codes is a series of numbers which define a transfer function in the case of the linear models[42,46-49] or a time trace of the reactor response to a perturbation in the case of the nonlinear models.[38-41,43-45] Most of the codes have proven to be reasonably accurate when applied to large commercial BWRs, but due to their complexity, a great part of the physical understanding is lost in the numerical calculation. Hence, there is a need for simpler, yet physically sound, models which provide tools allowing the analyst to achieve a clear understanding of the various physical phenomena. The motivation of this research is to obtain a deeper understanding of the BWR behavior in both the linear (stable) and nonlinear (unstable) regimes of operation. The specific objectives of this work are:

1.3.1 Model Development

To develop a reduced order physical model of the dynamics of a BWR. The main objective of this step is the determination of the

minimal model order (i.e., number of equations) needed to represent the reactor dynamics.

1.3.2 Linear Study

To use the model to study the linear dynamics of BWRs. This will include the identification of observable transfer function features (such as resonances, zeros, or break frequencies) with reactor parameters (such as fuel heat transfer coefficients, void sweep time, etc).

1.3.3 Stochastic Linear Study

To use the results from the linear analysis to study the possibility of applying noise analysis techniques to BWR parameter identification. This objective includes the development of an algorithm to identify the decay ratio of an operating BWR.

1.3.4 Nonlinear Study

To develop a nonlinear model of the BWR dynamics to study the experimentally observed limit cycles as well as its stability against changes in operating conditions and reactor parameters.

1.3.5 Stochastic Nonlinear Study

To study nonlinear noise propagation in nonlinear reactors. The goal of this work is to develop noise analysis techniques for the identification of the onset of the nonlinear regime.

1.4 Organization of the Text

This work has two main parts: the linear dynamics part, contained in Chapters 3, 4, and 5 and the nonlinear dynamics part, presented in Chapters 6, 7, and 8. An introduction to both parts is contained in Chapter 2.

Chapter 2 contains a description of the physical model. First, a description of the BWR architecture and of the dynamic processes involved is presented. The general equations are given in Section 2.2. An introduction to the problem of BWR stability and its relationship with the physical reactor parameters is presented in Section 2.3. Finally, section 2.4 contains a discussion of nonlinearities found in BWRs.

Chapter 3 contains an overview of some topics of linear dynamics theory related to the work presented in this dissertation. The concepts of transfer function and stability are presented. Chapter 4 documents the development and applications of a reduced order linear model. Chapter 5 describes an algorithm to measure the asymptotic stability of BWRs by using the normally occuring fluctuations in neutron density known as noise.

An introduction to the nonlinear part of this dissertation is contained in Chapter 6. Some basic concepts of nonlinear dynamics are described. Chapter 7 contains the development and qualification of a reduced order nonlinear model for BWR dynamics. The main characteristics of nonlinear BWR operation are studied by means of this simple model. In Chapter 8 a more complete model, which takes into

account the nonlinearities in the thermohydraulic feedback, is developed and applied to study nonlinear BWR dynamic behavior in more detail.

Chapter 9 contains the highlights of the work presented in this disssertation and some recommendations for future work.

1.5 Original Contributions

The original contributions of this dissertation to the field of BWR dynamics in general can be divided in three main categories:

1.5.1 Linear Dynamics

This part of the dissertation contains the first study known to the author of the pole-zero configuration of the reactivity-to-power transfer function of a commercial BWR. A consequence of this study is the development, based on a nodal synthesis technique, of a reduced-order linear model which is used to associate physical reactor processes with transfer function features, hence leading to a thorough understanding of the basic causative mechanisms which control BWR dynamics.

1.5.2 Stability Measurements Using Neutron Noise Analysis

Although several works have been published in this area previously, the work reported in this dissertation has produced several original contributions:

(a) The concept of asymptotic decay ratio has been developed; this concept is of great importance since, as shown in this work,

the apparent decay ratio (measured by most existing tech-
niques) does not provide the necessary information to guaran-
tee the reactor stability.

(b) Existing techniques to measure the decay ratio have been
improved to allow for the measurement of the asymptotic decay
ratio. In addition, two completely new techniques have been
developed for this purpose: the autocorrelation function
technique and the frequency domain pole search of the
autoregressive (AR) model.

(c) The concept of AR-consistent correlation has been developed to
improve the estimate of the asymptotic decay ratio. This con-
cept is also an original contribution.

(d) An algorithm for the evaluation of an error estimate for the
measured decay ratio has been developed. This error estimate
accounts for the stochasticity of the noise measurement and
can be used to determine the amount of data required for an
accurate determination of the decay ratio.

(e) A technique for the calculation of a confidence level of the
estimate has been outlined. This level represents the good-
ness of the fit. It is obtained on the basis of a priori
knowledge about general BWR dynamics and selfconsistency
checks.

1.5.3 Nonlinear Dynamics

The field of nonlinear BWR dynamics has been scarcely studied
previously. This dissertation is (to the knowledge of the author) the

first attempt to obtain an understanding of the basic physical
processes involved as opposed to numerically calculating the reactor
response for each particular set of conditions. In this sense, most
of the work reported here is an original contribution to this field.
The major results obtained from this research are:

(a) Nonlinearities become important when the linear stability
threshold is reached; thus, nonlinear modeling is required
when trying to describe the reactor dynamic behavior above
that threshold.

(b) Nonlinearities in BWRs manifest themselves through the
appearance of limit cycles. It has been shown in this work
that the limit cycles are caused by nonlinearities inherent to
the neutron field equations (the term ρn in the point kinetic
representation), rather than thermal-hydraulic charac-
teristics. It has been found that the amplitude of the
oscillations is very sensitive to the reactor operating
condition. Under some conditions, the limit cycles may have
large amplitudes that may cause unwanted reactor scrams or
fuel damage if the reactor is not shut down in time.

(c) The limit cycles are susceptible to instabilities as the
operating conditions or the reactor design parameters are
changed. As a function of these changes, the limit cycles may
sustain a phenomenon known as period-doubling bifurcations.
Further changes in the reactor parameters originates a cascade
of bifurcations which lead to a region of aperiodicity where

the reactor power oscillations are finite but not of periodic nature. This is the first time that such a behavior has been reported as possible in BWR operation.

(d) It is shown in this work that the cascade of bifurcations experienced by the reactor response behaves in the universal manner described by Feigenbaum's theory.[80-83] This is indeed a remarkable result considering the fact that such diverse systems as weather prediction equations or the transition from laminar to turbulent flow behave in the same universal manner.

CHAPTER 2

THE PHYSICAL MODEL

This chapter describes the general physical dynamic model of a
BWR. First a general description of the processes involved and the
typical configuration of this type of reactors are given. The general
equations for each process are presented in Section 2.2; these
equations will be used later in particular applications throughout
this dissertation. The concept of BWR stability, its causes, and its
effect on reactor operation is presented in section 2.3. Finally,
some nonlinear effects which have been observed in operating BWRs are
presented and the nonlinearities of the reactor equations are studied.

2.1 General Description of the Dynamics of a BWR

The general design of light water reactors is always the same,
in the sense that water circulates through the reactor core where a
nuclear fission reaction is maintained. The energy released by the
fission reaction is absorbed by the water which acts as coolant. At
the same time, the water acts as moderator by slowing down fast
fission-neutrons. Therefore, two different loops or dynamic processes
can be identified in the reactor: (a) the neutronic loop, which
controls the way in which heat is produced, and (b) the thermal-
hydraulic loop, which in a sense controls the way in which the heat is
removed from the core. The two loops are coupled via the reactivity
feedbacks due to the Doppler and density reactivity coefficients.

This concept is presented graphically in the block diagram of Figure 2.1.

In BWRs the water flows through individual channels or fuel assemblies. Each of these assemblies is surrounded by a channel wall which inhibits cross flow between neighboring assemblies. Older BWR fuel designs contained arrays of 7 x 7 cyclindrical fuel rods per assembly. Newer designs contain 8 x 8 or 9 x 9 arrays. typical dimensions of a channel are 13 x 13 cm wide and 411 cm high, of which only 365 cm (12 ft) have active fuel. The number of assemblies varies from 500 to 800 depending on the reactor. Each of these assemblies forms an independent flow path between the lower and upper plena.

The water flow enters the channel through the lower plenum at temperatures close to saturation. Typical inlet subcooling values (the difference between inlet and saturation temperatures) are 10 to 30 K, depending on the reactor operating condition. There is a small region at the bottom of the channel (typically 30 to 50 cm) in which no boiling occurs. The majority of the channel is occupied by the sucooled-boiling region (typically from 50 to 300 cm); in this region the water and steam bubbles are not in thermodynamic equilibrium and the liquid phase is slightly subcooled. The upper part of the channel is the bulk boiling region, which is charaterized for thermodynamic equilibrium between phases. Typical exit qualities vary from 0.05 to 0.3 and the exit void fractions from 40 to 80%, depending mainly on the power-to-flow ratio.
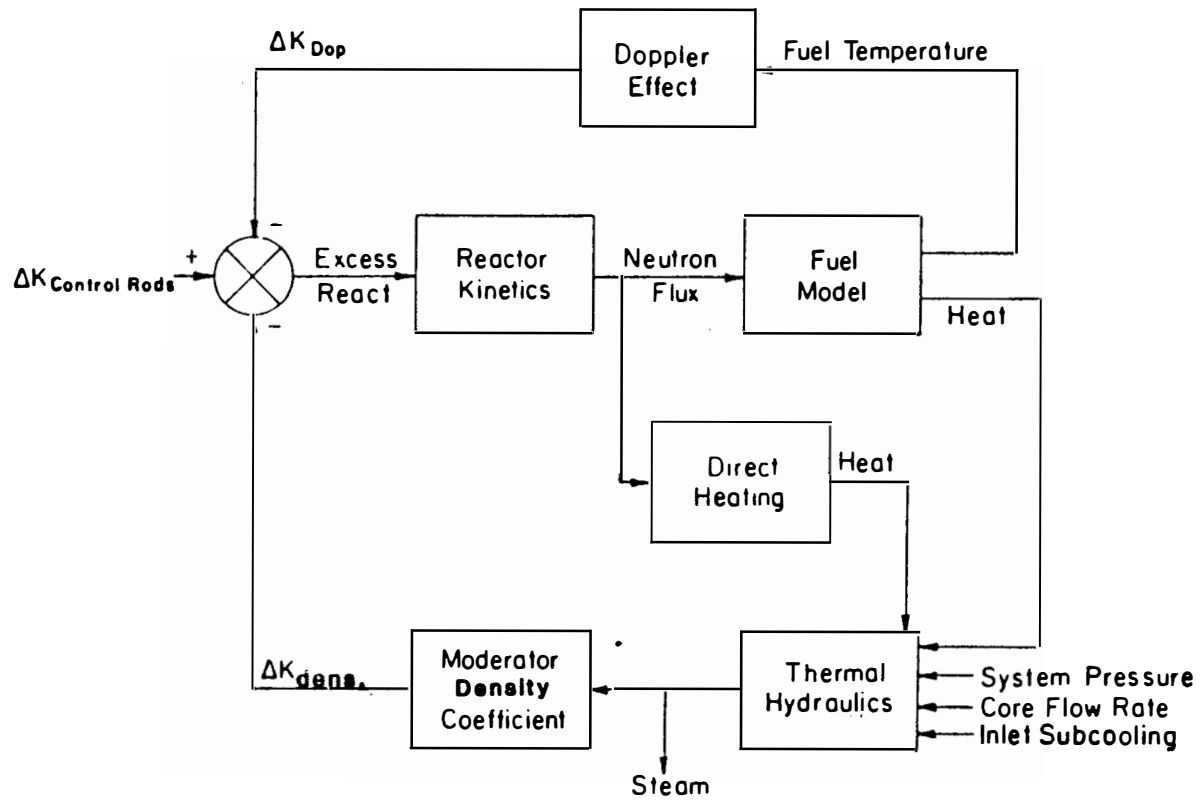
Figure 2.1 Block diagram of the dynamics of a BWR.

Given the low exit qualities characteristic of normal BWR operation, most of the water leaving the core is recirculated through what is called the recirculation loop. This loop includes the upper plenum, the steam separators and driers, the downcommer region, the jet pumps, and the lower plenum. For a typical exit quality of 0.05 at 100% rated conditions, the ratio of recirculated-to-inlet water is 20 to 1. The recirculation loop provides a coupling mechanism between the upper and lower plena pressures and the recirculation flow. Thus, variations in the channel thermal-hydraulics result in changes of inlet flow through pressure variations.

The energy source in the core comes from the fission chain reaction. Most of this energy is released inside the fuel rods, but a small fraction (about 3%) is deposited directly in the coolant by means of γ-ray absorption and neutron moderation. The heat transfer between fuel rods and coolant has its own dynamic characteristics which couple the neutron field to the channel thermal-hydraulics. Fuel rods are formed by a stack of cylindrical UO_2 fuel pellets surrounded by a zircaloy cladding. The gap between fuel and cladding introduces a noticeable resistance which affects the overall dynamic response of the fuel. The gap heat conductance increases towards the end of the fuel cycle due to the accumulation of gaseous fission products.

In summary, we have described the four major dynamic processes present in BWRs: (a) the neutron field, (b) the fuel heat transfer, (c) the channel thermal-hydraulics, and (d) the recirculation loop.

All of these processes act together forming the closed loop dynamics characteristic of BWR operation.

2.2 The Physical Equations Representing the Reactor Processes

As described in the previous section, there are four main processes which have to be modeled to represent the general BWR dynamic behavior: (a) the neutron field, (b) the fuel, (c) the channel, and (d) the recirculation loop. This section presents the general physical equations corresponding to these processes and some commonly used approximations. The coupling mechanisms between the different processes will also be needed. They are: (a) the reactivity feedback, which couples the thermal-hydraulics with the neutronics, and (b) the upper and lower plena, which couple the channel thermal-hydraulics with the recirculation loop dynamics.

2.2.1 The Neutron Field Equations

The purpose of this section is to derive the one-point reactor kinetic equations describing the neutronics in the present BWR model as well as an expression for the reactivity changes introduced by the thermohydraulic loop.

The neutronics loop is described by the Boltzman equation

$$\underline{V}^{-1} \frac{\partial \underline{\Psi}}{\partial t} = \underline{H} \ \underline{\Psi} \ , \tag{2-1}$$

where the vector $\underline{\Psi}$ has as its components the neutron flux, $\phi(E,\Omega,r,t)$, and the delayed neutron precursor concentrations, $C_j(r,t)$ $(j = 1,..,J)$, i. e.,

$$\underline{\Psi}^T = (\Phi, C_1, \ldots, C_j) \tag{2-2}$$

and where

$$\underline{V}^{-1} = \begin{bmatrix} 1/v & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdots & 1 \end{bmatrix} \quad , \tag{2-3}$$

with the matrix operator, $\underline{H}$, given by

$$\underline{H} = \begin{bmatrix} -\Omega\cdot\nabla+B & \lambda_1\chi_1 & \cdots & \lambda_j\chi_j \\ \beta_1 F_d & -\lambda_1 & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \beta_j F_d & 0 & \cdots & -\lambda_J \end{bmatrix} \quad , \tag{2-4}$$

where we introduced the following operators

$$B = S + (1-\beta)F \tag{2-5}$$

S = Scattering operator

$$= \int d\Omega' \int dE' \left\{ \Sigma_s(E',\Omega'|E,\Omega,r,t) - \frac{1}{4\pi}\Sigma_t(E',r,t)\delta(E-E') \right\} \tag{2-6}$$

F = Fission operator

$$= \int d\Omega' \int dE' \; \chi(E')\nu(E')\Sigma_f(E',r,t) \tag{2-7}$$

$$F_d = \int d\Omega' \int dE' \; \chi_j(E')\nu(E')\Sigma_f(E',r,t) \tag{2-8}$$

$\chi(E)$ = Prompt neutron fission spectrum

$\chi_j(E)$ = Delayed neutron fission spectrum

λ_j = Decay constant for the jth group of delayed neutron precursors

β_j = jth group of delayed neutron precursors fraction

$\beta = \sum \beta_j$

$\Sigma_f(E,r,t)$ = Macroscopic fission cross section

$\nu(E)$ = Neutron multiplicity

$\Sigma_s(E',\Omega'|E,\Omega,r,t)$ = Scattering Kernel

$\Sigma_T(E,r,t)$ = Macroscopic total cross section

The following initial and boundary conditions are satisfied

$$\Psi(E,\Omega,r,t=0) = I_0(E,\Omega,r) \quad , \tag{2-9}$$

where I_0, the vector of initial conditions, is given by

$$I_0^T = (f(E,\Omega,r),d_1(r),\ldots,d_j(r)) \tag{2-10}$$

and

$$\Phi(E,\Omega,r(S),t) = 0 \quad ; \quad (n\cdot\Omega<0) \quad , \tag{2-11}$$

where $r(S)$ are the position coordinates of the neutron boundaries and n is the unit normal vector.

The adjoint system is defined by the matrix equation

$$-\underline{V}^{-1}\frac{\delta\Psi^+}{\delta t} = \underline{H}^+\Psi^+ \tag{2-12}$$

where Ψ^+ is a vector whose components are the adjoint flux, Φ^+, and the adjoint delayed neutron precursors, C_j^+, i. e.

$$\Psi^{+T} = (\Phi^+,C_1^+,\ldots,C_j^+) \tag{2-13}$$

and where the matrix operator, $\underline{H}^+$, is obtained by transposing the forward matrix operator, $\underline{H}$, and adjoining each one of its elements containing differential and integral operators. Under this set of conditions, the forward and adjoint operators are related by the conmutation relation

$$\langle\Psi^{+T}|\underline{H}\Psi\rangle = \langle\Psi^T|\underline{H}^+\Psi^+\rangle \quad , \tag{2-14}$$

where the brackets symbolize integration over the phase space variables (energy, angle, and space) and where, for relation (2-14) to hold, the adjoint flux must satisfy the boundary condition below

$$\Phi^+(E,\Omega,r(S),t) = 0 \quad ; \quad (n\cdot\Omega>0) \quad . \tag{2-15}$$

Because of the time reversal implicit in equation (2-12) for the adjoint vector, Ψ^+, one must specify final conditions which we write in a general fashion as

$$\Psi^{+T}(E,\Omega,r,t=t_f) = I_F(E,\Omega,r) \quad , \tag{2-16}$$

with

$$I_F^T = (f^+(E,\Omega,r),d_1^+(E,r),\ldots,d_J^+(E,r)) \quad . \tag{2-17}$$

The reactivity changes are defined with respect to a critical reactor with the fuel temperature and moderator void fraction at their steady state values, T_{f_0} and α_0 respectively. The reference reactor is then defined by the steady state transport equation

$$\underline{H}_0 Y = 0 \tag{2-18}$$

where the vector Y has as its components the flux, ϕ, and the delayed neutron precursor concentrations, θ_j, i.e.,

$$Y^T = (\phi,\theta_1,\ldots,\theta_J) \quad . \tag{2-19}$$

The matrix operator, $\underline{H}_0$, corresponds to the matrix operator $\underline{H}$ evaluated at the equilibrium point (steady state) values of the reactor parameters. The adjoint reference reactor is in turn described by

$$\underline{H}_0^+ Y^+ = 0 \quad , \tag{2-20}$$

where the adjoint operator, $\underline{H}_0^+$, is obtained from $\underline{H}_0$ as previously shown.

We now rewrite the material properties of the altered reactor in terms of the properties of the reference reactor and the changes arising from altered plant conditions. The energy angle and time dependence are not explicitly written for the sake of clarity.

$$\Sigma_s = \Sigma_{0s} + \delta\Sigma_s \tag{2-21}$$

$$\Sigma_T = \Sigma_{0T} + \delta\Sigma_T \tag{2-22}$$

$$\Sigma_f = \Sigma_{0f} + \delta\Sigma_f \quad . \tag{2-23}$$

It should be emphasized that the changes in the system material properties are not necessarily restricted to be small. On account of the definitions (2-21) through (2-23), the operator H can be writen as

$$\underline{H} = \underline{H}_0 + \delta\underline{H} \quad , \tag{2-24}$$

with

$$\delta H = \begin{bmatrix} \delta S + \delta F & \cdots & 0 \\ \beta_1 \delta F_d & \cdots & 0 \\ \cdot & \cdots & \cdot \\ \cdot & \cdots & \cdot \\ \beta_j \delta F_d & \cdots & 0 \end{bmatrix} \tag{2-25}$$

and

$$\delta S = \int dE' \int d\Omega' \left\{ \delta\Sigma_s(E',\Omega'|E,\Omega) - \frac{1}{4\pi} \delta\Sigma_T(E') \delta(E-E') \right\} \tag{2-26}$$

$$\delta F = (1-\beta) \int dE' \int d\Omega' \chi(E')\nu(E')\delta\Sigma_f(E',r,t) \tag{2-27}$$

$$\delta F_d = \int dE' \int d\Omega' \nu(E')\delta\Sigma_f(E',r,t) \quad . \tag{2-28}$$

The derivation of the one-point reactor kinetics approximation is based on the assumption of separability between time and the phase-space variables. In the spirit of this assumption we write

$$\Psi(E,\Omega,r,t) = \underline{N}(t)Y(E,\Omega,r) \tag{2-29}$$

$$\Psi^+(E,\Omega,r,t) = \underline{N}^+(t)Y^+(E,\Omega,r) \quad , \tag{2-30}$$

where we introduced the diagonal matrices

$$\underline{N}(t) = \begin{bmatrix} n(t) & 0 & \cdots & 0 \\ 0 & D_1(t) & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 0 & \cdot & \cdots & D_J(t) \end{bmatrix} \tag{2-31}$$

and

$$\underline{N}^+(t) = \begin{bmatrix} n^+(t) & 0 & \cdots & 0 \\ 0 & C_1(t) & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdots & C_J(t) \end{bmatrix} .$$

(2-32)

Straight forward insertion of the "ansatz" (2-29) in the transport equation (2-1) followed by integration over the phase-space variables would indeed reduce that equation to a lumped parameter model in terms of the time variable alone. However it can be shown that such procedure does not yield optimal estimates of the quantities $n(t)$ and $D_j(t)$ ($j = 1,\ldots,J$) [in the sense that first order errors in the shape function Y lead to first order errors in the time-dependent quantities $n(t)$ and $D_j(t)$]. It is then important that the point-kinetics equations be obtained by methods which ensure that first order errors in the trial functions result in only second order errors in the estimate of the neutron and delayed precursors populations. Such a method can be formulated in terms of a variational principle.

The variational derivation of the point-kinetics approximation proceeds through the following steps:

(a) Formulation of a functional, L, of the forward and adjoint vectors, Ψ and Ψ^+, which is stationary (e.g., $dL = 0$), for the arbitrary variations $\delta\Psi$ and $\delta\Psi^+$.

(b) Insertion of the "ansatz" (2-29) and (2-30) into the stationary functional, L, followed by integration over the phase-space variables. This step generates the reduced functional L_1.

(c) The point-kinetics approximation is obtained by demanding that the reduced functional, L_1, be stationary.

The stationary functional, L, can be shown to be

$$L = \int_{t_0}^{t_f} dt \ \langle \Psi^{+T} | (\underline{V}^{-1} \frac{\delta \Psi}{\delta t} - \underline{H}\Psi) \rangle$$

$$+ \langle \Psi^{+T}(t_0) | \underline{V}^{-1}(\Psi(t_0) - I_{0q}) \rangle - \langle \Psi^T(t_f) | \underline{V}^{-1} I_f \rangle \quad , \qquad (2\text{-}33)$$

where t_0 is the initial time (beginning of the altered plant operation) and t_f is an arbitrary final time.

We now implement step (b), which on account of Equations (2-18) and (2-14) yields the following reduced functional, L_1

$$L_1 = \int_{t_0}^{t_f} dt \left\{ n^+(t) \left[M \frac{dn(t)}{dt} - P_0(\delta\rho - \beta)n(t) - \sum_{j=1}^{J} \lambda_j U_j D_j(t) \right] \right.$$

$$+ \sum_{j=1}^{J} C_j^+(t) \left[U_j \frac{d}{dt} D_j(t) - P_0 \beta_j n(t) + \lambda_j U_j D_j(t) \right] \Bigg\}$$

$$+ n^+(t_0) \left[Mn(t_0) - \langle \phi^+ | f \rangle \right] + \sum_{j=1}^{J} C_j^+(t) \left[U_j D_j(t_0) - B_j \right]$$

$$- Gn(t_f) - \sum B_j^+ D_j(t_f) \quad , \qquad (2\text{-}34)$$

where we introduced the following quantities

    M = Weighted neutron population

$$= \langle \phi^+ | 1/v\phi \rangle \ (\text{neutrons}) \qquad (2\text{-}35)$$

    P_0 = Weighted neutron production

$$= \langle \phi^+ | F_0 \phi \rangle \ (\text{neutrons/s}) \qquad (2\text{-}36)$$

    U_j = Weighted j^{th} delayed neutron precursor population

$$= \langle \theta_j^+ | \theta_j \rangle \ (\text{precursors}) \qquad (2\text{-}37)$$

    $\delta\rho$ = Reactivity change = $1/P_0 \ \langle \phi^+ | (\delta S + \delta F)\phi \rangle$

        (dimensionless) $\qquad (2\text{-}38)$

β_j = Effective delayed neutron fraction

$$= 1/P_0 \langle \phi^+ | \chi_j F_d \phi \rangle \beta_j \tag{2-39}$$

$$\beta = \sum_j \beta_j \tag{2-40}$$

B_j = Weighted initial conditions for the delayed neutron

precursors = $\langle \theta^+ | d_j^+ \rangle$ \hfill (2-41)

B_j^+ = Weighted final conditions for the adjoint delayed

neutron precursors = $\langle \theta_j | d_j^+ \rangle$ \hfill (2-42)

G^+ = Weighted final conditions for the adjoint flux

$$= \langle \phi | 1/vf^+ \rangle \quad . \tag{2-43}$$

At this point we carry out step (c) by taking variations in Equation (2-34) with respect to $n(t)$, $n^+(t)$, $D_j(t)$, and $C^+(t)$. By demanding that the functional, L, be stationary (i.e., $\delta L = 0$), for arbitrary variations of forward and adjoint neutron densities and precursor concentrations, we obtain the initial and final conditions

$$n(t_0) = \langle \phi^+ | f \rangle / M \tag{2-44}$$

$$D_j(t_0) = B_j / U_j \tag{2-45}$$

$$n^+(t_f) = G^+ / M \tag{2-46}$$

$$c_j^+(t_0) = B_j^+ / U_j \tag{2-47}$$

and the following equations

$$\frac{d}{dt} n(t) = \frac{1}{\Lambda} (\delta\rho - \beta) n(t) + \sum_{j=1}^{J} \lambda_j c_j(t) \tag{2-48}$$

$$\frac{d}{dt} c_j(t) = \frac{1}{\Lambda} \beta_j n(t) - \lambda_j c_j(t) \tag{2-49}$$

$$-\frac{d}{dt} n^+(t) = \frac{1}{\Lambda} (\delta\rho - \beta) n(t) + \frac{1}{\Lambda} \sum_{j=1}^{J} \beta_j c_j^+(t) \tag{2-50}$$

$$-\frac{d}{dt} c_j^+(t) = \lambda_j n^+(t) - \lambda_j c_j^+(t) \quad , \tag{2-51}$$

where we introduce the neutron generation time, Λ,

$$\Lambda = \frac{M}{P_0} = \frac{\langle \phi^+ | 1/v\phi \rangle}{\langle \phi^+ | F_0\phi \rangle} \qquad (2-52)$$

and we defined the delayed neutron precursor amplitudes as

$$c_j^+(t) = (U_j/M) \, D_j(t) \qquad (2-53)$$

The reactivity change is given from Equation (2-38), on account of equations (2-26) and (2-27) as an adjoint-flux weighted average

$$\delta\rho = \frac{1}{P_0} \langle \phi^+ | \{ \frac{\partial\rho}{\partial\alpha} \delta\alpha + \frac{\partial\rho}{\partial T_f} \delta T_f \} \phi \rangle \quad , \qquad (2-54)$$

where the void reactivity feedback and Doppler coefficient of reactivity are given respectively by

$$\frac{\partial\rho}{\partial\alpha} = \int dE' \int d\Omega' \{ \frac{\partial}{\partial\alpha} \Sigma_s(E',\Omega';E,\Omega,r,t)$$

$$- \frac{1}{4\pi} \frac{\partial}{\partial\alpha} \Sigma_T(E',r,t)\delta(E-E') \} \qquad (2-55)$$

$$\frac{\partial\rho}{\partial T_f} = (1-\rho) \int dE' \int d\Omega' \chi(E)\nu(E')\frac{\partial}{\partial T_f} \Sigma_F(E',r,t) \quad , \qquad (2-56)$$

where $\alpha(r,t)$ and $T_f(r,t)$ are the void fraction and fuel temperature process variables.

In summary, the use of the present variational technique allowed the derivation of a lumped parameter model for the neutronics loop, where the parameters are defined as bilinear averages of the forward and adjoint reference reactor fluxes; thus resulting into optimal estimates of the various reactor parameters.[109] In particular, Equation (2-54) gives an expression for the reactivity changes due to variations in the process variables; thus, defining the coupling between the neutronic and thermohydraulic dynamic loops in terms of the void and Doppler coefficients of reactivity.

2.2.2 The Fuel Equations

The equation describing the heat transfer process in the fuel is

$$\rho c_p \frac{\partial T}{\partial t} = Q + \nabla k \nabla T \quad , \tag{2-57}$$

where

Q = Volumetric rate of heat generation in fuel

ρ = Fuel density

C_p = Fuel heat capacity

K = Fuel heat transfer coefficient.

The fuel rods have cyclindrical geometry. Therefore, neglecting axial heat transfer, the equation becomes

$$\rho c_p \frac{\partial T}{\partial t} = Q + \frac{1}{r} \frac{\partial}{\partial r} \left(rk \frac{\partial T}{\partial r} \right) \quad . \tag{2-58}$$

An important component of the fuel dynamics is the gap between pellets and cladding. This gap can be modeled as a boundary condition. In this way, the heat transferred per unit surface, Q", is

$$Q''(r_g) = h_g (T(r_p) - T(r_g)) \quad , \tag{2-59}$$

where

h_g = Gap heat transfer coefficient

r_p = Pellet outside radius

r_g = Cladding inside radius.

Equation (2-59) serves as coupling between the fuel and cladding equations by taking into account that

$$Q''(r_0) = -k \frac{\partial T}{\partial r} \Big|_{r_0} \quad . \tag{2-60}$$

The boundary condition at the external cladding radius, r_c, can be handled by the equation

$$-k \left.\frac{\partial T}{\partial r}\right|_{r_0} = h(T(r_c) - T_c) \quad , \tag{2-61}$$

where

T_c = Coolant temperature

h = Convective heat transfer coefficient

2.2.3 The Channel Equations

The channel thermal-hydraulics can be modeled by mass, energy, and momentum balances. In the bulk boiling region they become[45]

$$\frac{\partial}{\partial t} \left[(1-\alpha)\rho_1 + \alpha\rho_g\right] + \frac{\partial G}{\partial z} = 0 \tag{2-62}$$

$$\frac{\partial}{\partial t} \left[(1-\alpha)\rho_1 h_1 + \alpha\rho_g h_g - p\right] + \frac{\partial}{\partial z} \left[(1-x)h_1 G + x h_g G\right] = Q' \tag{2-63}$$

$$\frac{\partial p}{\partial t} = -\frac{\partial G}{\partial t} - \frac{1}{2}\frac{\partial}{\partial z}\left[\frac{(1-x)^2 G^2}{\rho_1(1-\alpha)} + \frac{x^2 G^2}{\rho_g \alpha}\right] - \left[(1-\alpha)\rho_1 + \alpha\rho_g\right]g_c$$

$$-f \frac{\phi^2 \Omega G^2}{2\rho_1 D} - \sum \frac{Ki\phi^2 \Omega G^2}{2\rho_1} \delta(z - z_i) \quad , \tag{2-64}$$

where

G = Coolant mass flux in $Kg/m^2 s$

α = Void fraction

x = Steam quality

p = Pressure in N/m^2

ρ_g = Saturated steam density in Kg/m^3

ρ_1 = Saturated liquid density in Kg/m^3

h_g = Saturated steam enthalpy in J/Kg

h_1 = Saturated liquid enthalpy in J/Kg

f = Single-phase friction coefficient

ϕ^2 = Martinelli-Nelson correction factor

for two-phase flow pressure drop

Ω = Jones correction factor

D = Equivalent flow diameter in m

K_i = Concentrated pressure loss coefficients

Boundary conditions needed are an inlet void fraction, mass flow rate, and inlet pressure. A series of empirical correlations have to be used to determine the slip ratio and the two-phase friction multipliers.

A commonly used approximation[29] assumes that the pressure transients related to acoustic phenomena are very fast compared to the enthalpy transients. In this approximation, the term dp/dt in the energy balance equation is neglected by assuming that the system pressure remains constant during the transient so that the thermo-dynamic properties (such as specific density and enthalpy) remain constant. With this assumption the acoustic pressure waves propagate instantaneously through the system. The equation without acoustic phenomena becomes

$$\frac{\partial}{\partial t} \left[(1-\alpha)\rho_1 h_1 + \alpha \rho_g h_g \right] + \frac{\partial}{\partial z} \left[(1-x)h_1 G + x h_g G \right] = Q' \quad . \qquad (2\text{-}65)$$

This approximation greatly simplifies the problem of solving for the channel dynamic behavior, since it essentially uncouples the momentum equation from the energy and continuity balances. Numerically, the solution requires only a fraction of the computation

time necessary if the pressure waves were considered. A saving in computer memory is also obtained, since it reduces the dimensionality of the problem.

2.2.4 The Upper and Lower Plena Equations

BWRs are formed by a multitude of individual channels. All of them are connected through the upper and lower plena. A complete model of a BWR should contain several channel types, which are coupled together through these plena. Commonly they are modeled just as a boundary condition for the inlet and outlet channel pressures. If more than one channel is being modeled, the constant pressure boundary condition forces flow redistribution among the different channel types. In addition, the plena pressures serve as coupling between the channel thermal-hydraulics and the recirculation loop. Variations in the pressure drop across the channel result in changes of the inlet mass flow rate.

2.2.5 The Recirculation Loop Equations

The recirculation loop is formed by the upper plenum, the steam separators and dryers, downcomer, jet pumps, and lower plenum. A complete model of this system would be extremely complex. We will treat it as a single path of fluid with variable flow areas but constant mass flow rate and incompressible flow. It is convenient, thus, to rewrite momentum equation in terms of flow areas, $A(z)$, and a constant flow rate, ω, instead of mass flux, G.

$$-\frac{\partial p}{\partial t} = \frac{\partial}{\partial t}\left(\frac{\omega}{A}\right) + \frac{1}{A}\frac{\partial}{\partial z}\left(\frac{\omega^2}{\rho_l A}\right) + \rho_l g + \frac{\partial}{\partial z}\left(\Delta P_f\right) \ . \qquad (2\text{-}66)$$

ΔP_f represents the integrated friction losses can be considered proportional to ω^2. Note that A depends on z but not on t, whereas ω depends on t but not on z. Integrating over the path of the recirculation loop we obtain

$$P_1-P_2 = \left(\frac{L}{A}\right)\frac{\partial\omega}{\partial t} + \frac{\omega^2}{2\rho_1}\left(\frac{1}{A_2^2} - \frac{1}{A_1^2}\right) + \rho_1 g(z_2-z_1) + \Delta P_f \quad (2\text{-}67)$$

where we define

$$\left(\frac{L}{A}\right) = \int_1^2 \frac{1}{A(z)}\, dz \quad .$$

This equation, thus, yields the variation in inlet masss flow rate as a function of variations in channel pressure drop. Therefore, this equation eliminates one of the boundary conditions needed for the channel equations, since it supplies the inlet flow. Note that in normal BWR operation the upper plenum pressure is kept constant by the pressure regulator and, thus, variations in the channel pressure drop produce changes in the lower plenum absolute pressure.

2.3 BWR Stability

In general three different types of instabilities are considered to be possible in BWR operation:

  (a) Plant Instability. This is related to the reactor control
      systems and how the plant as a whole reacts to external
      disturbances such as a power load demand.

  (b) Reactivity Instability. BWRs form a closed-loop system, in
      the sense that power feeds back to the reactivity through both

the density and Doppler reactivity coefficients. Both of these feedbacks are negative, and the reactor is stable without the need of a control system under normal operating conditions; however, the density reactivity feedback is sufficiently strong to cause the reactor power to oscillate when a perturbation in reactivity is applied. Several parameters affect the gain and phase of this feedback, and over the critical value of these parameters the reactor becomes linearly unstable. This type of stability manifests itself as a diverging power oscillation with the core acting as a whole. Reactors are susceptible to the reactivity instability mostly when operated at reduced flow conditions, for instance during natural circulation.

(c) <u>Channel Thermal-Hydraulic Instabilities</u>. These types of instabilities are related to the dynamic characteristics of two phase flow in heated channels. They can occur in any type of heating system since the neutronic loop is not involved. Several types of these instabilities have been recognized;[100] the most important one is the density wave instability, which is caused by a downflow pressure drop. Other recognized types of channel instabilities are the buoyancy wave for natural circulation systems, the pressure or Helmholtz wave due to the compressibility of the flow, and out-of-phase multichannel oscillations. All these types of instabilities cause flow oscillations in a particular group of channels, which can be a

small number compared to the whole reactor. Therefore, these oscillations will be seen in the local power through the density reactivity coefficients, but the global or overall power will not be affected as much. For this reason, these types of instabilities are often called "local instabilities."

Several stability experiments have been performed recently.[50-59] These tests show that BWRs are susceptible to the reactivity instability when operated at low flow and high powers (for instance, 51% power and 32% flow in the case of Reference 58). For this reason, utilities are required to submit calculations relating to the stability of the reactor prior to any new fuel reloading.

As stated before, this type of instability arises from the dual role of water as coolant and moderator. As the power is increased, the heat transferred to the coolant increases and more steam voids are produced. The increase in voids, however, reduces the moderating power of the water and, thus, reduces the reactivity. This causes a reduction in power. This negative feedback process is what maintains the reactor power constant during normal operation without the need for a control system. It is a well known fact that negative feedback systems can be susceptible to oscillatory-type instablities if the feedback gain or the phase lag are increased. Thus, the negative void feedback may cause a reactor instability if it becomes too strong.

2.3.1 Parameters Affecting the Reactivity Stability

In this section we discuss the effects of various parameters on the gain and phase lag of the core's reactivity feedback dynamics. Most parameters affect both phase and gain of the reactivity feedback, and in some cases it is difficult to determine the impact of changes in a given parameter on core stability. In a BWR one parameter can hardly be changed without affecting others. In this section, however, we discuss the direct effect of each of the parameters in isolation, thus identifying their individual partial contributions to the integrated effect of any perturbation.

Steady state moderator density distribution. For illustrative purposes, the density reactivity coefficient (DRC) for a typical fuel[63] is shown in Figure 2.2 as a function of coolant density relative to the liquid phase density. Figure 2.2a represents an uncontrolled cell (four fuel assemblies) and Figure 2.2b represents a controlled cell (four fuel assemblies plus a control rod in between). In Figure 2.2a it can be seen that for the uncontrolled case the DRC increases as moderator density increases (i.e., as void fraction increases). Since the gain of the reactivity feedback is proportional to the value of the DRC, in general a high void fraction in a core with a low degree of control (i.e., operation at high power-to-flow ratios) will result in a less stable condition. Figure 2.2b shows, however, that the behavior of a controlled cell is the opposite, but since the controlled regions of the reactor have lower power, they have a smaller weight in computing the DRC (see Section 2.2.2), and the overall effect of an increase in voids is destabilizing.
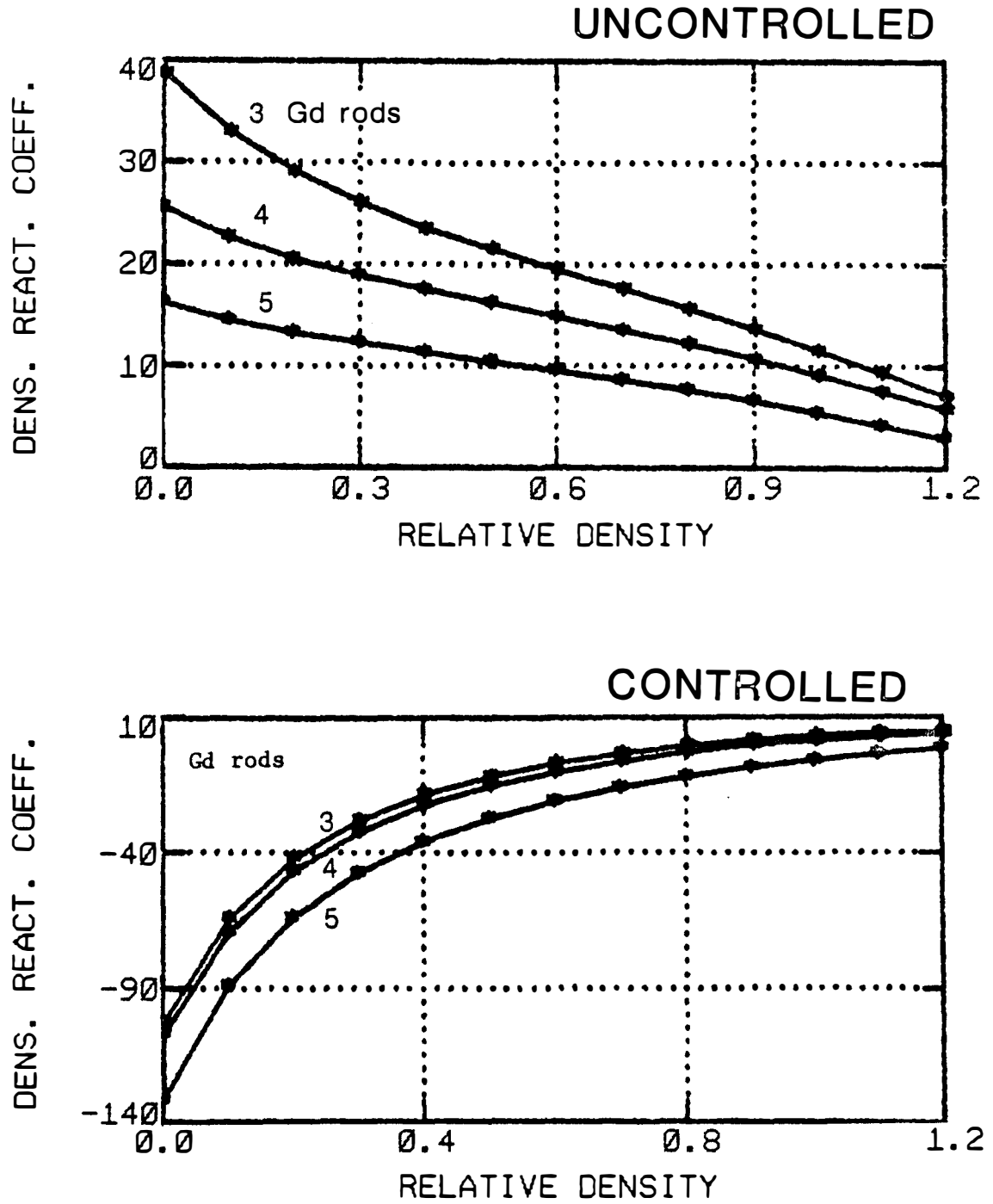
Figure 2.2 Density reactivity coefficient as a function of coolant
density and number of gadolinium rods.

Axial power shape. The power shape affects BWR stability in two ways: first, by a direct effect on the void fraction distribution, and second, through its square-weighting contribution to the reactivity feedback algorithm (see Section 2.2.2). A bottom-peaked power shape preferentially generates steam voids in the lower (inlet) part of the core. This effectively adds voids to all the core, thus maximizing the average core void fraction for a given power-to-flow ratio; it also increases the effective void residence time in the core which increases the reactivity feedback lag. Both effects contribute to making the reactor more unstable when the axial power shape is tilted towards the bottom-peaked configuration.

Steam void velocity. The higher the void velocity, the smaller the effective void residence time in the core; this variable, therefore, has two effects: (a) decreasing the phase lag and (b) increasing the rate at which moderator density perturbations are removed from the core, thus reducing the feedback gain. Both effects tend to make the reactor more stable. Therefore, for the same power-to-flow ratio, the condition with higher flow will be more stable.

Fuel gap conductance. A change in gap conductance alters the fuel-to-coolant heat flow transfer function. An increase in conductance (for instance, smaller gap) will increase the gain, thus making the reactor more unstable; at the same time, however, it will reduce the phase, a fact that tends to make the reactor more stable. Although these effects are opposite, the overall stability impact of the increase in gain is larger than that of the phase reduction.

Recirculation loop parameters. The pressure-to-core-inlet-flow transfer function in the recirculation loop has an important effect on stability.[48] It couples the inlet flow with the core hydraulics, thus closing the loop between the upper and lower plena. The reactor becomes less stable as a result of the increase in its gain or its time constant. In general, a decrease in recirculation pump speed increases both the gain and the time constant;[63] thus, reduced flow conditions are more unstable.

Control rod pattern. For a given axial power shape and void fraction distribution, the control rod pattern affects the reactor stability only through the DRC. An increase in the degree of control makes the reactor more stable.

Fuel isotopic composition. In Figure 2.2 the curves of DRC versus moderator density are shown as a function of gadolinium concentration. It can be observed that the DRC is smaller for higher Gd concentrations. Therefore, the higher the Gd contents (for instance, at the beginning of cycle) the more stable the reactor.

Radial power and flow distribution. In a BWR each channel contributes to the global core reactivity according to the integral over its length of the local DRC, weighted by the local power squared. Since the local DRC is determined by the void fraction distribution along the channel, changes in the core's radial power and flow distributions will affect the relative contribution of the particular channel to the overall reactivity feedback, thus affecting the stability of the core.

Inlet subcooling. This parameter affects the position of the boiling boundary thus changing the effective void residence time in the core. It also affects the average channel void fraction. Increasing inlet subcooling (i.e., lower inlet temperature) will decrease the residence time and the average void fraction. Therefore, it will result in a more stable reactor.

Total core flow. This parameter affects mainly the average void fraction and the bubble velocity. For the same power level, an increase in flow will reduce the void fraction and increase the bubble velocity. Both effects make the reactor more stable.

Core thermal power. An increase in power while keeping the flow constant (i.e., along a constant recirculation pump speed line) has the effect of increasing the void fraction and, thus, it has a destabilizing effect. However, if the power is increased through an increase in recirculation pump speed, the flow is also increased; in this case, the void fraction is kept essentially constant (in order to maintain the reactor critical) but the bubble velocity is increased. Therefore, a power increase along the flow control line results in a more stable reactor configuration.

2.4 Nonlinearities in BWRs

The reactor equations presented in Section 2.2 are not linear. However, any nonlinear system will behave in a linear manner if the perturbations around equilibrium are small enough. This is the case in BWRs. It has been shown experimentally[53-56] that this type of

reactors behave linearly during normal operation; thus, linear modeling is appropriate for these conditions. However, as explained in Section 2.3, BWRs are susceptible to instabilities. When the equilibrium point becomes unstable the oscillations become undamped and they grow large enough so that nonlinearities become important.

This effect has been observed in recent experiments.[57-58] In these experiments the reactor power was increased slowly while keeping the flow rate essentially constant. When the critical power level was reached, the reactor became unstable. At this moment, if the reactor were linear, the oscillations should have continuously diverged exponentially. However, this was not the case since the oscillation amplitude grew originally but it stayed bounded due to the appearance of a limit cycle which is a typical nonlinear effect. The observed limit cycles were of small amplitude, with about a ±15% oscillation in power. In this section we will describe the nonlinearities in the reactor equations which could account for the observed effects. In chapters 7 and 8, we will study these nonlinearities more carefully and their effects on the reactor's dynamic response.

The first nonlinearity appears in the neutron field equations. In the point kinetics representation, the term "reactivity-times-neutron-density" (ρn) is a nonlinearity because, due to the inherent reactivity feedback, ρ depends on n. The physical meaning of this term is that reactivity perturbations are weighted by the neutron density. This term forces the neutron density to be positive at all times. No matter how large a negative reactivity perturbation is

imposed in the reactor, the actual perturbation (as seen by the equations) will tend to zero as the neutron density is reduced.

The way in which the reactivity feedback is computed introduces another nonlinearity because the cross-sections and the associated reactivity coefficients are complex nonlinear functions of temperature and moderator density.

The fuel equation (2-64) is linear except for the temperature dependence of the heat conductance, density, and fuel heat capacity.

Assuming the operating pressure constant (see Section 2.2.4), the specific densities and enthalpies are constant and, thus, the mass conservation for the channel is linear. The energy and momentum conservation equations, however, are nonlinear. The product of G, the mass flux, times x, the steam quality, appears in the convection term of the energy equation. Furthermore, x is a complex nonlinear function of the void fraction, α. The non-linearities in the momentum equation are in the kinetic energy (G^2x^2/α) and friction (G^2) terms, which include the nonlinear dependence of the two-phase friction coefficient.

The recirculation loop is modeled using the integrated momentum equation and, thus, it contains the same nonlinerities described above.

In summary, we have seen that the true reactor equations have a multitude of nonlinearities in them. Although experiments have shown that none of them affects significantly the reactor normal operation, it has been observed that they become important when the linear stability limit is reached.

CHAPTER 3

LINEAR SYSTEMS DYNAMICS: THE CONCEPT OF STABILITY

The relationship between two variables in a system can be represented mathematically by a differential equation. If this equation is linear, or if it can be linearized for small perturbations, the equation can be Laplace transformed and the transfer function between two variables can be obtained as the ratio of two poynomials in s, the Laplace variable. The roots of the denominator polynomial are called the poles of the transfer function and the denominator are the zeros. Once the transfer function, G(s), of a linear system is known, the output of this system, R(s), for any input, I(s), is given in the Laplace domain by the product of the input times the transfer function

$$R(s) = G(s) \ I(s) \quad . \tag{3-1}$$

The output, r(t), in the time domain can be obtained using the well-known convolution theorem

$$r(t) = \int_0^t i(t) \ h(t-\tau) \ d\tau \quad , \tag{3-2}$$

where h(t) is the inverse Laplace transform of the transfer function. It can be shown that h(t) is also the response of a system to an impulse (or Dirac delta function) and is, therefore, usually called the impulse response of the system. Equation (3-2) clearly shows that the characteristics of the impulse response determine the response of the system to an input and consequently it determines the stability of

the system. If the impulse response decays to zero, then the output
follows the input in the sense that r(t) depends only on the latest
values of i(t); however, if h(t) increases with time, the old values
of i(t) are multiplied by a function which increases in magnitude. In
this case, the response becomes divergent and the system is said to be
unstable.

The impulse response can be calculated as a function of the
poles of the transfer function, $\{p_i, i = 1,2,\ldots,N\}$, as

$$h(t) = \sum_{i=1}^{N} R_i e^{p_i t} \quad , \tag{3-3}$$

where R_i is the residue of pole p_i. If p_i is a single pole, the
residue is given by the expression

$$R_i = \lim_{s \to p_i} (s-p_i)\, G(s) \quad . \tag{3-4}$$

Equation (3-3) shows that if at least one of the poles has a
positive real part, the impulse response will grow exponentially and
the system will be unstable. If all the real parts of the poles are
negative, the system will be stable and the impulse response will
asymptotically behave like the component of the pole with the smallest
real part, because all other components decay faster. Hence, the real
part of the most unstable pole (the one with smallest real part)
determines the relative stability of the system.

Usually we are interested in measuring the stability of
systems with oscillatory characteristics when the most unstable pole
is complex. It is in this context that the concept of decay ratio

arises. It can be shown that if a system has only a pair of complex conjugate poles, the impulse response is

$$h(t) = e^{\sigma t} \cos(\omega_d t + \psi) \quad , \tag{3-5}$$

where σ is the real part and ω_d is the imaginary part of the pole. The decay ratio is defined as the ratio between the second and first peaks in the impulse response. This ratio is constant for any two consecutive peaks and equal to

$$DR = e^{2\pi\sigma/\omega_d} \quad . \tag{3-6}$$

Therefore, we see that for a second order system, the decay ratio is directly related to the position of the poles and it is a good measure of the system stability.

For higher order systems, the impulse response is not formed by just one term but by the contributions from all the poles; therefore, the decay ratio between consecutive peaks is not constant. However, if a series is formed with the value of the decay ratio for every two consecutive peaks, it can be shown that it converges to a value equal to the decay ratio of a second order system with only a pair of complex poles at the same position as the least stable pair of poles in the original system. We call this value the asymptotic decay ratio, whereas the ratio between the first two peaks is the apparent decay ratio.

The asymptotic decay ratio is related to the position of the least stable pole as shown in Equation (3-6) and is, therefore, a good measure of the stability of the system. On the other hand, in

general the apparent decay ratio is not related to the stability of
the system. This last point is best clarified with an example:
Figure 3.1a shows the impulse response of a second order system. The
number on top of each peak is the decay ratio between that peak and
the previous one. It can be seen that all the decay ratios are equal
and that in the second order case the apparent and the asymptotic
decay ratios coincide. Figure 3.1b shows the impulse response of a
fourth order system whose least stable pair of poles is the same as
the system of Figure 3.1a. It is seen in this figure that the decay
ratios are not the same for all peaks; however, they converge to an
asymptotic decay ratio of 0.73, the same as in figure 3.1a. The
apparent decay ratio is 0.41, obviously a nonconservative estimate of
the stability of the system. The nonconservative nature of the
apparent decay ratio can be seen more dramatically in Figure 3.1c:
here, the same pair of poles as in Figure 3.1a are retained, but we
add a pair of unstable poles (positive real part) with a pair of zeros
very close, but not completely canceling them. The effect of these
zeros is to make the residue in Equation (3-3) for the unstable poles
very small; then, for short times, the dominant factor in the impulse
response is the stable poles, so the apparent decay ratio is less than
1.0 (equal to 0.55); however, as time increases the exponential nature
of the unstable poles dominates and makes the impulse response diverge
with an asymptotic decay ratio greater than one. Consequently, we can
conclude that the apparent decay ratio of the impulse response, even
though it is somehow related to the stability of the system, is not a
conservative estimate and is, thus, not a good measure of stability.
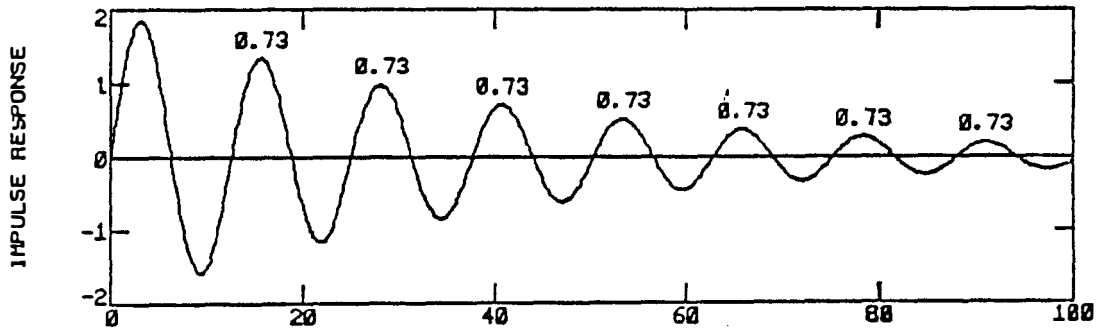
(a) Second order system
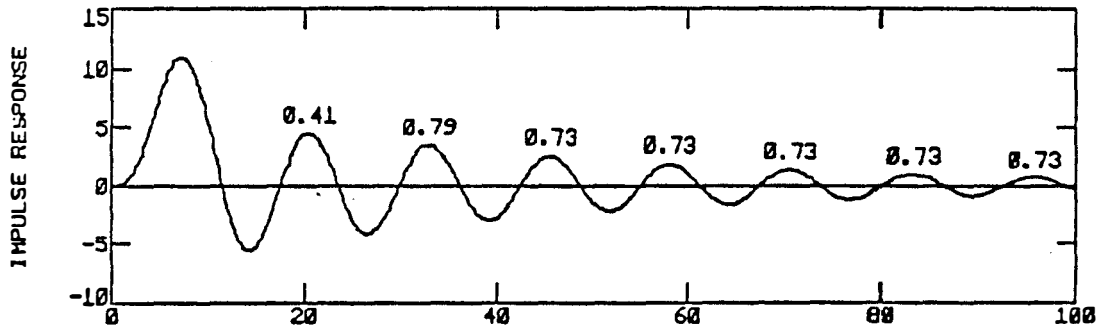
(b) Fourth order system
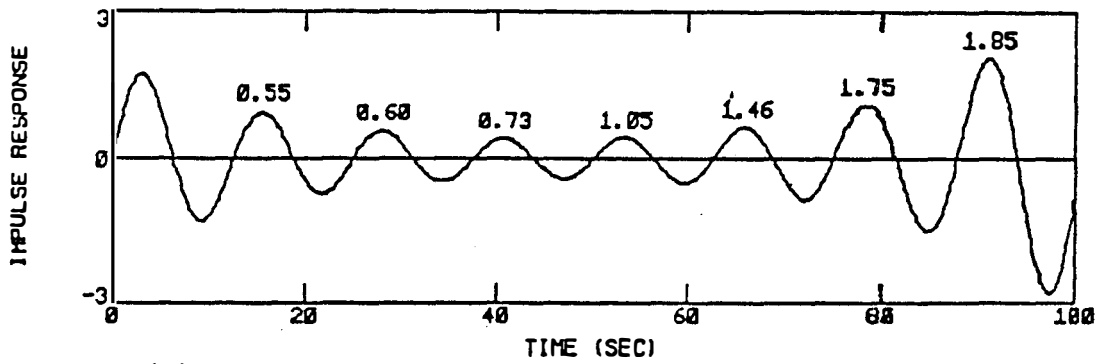
(c) Unstable fourth order system with two zeros

Figure 3.1 Typical impulse responses.

The impulse response is not a measurable quantity in a system. Once a system is excited by an input, the only measurable quantity is its response, $r(t)$, which contains information about both, the input and the system transfer function. Furthermore, in systems like BWRs, the input is unknown and stochastic; therefore, only average quantities like power spectral densities (PSDs) or correlation functions can be used to describe the output of such systems. For this reason, we must rely on these functions to estimate the output stability.

The power spectral density of the output, $PSD_R(\omega)$, can be calculated in terms of the system transfer function, $G(\omega)$, and the input noise spectrum, $PSD_I(\omega)$[109]

$$PSD_R(\omega) = G(\omega) \; G^*(\omega) \times PSD_I(\omega) \quad . \tag{3-7}$$

Therefore, the poles of the output PSD can be attributed to either the input PSD or the system transfer function.

The autocorrelation is defined as the inverse Fourier transform of the PSD and thus, it can be calculated as

$$C(t) = \sum_{i=1}^{M} R_i \; e^{p_i t} + \sum_{i=M+1}^{N} R_i \; e^{p_i t} \quad , \tag{3-8}$$

where R_i are the residues of the output spectrum and p_i its poles. The first M poles are assumed to come from the system transfer function and the rest from the input spectrum.

For oscillatory systems, we can define apparent and asymptotic decay ratios in the autocorrelation function in a manner similar to the impulse response. For the case in which the transfer function is less stable than the input (i.e., the least stable pole of PSD_R is

associated to a pole of $G(\omega)$), the asymptotic decay ratio of the auto-correlation function must coincide with that of the impulse response (see Equations (3-3) and (3-8)); however, the apparent decay ratios will in general be different. Figures 3.2a, b, and c show the auto-correlation function of the output of the systems in figure 3.1 when driven by white noise. These figures show that the apparent decay ratio of the autocorrelation is not a direct estimate of the system's stability and that it could be nonconservative; there is, however, an improvement over the apparent decay ratio of the impulse response in that if the system is unstable, the apparent decay ratio of the auto-correlation is greater than 1.0 and, therefore, provides a good measure of the approach to instability. Note that the autocorrelation function is an a posteriori measurement, and, therefore, its apparent decay ratio gives a measure of how much the output is actually oscillating.

In summary, we have seen that the stability of a system is defined by the position of the least stable pole of its transfer function. For oscillatory-type systems, the decay ratio is a measure of the system stability. Three types of decay ratios can be defined: (a) Asymptotic decay ratio, which is directly related to the absolute stability of the system; it is independent of the function used for its definition. (b) Apparent decay ratio of the autocorrelation function, which is related to the amount of oscillations during opera-tion; it coincides with the asymptotic decay ratio as it approaches the value of 1.0. (c) Apparent decay ratio of the impulse response,
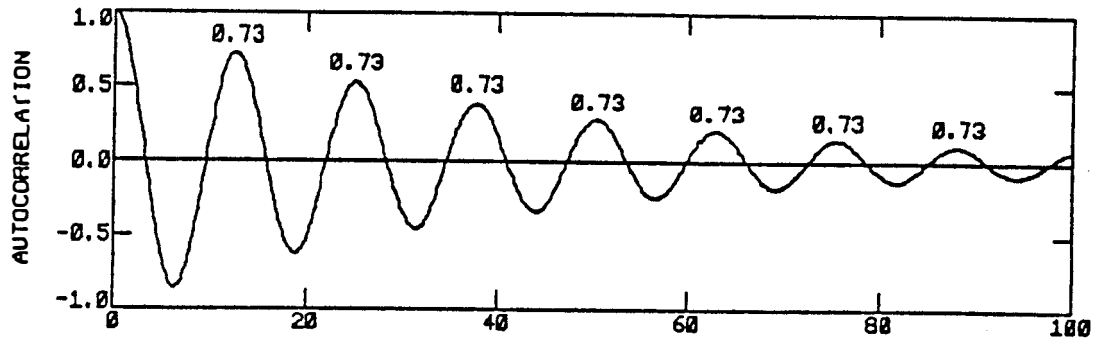
(a) Second order system
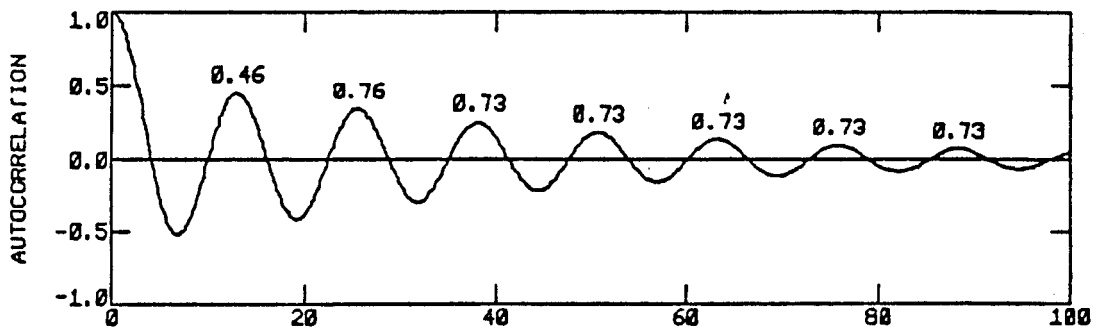
(b) Fourth order system
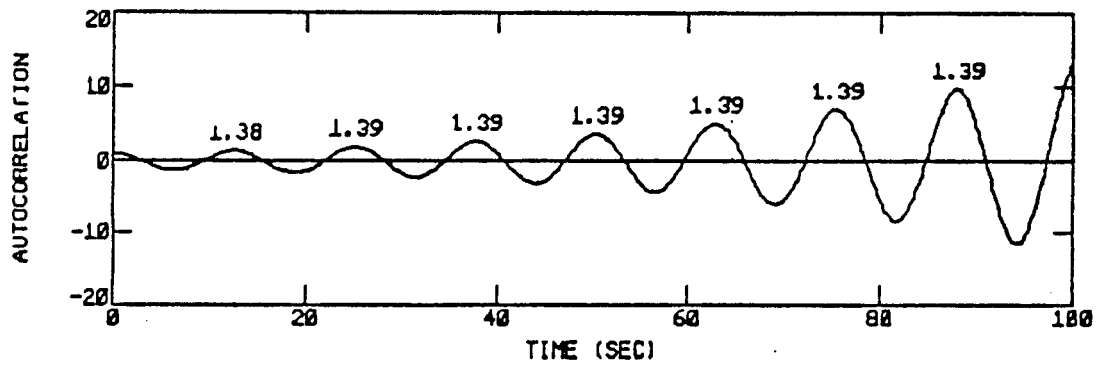
(c) Unstable fourth order system with two zeros

Figure 3.2 Typical autocorrelation functions.

which is only related to short-time behavior of the system; it does not measure the absolute stability of the sytem and, thus, should not be used for stability measurements.

.

.

CHAPTER 4

A PHYSICAL MODEL OF BWR LINEAR DYNAMICS

The goal in this chapter is to develop a linear model of the
dynamic behavior of BWRs through a detailed study of the various
physical processes involved.

In the past, models have been obtained by integrating the
reactor equations over a finite number of nodes. In this way a set
of coupled ordinary differential equations are generated for node-
averaged variables. The coefficients of these models are node-
averaged parameters which can be computed from first principles or
empirical correlations. This approach was taken by Otaduy[48] in deve-
loping the detailed BWR model implemented in the code LAPUR. It was
found[63] that the code yielded excellent results when compared with
reactor experiments, but a large number of nodes were needed (a mini-
mum of 50 axial and 3 radial nodes). A model with this number of
nodes is adequate to calculate numerical results; however, it is very
hard to extract a physical understanding about the dynamic process
from these numerical solutions.

The approach taken in this work is to minimize the number of
nodes involved in the modeling so that the results provide physical
information without sacrificing accuracy. In order for the low-order
model to represent accurately the dynamics of the system, nodal
synthesis was used. A series of "supernodes" were synthesized from
the results of a fine-mesh nodal calculation. Each of these

supernodes describes the physical process taking place in each of the system regions, such as fuel heat transfer or channel thermal-hydraulics. Associated with a particular supernode there is a set of average parameters such that the overall system dynamics is properly represented in the sense of matching the results of the fine-mesh nodal calculation.

The nodal synthesis is performed in two steps:

(a) Determination of the minimum number of poles and zeros needed for an accurate representation of the system transfer function for each set of operating conditions.

(b) Identification of the physical processes associated with the poles and zeros obtained as a result of the previous step.

From step (a) one obtains the equivalent linear set of dif-ferential equations which exhibits the pole-zero configuration describing the system transfer function, while step (b) associates the corresponding physical processes to each of the differential equations in the set. To implement step (a) one performs fits of polynomial ratios to the fine-mesh results with various model orders until model order convergence is reached. Order convergence is characterized by the appearance of spurious poles which cancel with zeros.

The final result of this fit is the smallest set of poles and zeros which give a reasonably accurate description of the system. The implementation of step (b) is carried out in two phases. First, para-meter-sensitivity studies are performed with the detailed model to

determine possible correlations to motions of transfer function features. This phase is followed by physical modeling of the process to justify that the effective nodal parameters are within reasonable physical ranges.

The outlined procedure was applied to the conditions of the stability test labeled 3PT3[56] which was performed in the Peach Bottom reactor. The results are shown in sections 4.2 through 4.5.

4.1 Brief Description of a Detailed BWR Model: the Code LAPUR

LAPUR[48] is a computer code developed at the Oak Ridge National Laboratory for the calculation of BWR core stability parameters. It uses a multinodal description of the neutron dynamics together with a distributed parameter model of the core thermal hydrodynamics to produce a space-dependent representation of the dynamics of a BWR in the frequency domain for small perturbations around a steady state condition. The LAPUR program consists of two autonomous modules, LAPURX and LAPURW, which are linked by means of an intermediate storage routine. The first module, LAPURX, solves the governing equations for the coolant and the fuel steady state. Maps of the core steady state are generated and stored in data files for subsequent utilization by LAPURW. The second module, LAPURW, solves the dynamic equations for the coolant, fuel, and neutron field in the frequency domain. A set of open-loop transfer functions are generated and the stability indices are estimated from the closed-loop reactivity-to-power transfer function.

The fuel equations in LAPUR assume no axial heat flow in the fuel rods, use a radial mesh within the fuel pellet to account for the radial dependence of the heat source as well as the temperature dependence of the UO_2 fuel's heat conductivity, and include the effect of the nonconductive transfer of heat to the coolant by γ-ray absorption and neutron moderation processes. Coolant dynamics include three flow regions in a flow channel: a nonboiling region, a subcooled-boiling region, and a bulk-boiling region. The conservation equations and two-phase fluid mechanical equations, in conjuction with the fuel equations, yield a set of transfer functions relating perturbations of the nodal coolant density and pressure drop to nodal perturbations of coolant temperature, flow rate, and power generation. Upon integration of these functions along the length of the channel and along the partial length of the channel pertaining to each nuclear subcore, a fuel-and-coolant matrix equation of transfer function results.

To obtain the reactivity feedback transfer function matrices, the fuel temperature and coolant density nodal transfer functions are weighted by both the local power and the local density reactivity coefficients and integrated over the volume of each of the neutronic subcores.

The overall space-dependent transfer function matrix representation of the BWR core dynamics with feedback is obtained by consideration of the hydraulic coupling through the inlet and outlet plena and the recirculation loop, in conjunction with the neutronic matrix equation. To calculate the stability indices (decay ratio and

natural frequency) the position of the most unstable pole of the core's closed-loop transfer function is determined according to Smith's frequency gradient method.[101]

The accuracy of the LAPUR model to predict the stability of commercial BWR cores has been demonstrated.[63] However, a large number of nodes (50 axial, 3 radial) are needed. The LAPUR code was used as reference to develop the reduced order model presented in this work.

4.2 The Pole-Zero Configuration of the Reactor Transfer Function

Figure 4.1 shows the magnitude and phase of the reactivity-to-power transfer function calculated by LAPUR for the conditions of test 3PT3[56] in the Peach Bottom reactor. The main dynamic features which can be observed at first glance are: a low frequency zero (0.03 Hz), a break frequency at about 21 Hz, and the resonance peak at about 0.3 Hz which determines the transfer function stability. These observations would suggest the use of a 1-zero/3-pole model to fit this transfer function, but this would yield an asymptotic value of the phase of -180 degrees at high frequencies. Since the phase converges only to -90 degrees at 1000 Hz, an extra zero is needed. Figure 4.2 shows a comparison between the LAPUR transfer function and the fitted 2-zeros/3-poles model; from this comparison we conclude that this model order is inadequate. When the next model order (3-zeros/4-poles) was used, the results shown in Figure 4.3 were obtained. This figure shows that this model order is more appropriate.
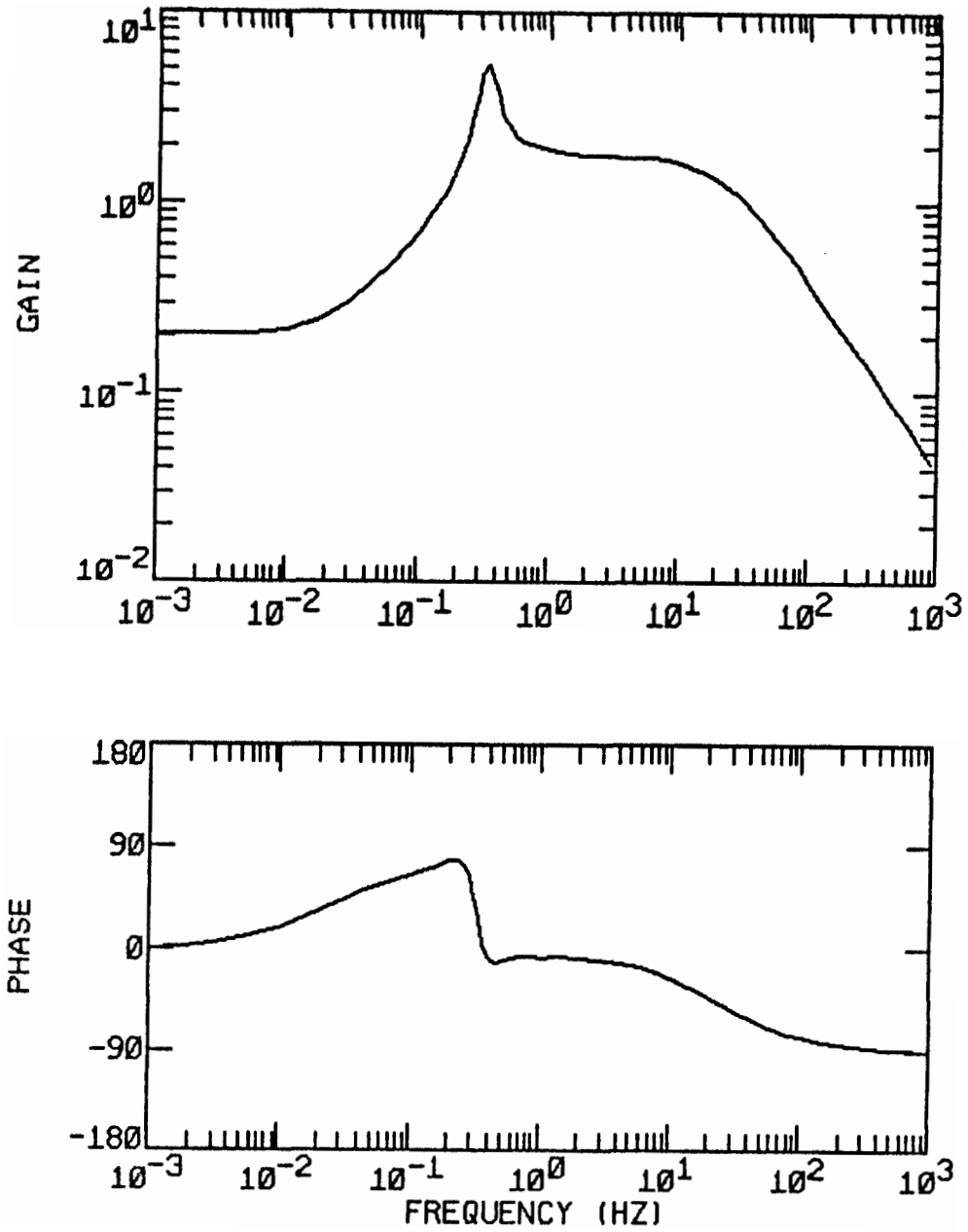
Figure 4.1 Closed loop reactivity-to-power transfer function for test 3PT3, Peach Bottom.
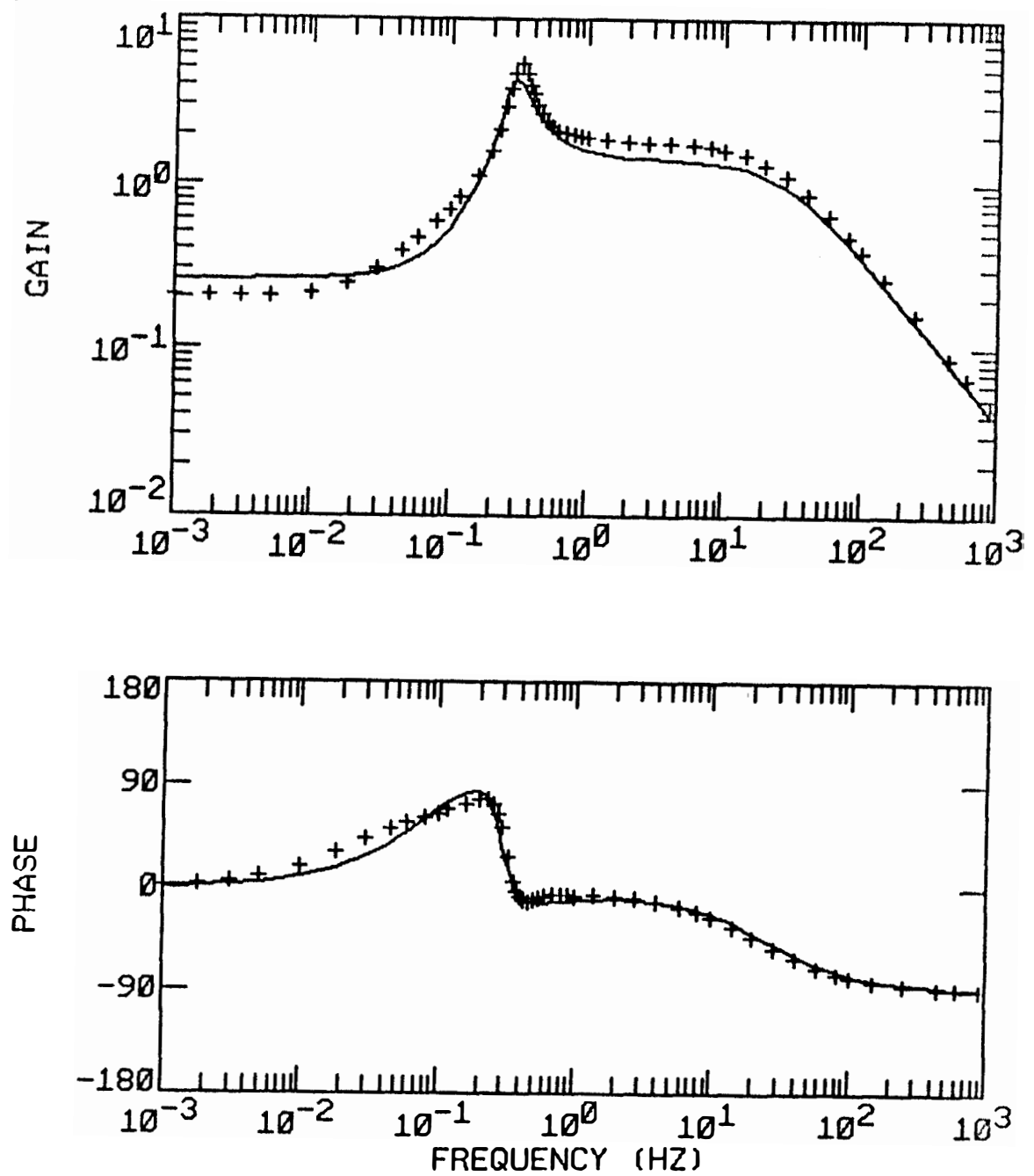
Figure 4.2 Comparison between closed loop transfer function and
2-zero/3-pole fit. Test 3PT3, Peach Bottom.
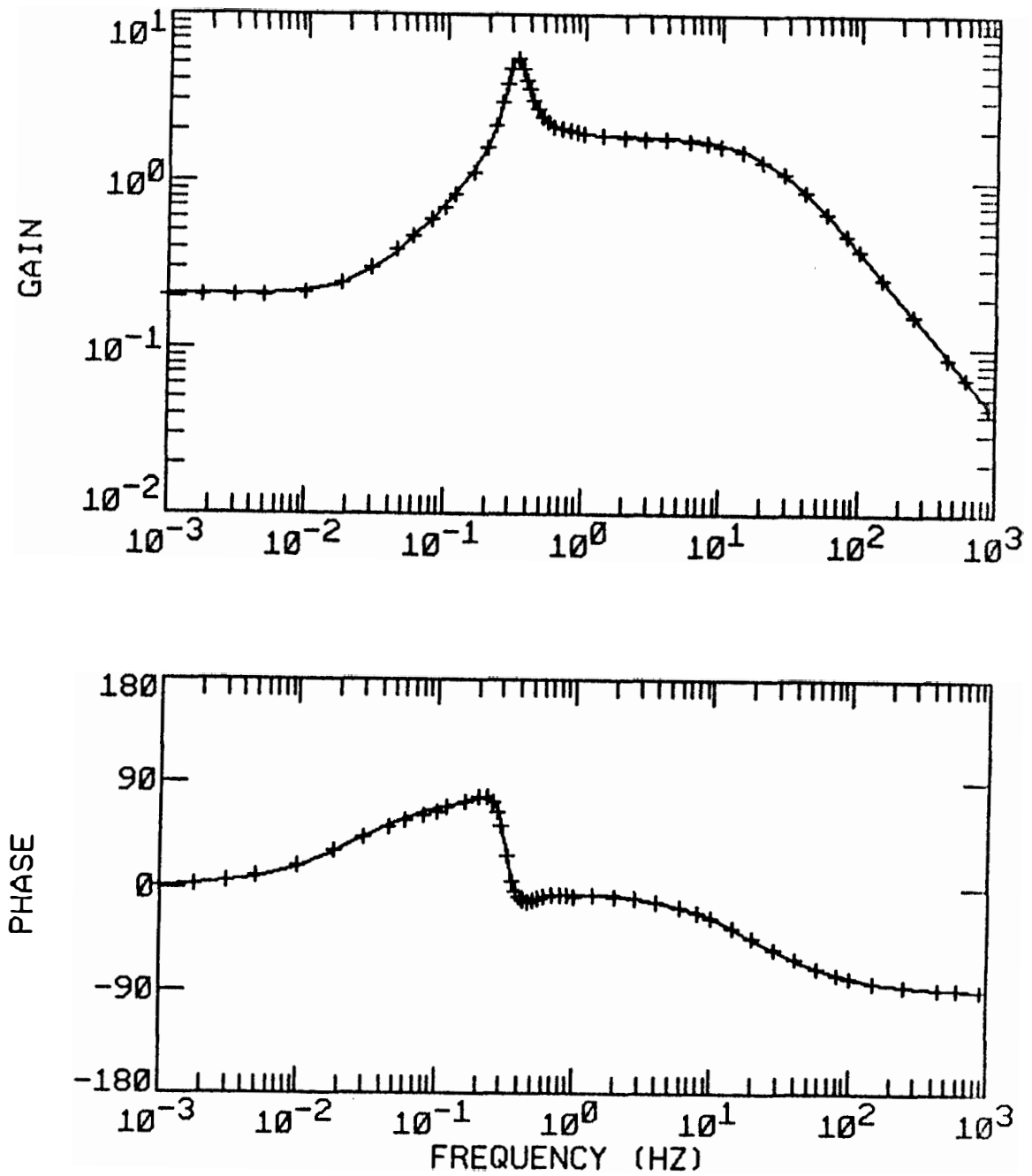
Figure 4.3   Comparison between closed loop transfer function and
3-zero/4-pole fit.   Test 3PT3, Peach Bottom.

Table 4.1 contains the numerical value of the poles and zeros corresponding to this fit. We observe that there is a low-frequency real zero, a complex pair of zeros, two real poles, and a complex pair of poles. The real part of this last pair of poles determines the reactor stability.

Table 4.1

RESULTS OF THE 3-ZERO/4-POLE MODEL
ORDER FIT TO THE BASE CASE

Zeros (Hz)	Poles (Hz)
-0.03	-0.25
-0.18±0.27i	-0.045±0.32i
	-21.7

To further determine if the pole-zero configuration obtained is appropriate, we constructed the root locus[99] for this transfer function. In conventional linear system dynamics, the open loop poles and zeros are known and there are standard techniques[99] to draw the root locus as the feedback gain, K, is increased. However, in the present work, the poles and zeros of the open loop are not readily available and, furthermore, there is a very large number of them. For this reason, the root locus was constructed by fitting the closed-loop transfer function obtained by LAPUR for several values of K. Following this procedure, only the significant poles and zeros are obtained, because most of the poles and zeros of the open loop transfer function will cancel as the feedback gain is increased.[99]

Figure 4.4 shows how the LAPUR calculated closed-loop transfer function for test case 3PT3[56] changes as the nominal feedback gain is multiplied by a factor K. In this figure a value of K = 0 gives the open loop forward transfer function and a value of K = 1 corresponds to exactly the transfer function calculated by LAPUR. We observe that as K is increased, the frequency of the characteristic peak increases, and the peak itself becomes sharper. A critical value of approximately K = 2.25 defines the limit of linear stability. For higher values of K, the complex poles become unstable, but the magnitude of the peak decreases as they move away from the imaginary axis. Note the appearance of the low frequency zero as K is made larger than 0.25.

Nonlinear fits were performed on the closed-loop transfer function for various values of K. The motion of the poles is shown in the root locus diagram of Figure 4.5. Note that only the low frequency part is represented in this figure, and, therefore, the 21 Hz peak is not present. We observe that

(a) The pair of complex poles which determine the reactor stability originate from the complex pair of zeros. These zeros correspond, therefore, to a pair of complex poles in the feedback part of the open loop transfer function (see Section 4.4.2 for more details).

(b) The low frequency zero is not present in the closed-loop transfer function when K = 0 (see Figure 4.4). Therefore, it must be caused by the presence of an open-loop pole in the
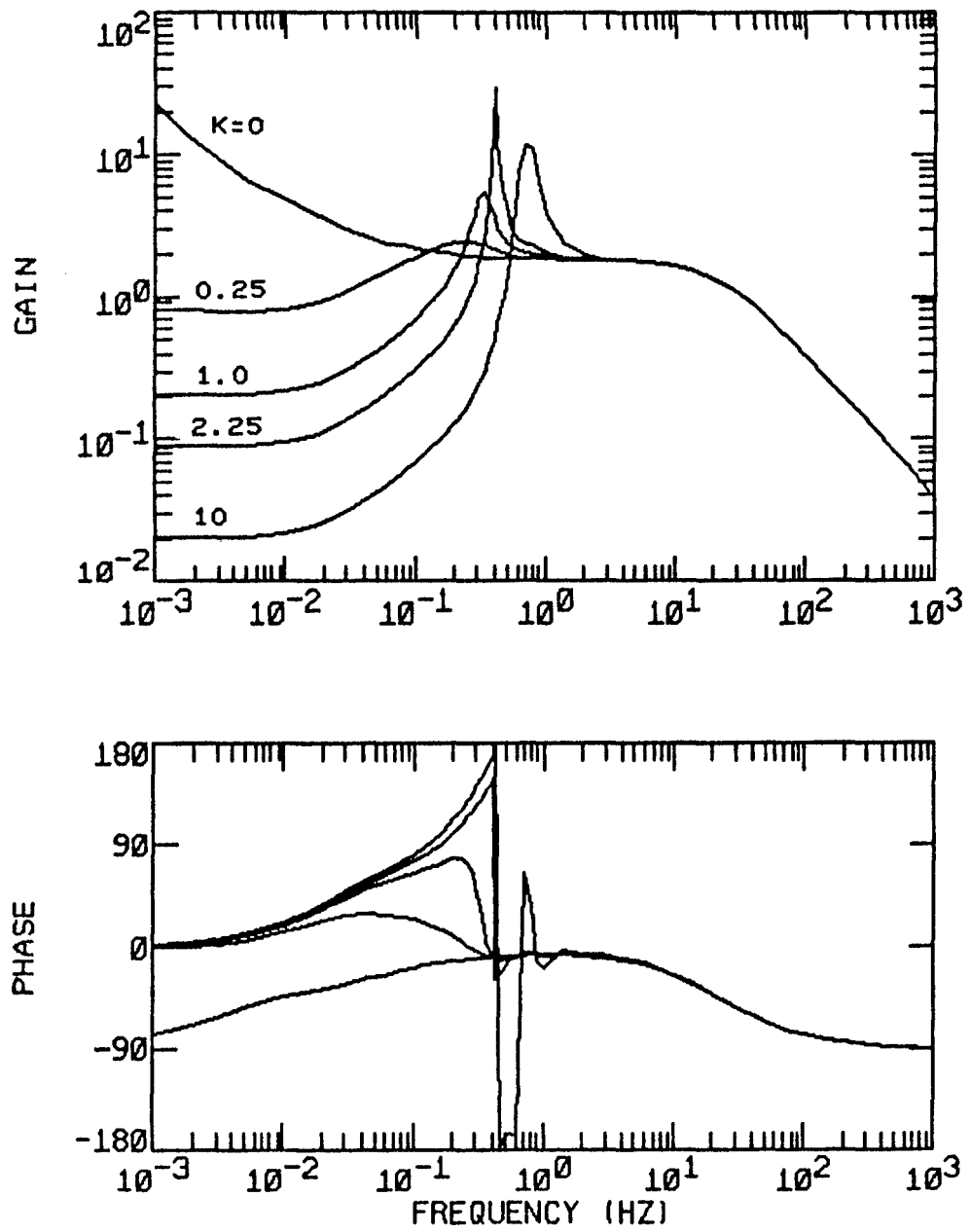
Figure 4.4 Closed loop transfer function as the feedback gain (K) is changed. (Test 3PT3, Peach Bottom).

ORNL–DWG 84-12083

CLOSED LOOP ZEROS

CLOSED LOOP POLES
FOR K = 1.0
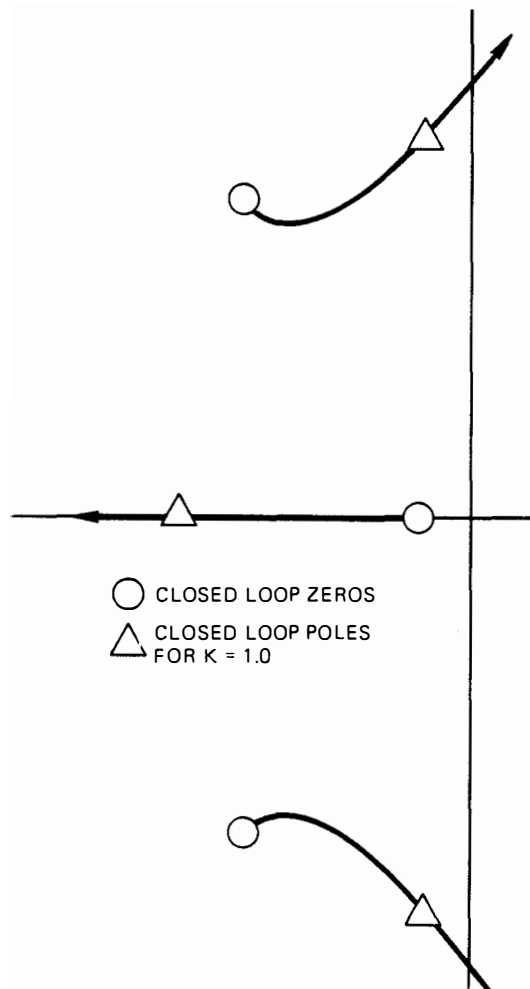
Figure 4.5   Root locus of reactivity-to-power transfer function
             obtained from the closed loop fit.
             (Test 3PT3, Peach Bottom).

feedback. The real pole at 0.27 Hz originates from this low
frequency zero.

Summarizing, we have determined that the closed-loop
reactivity-to-power transfer function can be appropriately represented
by an empirical model containing three zeros and four poles. If only
low frequencies (<10 Hz) are of interest, a 3-zero/3-pole model is
sufficient. The three zeros identified in the closed-loop transfer
function correspond to open loop feedback poles. Finally, the complex
pair of poles which determine the reactor stability originate from a
pair of open-loop complex poles.

4.3 Sensitivity of the Poles and Zeros
of the Reactor Transfer Function

Within the framework of linear dynamics theory, given a
closed-loop system formed by a forward transfer function, G, and a
feedback transfer function, H, the closed-loop transfer function, T,
is given by

$$T = \frac{G}{1 + GH} \quad . \tag{4-1}$$

The zeros that appear in T are, thus, the zeros of G and the poles
of H. The poles of T, however, take a more complex expresion and
are determined by the characteristic equation

$$1 + GH = 0 \quad . \tag{4-2}$$

Without loss of generality the BWR forward transfer func-
tion, G(s), can be represented by the point kinetics approximation
with a single group of delayed neutrons

$$G(s) = \frac{dn}{d\rho} = \frac{N_0}{\Lambda} \frac{s+\lambda}{s(s+\beta/\Lambda+\lambda)} \quad , \tag{4-3}$$

where λ is approximately 0.08 s^{-1} and β/Λ is of the order of 20 s^{-1}.

Therefore, the zeros that we have identified in the LAPUR closed-loop

transfer function of the reactor do not correspond to zeros of G.

Thus, they must be poles of H. At this point we cannot identify the

poles with the physical processes that they represent. Similarly,

since the poles of T are given by the complex Equation (4-2), we can

not identify them either. We shall concentrate for the moment on

identifying the three zeros of T by studying their sensitivity to

changes in reactor physical parameters.

Table 4.2 shows the sensitivity of the fitted poles and zeros

to changes in the fuel gap conductance. We observe that the complex

zeros remain practically unchanged, but the low-frequency zero changes

from 0.028 to 0.036 Hz, a 30% change. This fact suggests that this

zero is related to the fuel heat transfer time constant. Indeed, as

Table 4.2

SENSITIVITY TO FUEL GAP CONDUCTANCE OF THE POLES AND
ZEROS FITTED TO THE CLOSED LOOP TRANSFER FUNCTION

	Gap −20%	Base	Gap +20%	Gap +50%
Zeros (Hz)	−0.028 −0.18±0.27i	−0.030 −0.18±0.27i	−0.033 −0.18±0.28i	−0.036 −0.18±0.28i
Poles (Hz)	−0.051±0.31i −0.26 −21.0	−0.045±0.32i −0.25 −21.7	−0.044±0.33i −0.28 −20.7	−0.042±0.34i −0.30 −21.9

the gap conductance is increased, the fuel becomes more responsive, this fact would explain the increase in frequency (i.e. a decrease of the time constant). Figure 4.6 shows the power-to-heat-flux transfer function as calculated by LAPUR. The break (real pole) at about 0.03 Hz is apparent. This last figure along with the results of Table 4.2 imply that the low frequency zero should be associated with the fuel dynamics.

Tables 4.3 and 4.4 show the results of the fit when the core flow and power are changed respectively. We observe that the frequency of the complex zeros increases as the power or flow are increased. The real zero remains essentially constant. These facts suggest that the complex zeros should be related to the void sweep time in the core: as the power or flow increases, the steam bubbles travel faster through the core and reduce the sweep time, thus producing a higher frequency response. Figure 4.7 shows the heat-flux-to-density-reactivity transfer function as calculated by LAPUR. The double pole break at about 0.3 Hz is evident. This confirms the hypothesis that the pair of zeros in the closed-loop reactivity-to-power transfer function are due to the channel thermal hydraulics and the associated density reactivity feedback.

Summarizing, the sensitivity study indicates that an approximation to the reactor transfer function could be composed of: (a) Point kinetics to represent the neutron field dynamics. This will form the forward open-loop transfer function. (b) A single-node approximation for the fuel dynamics, which will yield the observed
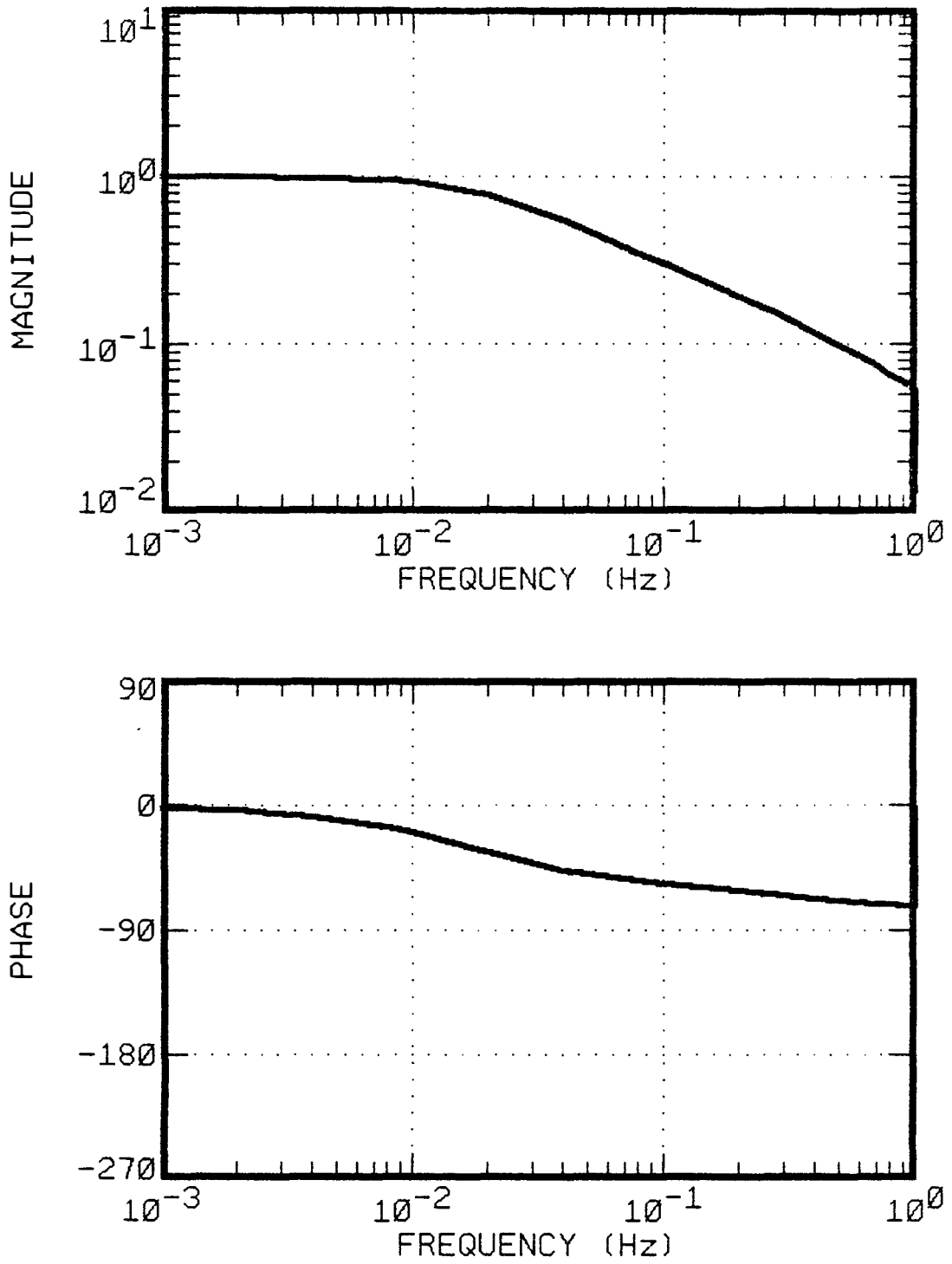
Figure 4.6   Typical power-to-heat-flux transfer function
calculated by LAPUR.

Table 4.3

SENSITIVITY TO TOTAL CORE FLOW OF THE POLES AND ZEROS
FITTED TO THE CLOSED LOOP TRANSFER FUNCTION

	Flow -10%	Base	Flow +10%
Zeros (Hz)	-0.030 -0.16±0.26i	-0.030 -0.18±0.27i	-0.030 -0.18±0.28i
Poles (Hz)	-0.037±0.31i -0.26 -20.5	-0.045±0.32i -0.25 -21.7	-0.058±0.33i -0.26 -21.0

Table 4.4

SENSITIVITY TO THERMAL POWER OF THE POLES AND ZEROS
FITTED TO THE CLOSED LOOP TRANSFER FUNCTION

	Power -10%	Base	Power +10%
Zeros (Hz)	-0.030 -0.18±0.25i	-0.030 -0.18±0.27i	-0.031 -0.18±0.29i
Poles (Hz)	-0.054±0.30i -0.25 -21.0	-0.045±0.32i -0.25 -21.7	-0.042±0.33i -0.28 -20.9
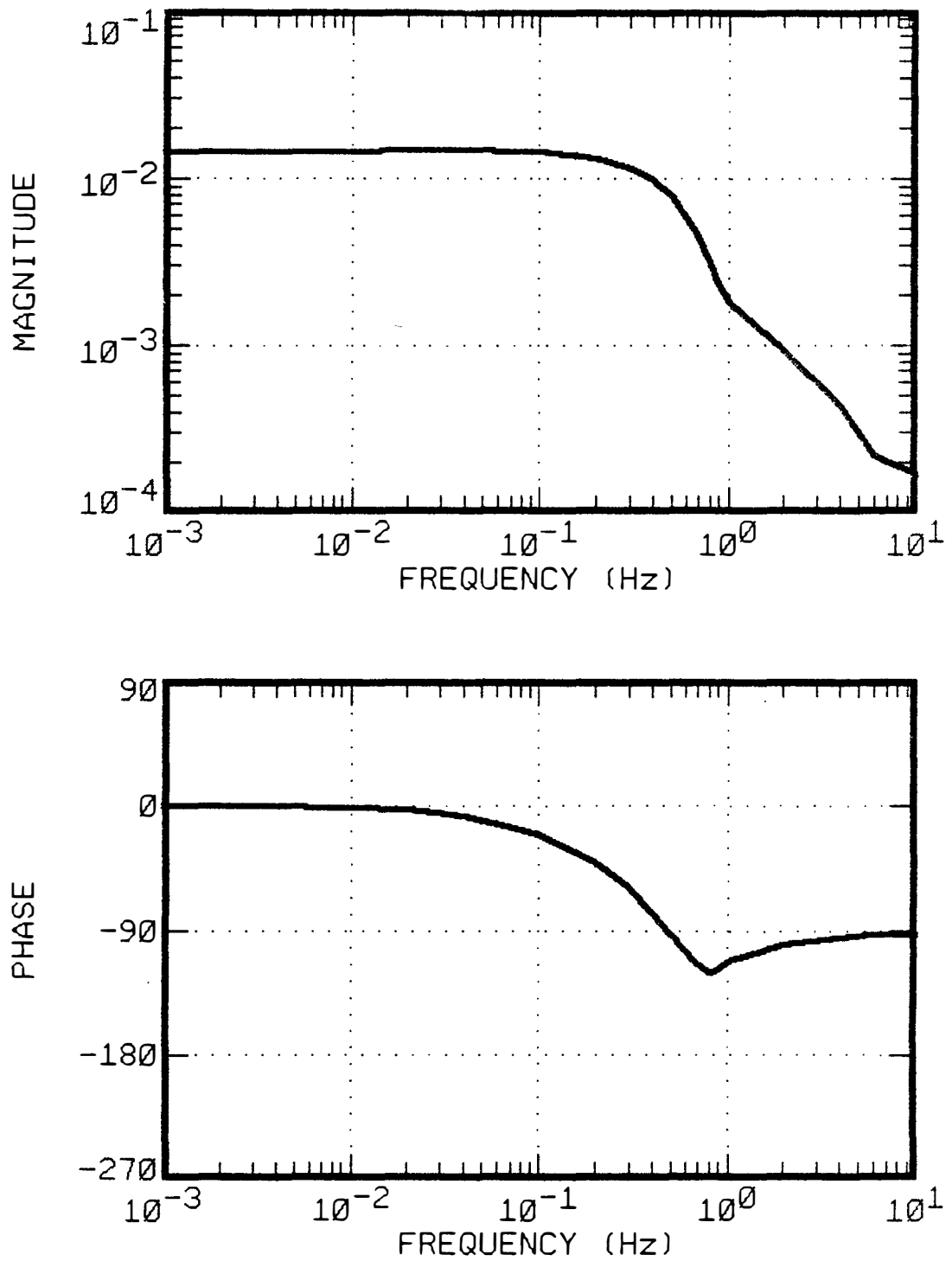
Figure 4.7   Typical heat-flux-to-density-reactivity transfer
            function calculated by LAPUR.

low-frequency zero in the closed-loop response. (c) A two-node

approximation for the channel thermal-hydraulics and reactivity feed-

back, which would contribute the complex pair of zeros.

Further proof that this model order is appropriate as well as

the identification of poles and zeros with physical components will be

given in Section 4.4.

4.4 The Linear Physical Model

In this section we study the linear equations representing the

reactor with the purpose of identifying the features observed in the

reactor transfer function.

4.4.1 The Fuel Transfer Function

We established in Sections 4.2 and 4.3 that only one pole was

needed to represent the fuel dynamics in the closed-loop reactivity-

to-power transfer function of a BWR. This can be accomplished from a

single-node expansion of the equation describing the temperature

distribution in the fuel. The effective coefficients of the resulting

lumped parameter model are selected to match the position of the known

poles (closed-loop zeros).

The equation governing the temperature distribution within

the fuel rod is

$$\rho c_p \frac{\partial T}{\partial t} = Q + \nabla k \nabla T \quad . \tag{4-4}$$

We can obtain a one-node lumped parameter model by integrating

Equation (4-4) over the fuel. We define an average fuel temperature

$$T_F = \frac{1}{4\pi r_F^2} \int_0^{r_F} 2\pi r \, T(r,t) \, dr \quad . \tag{4-5}$$

Applying the operator $\frac{1}{\pi r_F^2} \int_0^{r_F} [\ ] \, 2\pi r dr$ to Equation (4-4), the

following nodal equation is obtained

$$\langle \rho c_p \rangle \frac{\partial}{\partial t} T_F = \langle Q \rangle + \frac{2}{r_F} \left[k \frac{\partial T}{\partial r} \right]_{r=r_F} \quad , \tag{4-6}$$

where

$$\langle \rho c_p \rangle = \frac{1}{T_F \pi r_F^2} \int_0^{r_F} 2\pi r \rho c_p T \, dr \quad , \tag{4-7}$$

and

$$\langle Q \rangle = \frac{1}{\pi r_F^2} \int_0^{r_F} 2\pi r Q \, dr \quad . \tag{4-8}$$

The boundary condition is

$$\left[k \frac{\partial T}{\partial r} \right]_{r=r_F} = -U(T_F - T_c) \quad , \tag{4-9}$$

where U is the effective overall reactor heat transfer coefficient

(in J/Ksm^2) and T_c is the coolant saturation temperature.

Equation (4-6) becomes

$$\frac{d}{dt} T_F = \frac{\langle Q \rangle}{\langle \rho c_p \rangle} - \frac{2U}{r_F \langle \rho c_p \rangle} (T_F - T_c) \quad . \tag{4-10}$$

The heat transferred to the coolant can be estimated as

$$Q' = 2\pi r_F U(T_F - T_c) \quad . \tag{4-11}$$

Taking small perturbations and Laplace transforming equation (4-9),

we obtain the fuel transfer function

$$\frac{\delta T_F(s)}{\delta \langle Q \rangle} = \frac{1/\langle \rho c_p \rangle}{s + 2U/(r_F \langle \rho c_p \rangle)} \quad , \tag{4-12}$$

which as expected contains a single real pole.

To match the conditions of the test 3PT3 for which we made the closed loop fits, this pole must be located at -0.03 Hz. Therefore, the effective heat transfer coefficient, U, must be

$$U = \frac{1}{2} (2\pi\ 0.03\ r_F\ \langle\rho c_p\rangle) = 1481\ \text{J/Ksm}^2 \qquad (4\text{-}13)$$

This value is of the order of magnitude expected. A typical value of the fuel-to-cladding gap heat transfer coefficient is 4500 J/Ksm^2 (800 BTU/hrft^{2}F). The effective U obtained is smaller, because it takes into account the fuel pellet and cladding conductances as well as the film coefficient between cladding and coolant. Furthermore, the temperature in Equation (4-11) is the average temperature over the whole rod, which is larger than the surface temperature.

We can, therefore, safely conclude that the low-frequency (0.03 Hz) zero in the closed-loop reactor transfer function is due to the fuel dynamics.

4.4.2 The Channel Transfer Function

It was suggested in Sections 4.2 and 4.3 that the channel dynamics influence the closed-loop reactivity-to-power transfer function through a complex pair of zeros. In other words, the channel transfer function can be represented by a two-node expansion of the energy and continuity equations. Let's first study the channel thermal hydraulics. Later we will introduce the reactivity feedback. Neglecting acoustic phenomena,[29] the channel equations (continuity and energy) described in Section 2 become

$$\frac{\partial}{\partial t}\left[(1-\alpha)\rho_1 + \alpha\rho_g\right] + \frac{\partial G}{\partial z} = 0 \tag{4-14}$$

$$\frac{\partial}{\partial t}\left[(1-\alpha)\rho_1 h_1 + \alpha\rho_g h_g\right] + \frac{\partial}{\partial z}\left[(1-x)h_1 G + x h_g G\right] = Q' \quad . \tag{4-15}$$

We are interested in a linearized version of these equations. For this we need the steady state relations

$$\frac{\partial G_0}{\partial z} = 0 \tag{4-16}$$

$$\frac{\partial x_0}{\partial z} = \frac{Q_0}{(h_g - h_1)G_0} \quad . \tag{4-17}$$

Linearizing (4-14) and (4-15) and considering (4-16) and (4-17) we obtain

$$\{(\rho_g h_g - \rho_1 h_1) + (h_1(1-x_0) + h_g x_0)(\rho_1 - \rho_g)\}\frac{\partial\delta\alpha}{\partial t}$$

$$+ (h_g - h_1)G_0\frac{dx}{d\alpha}\frac{\partial\delta\alpha}{\partial z} = Q^0 \left(\frac{\delta Q}{Q_0} + \frac{\delta G}{G_0}\right) \quad , \tag{4-18}$$

where

$$G(z,t) = G(0,t) + (\rho_1 - \rho_g)\int_0^z \frac{\partial\alpha(z,t)}{\partial t}\,dz \quad . \tag{4-19}$$

Define

$$V_0 = \frac{G_0(h_g - h_1)\dfrac{dx}{d\alpha}}{(\rho_g h_g - \rho_1 h_1) + (h_1(1-x_0) + x_0 h_g)(\rho_1 - \rho_g)} \tag{4-20}$$

and

$$H_0(z) = (\rho_g h_g - \rho_1 h_1) + (h_1(1-x_0) + x_0 h_g)(\rho_1 - \rho_g) \quad . \tag{4-21}$$

Note that both V_0 and H_0 are functions of z but not time.

Equation (4-18) can then be rewritten as

$$\frac{\partial\delta\alpha}{\partial t} + V_0\frac{\partial\delta\alpha}{\partial z} = \frac{Q_0}{H_0}\left(\frac{\delta Q}{Q_0} + \frac{\delta G}{G_0}\right) \quad . \tag{4-22}$$

This last equation has the form of a convection equation, where V_0 is the propagation velocity. There are two possible perturbation sources in the right hand side: the volumetric heat flux transferred to the coolant, δQ, or the mass flux, δG, which could be altered through the inlet flow (see equation (4-19)). The z-dependence in this equation can be eliminated using an integrating factor. If we Laplace transform and multiply Equation (4-21) by e^{sz/V_0}, we obtain

$$\delta\alpha(z,s) = e^{sz/V_0(z)} \int_0^z e^{-sz'/V_0(z')} \frac{Q_0}{V_0(z')H_0(z')} \left(\frac{\delta Q}{Q_0} + \frac{\delta G}{G_0}\right) dz'. \quad (4-23)$$

Note that H_0 has units of enthalpy and it is not related to the variable H, the core height. If δG did not depend on α or if the dependence was weak compared to δQ, then this equation would be a closed-loop solution for the channel thermal hydraulics. Note, however, that δG depends on α not only directly as seen on Equation (4-18), but also indirectly through the momentum equation and the recirculation loop, which determine $G(0,t)$.

Let us assume for the sake of simplicity that $\delta G/G_0$ can be neglected with respect to the term $\delta Q/Q_0$. This would be the case when no recirculation loop is considered and the dynamics of the channel are driven by power perturbations. In this case,

$$\delta\alpha(z,s) = e^{sz/V_0(z)} \int_0^z e^{-sz/V_0(z')} \frac{\delta Q}{V_0(z')H_0(z')} \, dz'. \quad (4-24)$$

What this equation tells us is that the void fraction, α, at a particular time and axial location is produced by all the previous

power perturbations at lower channel heights, which are seen with a time delay introduced by the effective velocity, $V_0(z)$. Note that this velocity (see equation (4-20)) depends only on the steady state steam quality distribution and the factor $dx/d\alpha$, which is a function of x_0 and the slip ratio.

Within the context of our model, the heat flux perturbations are spacially correlated. This means that δQ can be separated into two components

$$\delta Q(z,s) = \Phi_0(z)\ \delta q(s) \tag{4-25}$$

but $\alpha(z,s)$ will not be separable due to the delay term inside the integral in Equation (4-24).

The transfer function that defines the channel dynamics in our model is the heat-flux-to-void-reactivity-feedback transfer function. It was shown in Section 2.2.2 that the void reactivity feedback is given by the expression

$$\delta\rho_\alpha(s) = \int_0^H \Phi_0^+(z)\ \frac{\delta\rho}{\delta\alpha}\ \delta\alpha\ \Phi_0(z)\ dz\ , \tag{4-26}$$

where $\delta\rho/\delta\alpha$ is the void reactivity coefficient at level z and the flux is normalized so that $\int\Phi_0^+ \Phi_0 = 1$. Since the flux in 1-D diffussion is self adjoint, and taking into account Equations (4-24), (4-25), and (4-26) the transfer function of interest becomes

$$\frac{\delta\rho_\alpha(s)}{\delta q(s)} = C \int_0^H \left[\Phi_0^2(z)\ \frac{\delta\rho}{\delta\alpha}\ e^{sz/V_0(z)} \right.$$

$$\left. \int_0^z e^{-s(z-z')/V_0(z')}\ \frac{\Phi_0}{V_0(z')H_0(z')}\ dz' \right] dz\ . \tag{4-27}$$

Let us make a further simplifying assumption in order to understand the physics behind this equation. To first order approximation, we can consider $\Phi_0(z)$ constant and that the steady state void fraction contents is very small, so $V_0(z)$, $H_0(z)$ and the density reactivity coefficient are constant over the length of the channel. Then the channel transfer function becomes

$$\frac{\delta\rho_\alpha(s)}{\delta q(s)} = C \int_0^H \left[\int^z e^{-s(z-z')/V_0} \, dz' \right] dz \quad . \tag{4-28}$$

With the previous assumptions, this equation can be integrated directly to give

$$\frac{\delta\rho_\alpha(s)}{\delta q(s)} = C \left[\frac{V_0 H}{s} - \frac{V_0^2(1-e^{-sH/V_0})}{s^2} \right] \quad . \tag{4-29}$$

The first term in this equation is a pure s^{-1} term. The numerator of the second term, however, oscillates between zero and one with a time constant $\tau = H/V_0$, the residence time of the bubbles in the channel. Since the second term has a s^{-2} dependency, the asymptotic behavior at high frequencies is that of a first order system. Let us try to find an approximation to Equation (4-29) at low frequencies. To this end we shall use the Pade approximant expression for the exponential function,[102] which is known to be an excellent approximation for small values of the independent variable. The Pade approximant to e^{-x} of order [2/2] is

$$P_{2,2} [e^{-x}] = \frac{1 - x/2 + x^2/12}{1 + x/2 + x^2/12} \quad . \tag{4-30}$$

Introducing it into Equation (4-29), we obtain an approximation to the channel transfer function in term of poles and zeros which is valid for low frequencies.

$$\frac{\delta\rho_\alpha(s)}{\delta q(s)} = C \frac{H^2}{\tau} \frac{s + 6/\tau}{s^2 + 6s/\tau + 12/\tau^2} \quad , \qquad (4\text{-}31)$$

where τ is the bubble transit time defined previously.

The two poles of this transfer function are always complex and located at

$$s = \frac{1}{\tau} (-3 \pm 1.41i) \quad . \qquad (4\text{-}32)$$

The zero is located at

$$s = -6/\tau \quad , \qquad (4\text{-}33)$$

where s is measured in radians per second.

Figure 4.8 shows a numerical calculation of the magnitude and phase of the channel transfer function using Equation (4-29) and assuming $V_0 = 2m/s$ and $H = 4m$. With these parameters, the natural frequency of the poles calculated using Equation (4-32) is $\omega = 0.26$ Hz and the zero is located at $\omega = 0.48$ Hz. We can locate in Figure 4.8 both the poles and the zero. There are more pole-zero combinations at higher frequencies which would result from higher order Pade approximants for e^{-x}. With the model order used, the equivalent to Equation (4-31) in the time domain is

$$\frac{d^2\rho_\alpha}{dt^2} + \frac{6}{\tau} \frac{d\rho_\alpha}{dt} + \frac{12}{\tau^2} \rho_\alpha = C \frac{H^2}{\tau} \left(\frac{dq}{dt} + \frac{6}{\tau} q\right) \quad . \qquad (4\text{-}34)$$

Note that the steady state gain is independent of τ.
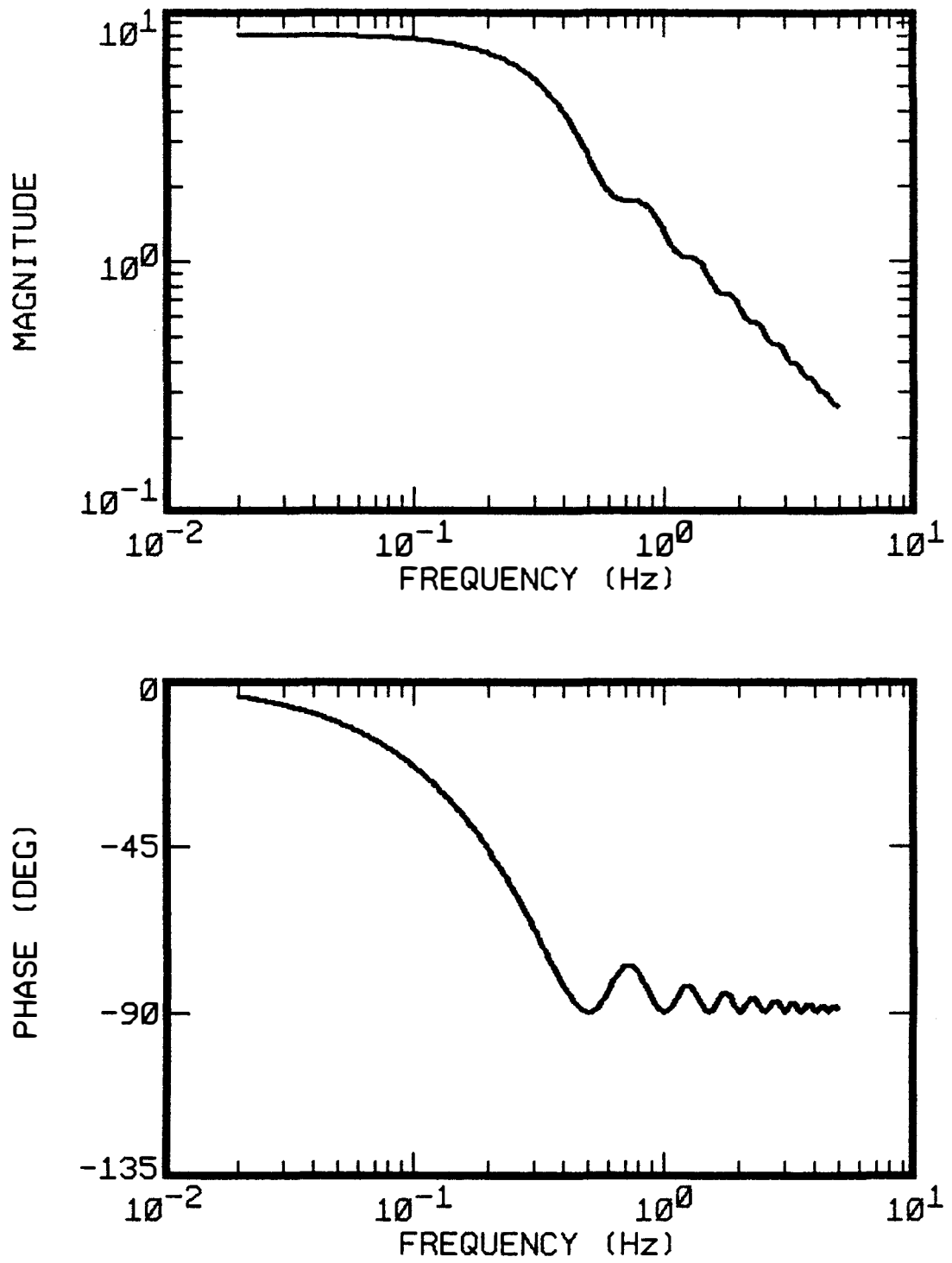
Figure 4.8 Channel transfer function.

In summary, we have proven that the channel transfer function has a pair of complex poles in the vicinity of 0.3 Hz. These must correspond to the pair of complex zeros found in the closed-loop reactivity-to-power transfer function (see Section 4.2). The frequency of these poles is inversely proportional to the bubble transit time in the core, τ, and they are introduced by the Φ^2 weighting (equation (4-26)) of the reactivity feedback rather than by channel thermohydraulic effects. Finally, in order to obtain the same natural frequency as the one associated with the complex zeros obtained for the Peach Bottom 3PT3 case (see Table 4.1), the equivalent bubble residence time must be $\tau = 1.63$ s. This residence time compares well with measured values.[77]

4.4.3 The Closed Loop Model

In the previous sections we have identified and modeled the various processes involved in the description of linear BWR dynamics. Now we can put them together and form the closed-loop model shown in Figure 4.9.
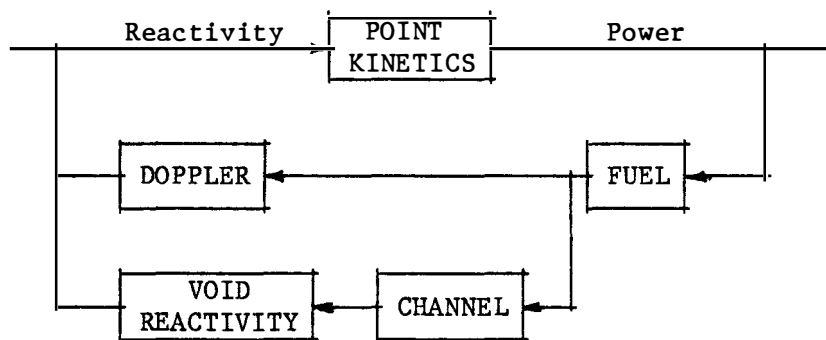
Figure 4.9 Block diagram of the reduced order model.

This figure shows a block diagram of this reduced order model. The forward loop is represented by the point kinetics approximation with a single group of delayed neutrons. The feedback has two loops, representing the Doppler and density reactivity feedbacks.

The time domain mathematical description of the closed loop for the reduced order model becomes

$$\frac{dn}{dt} = \frac{\rho_0 - \beta}{\Lambda} n + \lambda c + \frac{\rho}{\Lambda} \tag{4-35}$$

$$\frac{dc}{dt} = \frac{\beta}{\Lambda} n - \lambda c \tag{4-36}$$

$$\frac{dT}{dt} = a_1 n - a_2 (T - T_c) \tag{4-37}$$

$$\frac{d^2 \rho_\alpha}{dt^2} + \frac{6}{\tau} \frac{d\rho_\alpha}{dt} + \frac{12}{\tau^2} \rho_\alpha = C \frac{H^2}{\tau} \left(\frac{dq}{dt} + \frac{6}{\tau} q \right) \tag{4-38}$$

$$\rho = \rho_\alpha + DT \quad , \tag{4-39}$$

where D is the Doppler reactivity coefficient and the parameters a_1, a_2, and τ are functions of the effective physical constants calculated in Sections 4.4.1 and 4.4.2. Note that the quantities n and c are normalized to the steady state neutron density, N_0. The model parameters for the Peach Bottom 3PT3 case are summarized in Table 4.5.

Table 4.5

MODEL PARAMETERS FOR PEACH BOTTOM TEST CASE 3PT3

Parameter	Value	Units
a1	19.08	$K\ s^{-1}$
a2	0.19	s^{-1}
τ	1.63	s
C	-3.65×10^{-4}	K^{-1}
D	-2.61×10^{-5}	K^{-1}
β	0.0056	
Λ	4.00×10^{-5}	s^{-1}
λ	0.08	s^{-1}

4.5 Identification of Features in the Closed-Loop Transfer Function

The root locus for the model we just developed is drawn schematically in Figure 4.10. We observe that as the feedback gain is increased, the fuel and point kinetics poles merge. A further increase results in one of the poles canceling the delayed neutrons zero. The remaining pole yields the real pole observed in the closed-loop transfer function (see Section 4.2). The trajectory of the closed-loop poles is well defined and has the origin (K = 0) at the channel open-loop poles, which appear in the closed loop as the complex pair of zeros. The frequency of the β/Λ (21 Hz) pole increases slightly and the pole remains real for all values of K. With these observations we can now confidently identify not only the zeros but also the poles of the closed-loop transfer function.

ORNL–DWG 84-12082

CHANNEL THERMAL-HYDRAULICS

DELAYED NEUTRONS

FUEL

POINT KINETICS

$\dfrac{\beta}{\Lambda}$

CHANNEL

CLOSED LOOP ZEROS

CLOSED LOOP POLES

O   OPEN LOOP ZEROS

X   OPEN LOOP POLES
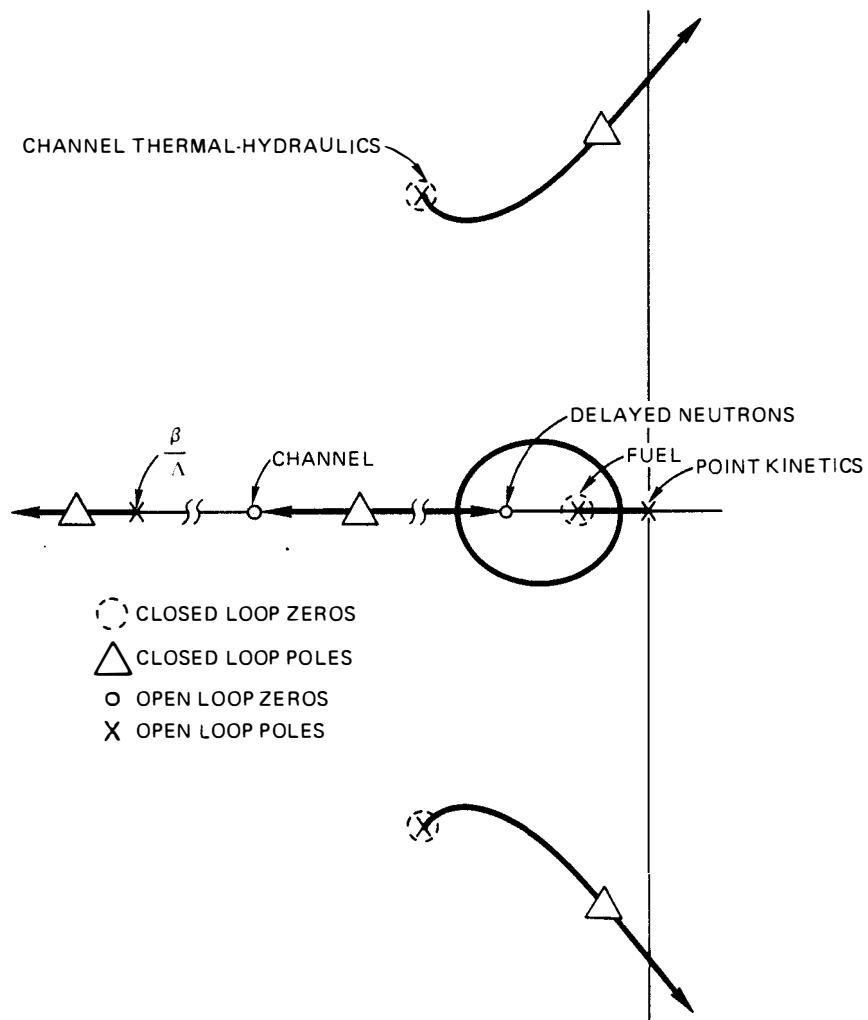
Figure 4.10   Root locus of the reduced order model and physical
processes associated with model features.

Table 4.6 and Figure 4.11 summarize the relationship between the

closed-loop transfer function features and their corresponding physi-

cal processes.

Table 4.6

RELATIONSHIP BETWEEN CLOSED LOOP TRANSFER FUNCION
FEATURES AND PHYSICAL PROCESSES

Features	Frequency (Hz)	Physical process
Low-frequency zero	0.03	Fuel element effective heat transfer coefficient
Compex pair of zeros	0.3	Reactivity feedback weighting of channel thermal hydraulics
Real pole	0.3	Fuel element
Complex pair of poles	0.3	Channel thermal hydraulics and density reactivity feedback
Real pole	20.	β/Λ pole in point kinetics equation

ORNL–DWG 84-12145
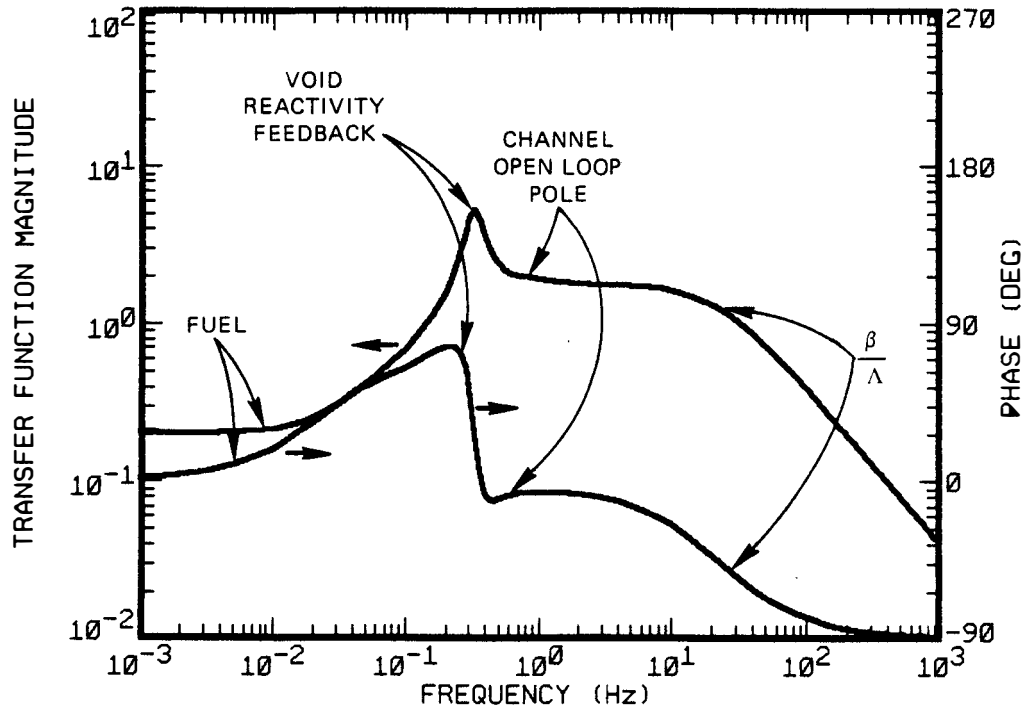
Figure 4.11 Transfer function of the reduced order model and physical processes associated with observable features.

CHAPTER 5

STABILITY MEASUREMENTS USING NOISE ANALYSIS

As seen in Section 2.3, three types of instabilities are considered to be possible in boiling water reactor operation: (a) plant instability, related to the reactor control systems; (b) local channel instability, related to two-phase flow dynamics of a heated channel; and (c) reactivity-type instability, related to the void reactivity feedback and its interaction with the neutronic loop. Experiments[50-59] have shown that commercial BWRs can be susceptible to the latter type of stability.

In this chapter we deal with the problem of determining the reactivity-type stability of an operating BWR by measuring only the output noise of the reactor power. This noise is caused by inherent fluctuations of process variables in the reactor such as flow, pressure, void fraction, etc. These fluctuations affect the power through the reactivity-to-power transfer function; therefore, the power noise contains information about the reactor dynamics and thus about its stability. However, since we can measure only the output noise and not the transfer function directly, it is impossible to distinguish between oscillations caused by the reactivity-to-power transfer function and those caused by the fluctuation in the process variables themselves. For instance, if the operating conditions were such that the channel flow were unstable, the power noise would show

an oscillation, but it would be due to the channel thermohydraulic instability rather than to the reactivity instability. Therefore, by measuring the noise, we can only measure the "output" stability; if an instability is found, it is up to the noise analyst to identify it with one of the three instabilities described before or with any other physical phenomenon.

The feasibility of using the neutron noise signal for stability related measurements in BWRs was first suggested by Thie.[9] Further studies[69-77] have shown that the power spectral density (PSD) of neutron noise in BWRs exhibits a noticeable resonance in the frequency range 0.3 to 0.7 Hz, as has been predicted by theoretical[48,63] and experimental studies[50-58] of the reactivity-to-power transfer function. Several papers have been published in the open literature describing methods to estimate BWR stability,[70,71,74-76] all of them based on one type or another of autoregressive (AR) modeling. However, there is ambiguity in the definition of decay ratio which is the parameter searched for in the stability analysis. Several authors use different definitions and, therefore, the results are not directly comparable. The problem becomes more important when a decay ratio obtained from noise data is to be compared with results of calculations or experiments.

The objectives of this section are: (a) to show that neutron noise has the necessary information for the determination of the reactivity-type stability in BWRs, (b) to evaluate several methods

to obtain the decay ratio from the noise data, and (c) to compare the results obtained using different definitions of decay ratio.

An overview of neutron noise in BWRs is presented in Section 5.1. In Section 5.2,the theoretical relationship between the neutron noise and the reactor stability is studied. Section 5.3 outlines several methods to obtain the necessary stability information from the noise. The noise techniques are evaluated against computer-generated data in Section 5.4. Section 5.5 outlines the proposed algorithm for estimating BWR stability from noise measurements. The new algorithm is validated through applications to computer generated and real BWR noise data.

5.1 Noise in Boiling Water Reactors

BWRs exhibit normally occurring fluctuations in power of the order of 4 to 8% peak-to-peak, which are normally referred to as noise. It is believed that the major source of normal BWR noise is the formation, collapse, and transport of steam voids in the reactor core. The voids modify neutron absorption and thermalization, thereby introducing perturbations in cross sections and thus in the neutron density as seen by the in-core fission detectors.[73]

Figure 5.1 shows the power spectral densities of a typical local power range monitor (LPRM) string at levels A, B, C, and D along with the PSD of the average power range monitor (APRM) for a BWR operating at rated power and flow. Figure 5.2 shows the PSD of
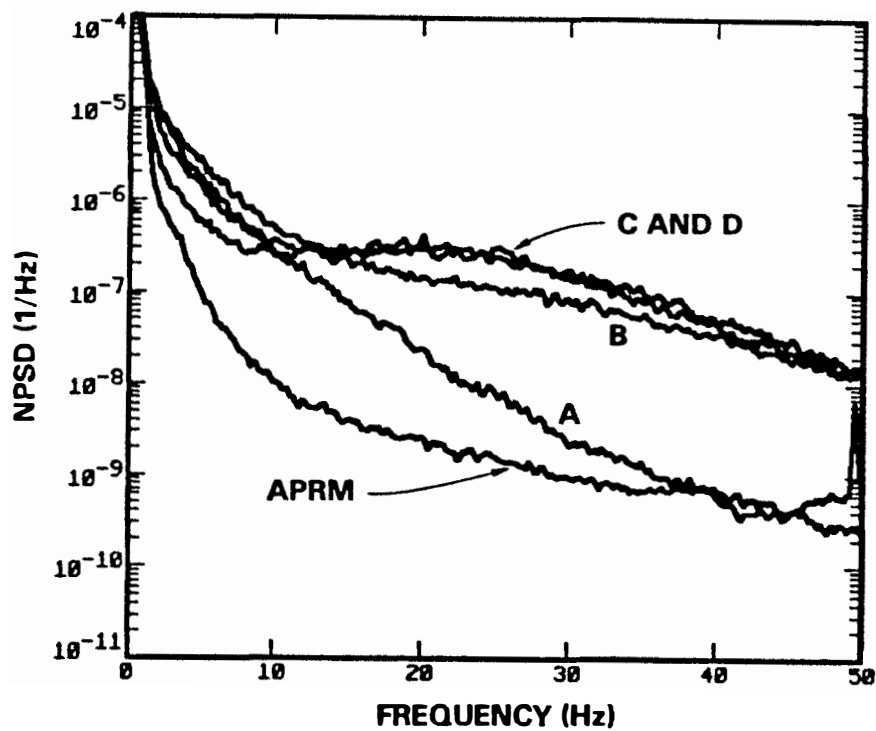
Figure 5.1 Normalized power spectral densities of a typical APRM
and LPRM string signals in a BWR.
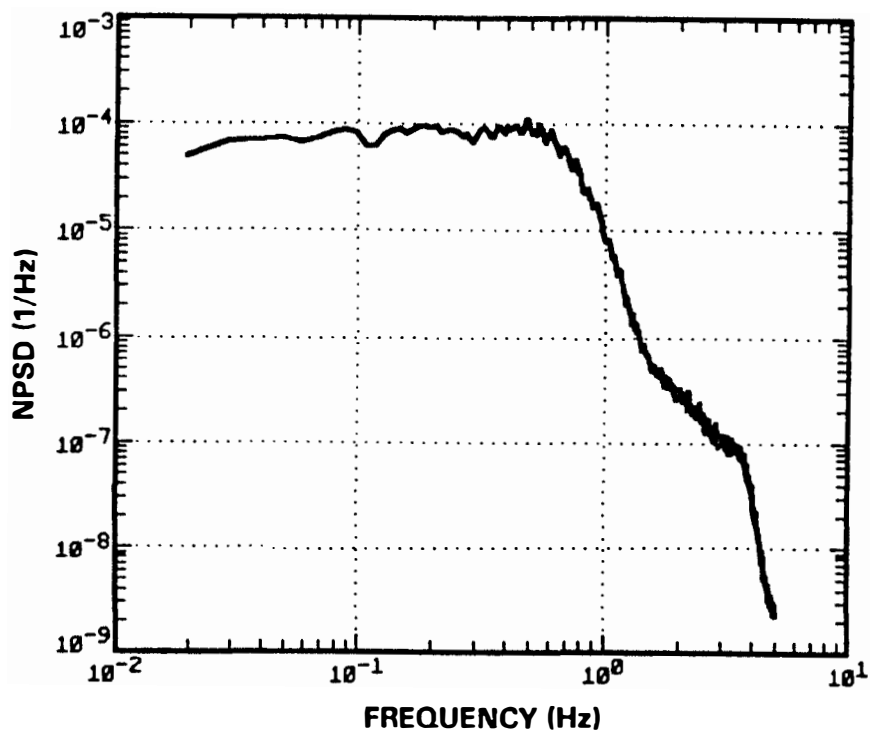
Figure 5.2 Normalized power spectral density of a typical APRM
signal at full power.

the same APRM signal; the characteristic 0.5 Hz resonance and associated double-pole break frequency is clearly observed.

Experience has shown that the BWR neutron noise has two components[79] the first is dominant at high frequencies (above 1 Hz) and its main characteristic is that it is radially uncorrelated from channel to channel. However it is axially correlated within the same channel and a time delay can be measured between this component of two LPRM detectors at different levels. The second component, which is dominant at low frequencies, is both radially and axially correlated; moreover, it oscillates in phase all over the core and is correlated with process variables such as core flow and pressure.

The major source of radially uncorrelated noise is the perturbations in neutron flux caused by voids in the vicinity of the neutron detectors. This generally accounts for the increased noise seen by the LPRM detectors at high frequencies. Note that the detectors near the top of the core (detectors C and D in Figure 5.1) see more radially uncorrelated noise (presumably because there are more voids) than those near the bottom.

The APRM signal is a measure of the radially correlated sources of neutron noise in the core, whereas the difference between APRM and LPRM noise is an indication of the uncorrelated noise at a given detector location. Because the APRM signal is made up of the sum of 20 to 30 LPRM signals, any portion of the individual signals that is uncorrelated will tend to be reduced in the total signal by a factor of approximately one over the square root of the number of signals in

the sum. The overall fluctuations in core-voids introduce a feedback to the core dynamics which, when coupled with the thermal hydraulics, causes correlated low-frequency fluctuations. This is believed to be the source of the characteristic 0.5 Hz resonance seen in the PSD of all APRM signals.

5.2 Relationship between BWR Noise and Stability

The stability of BWRs is normally evaluated using the reactivity-to-power transfer function obtained either from deterministic codes[42-49] or from deterministic experiments using pseudorandom binary signals (PRBS) to stimulate reactivity perturbations.[53-59] However, when trying to estimate the stability from noise measurements, the transfer function is not measurable because the input is unknown and stochastic. It will be shown in this section that, in spite of this handicap, the stability can be estimated from noise measurments; in other words, that there is stability information in the neutron noise. Section 5.3 will describe some of the mathematical methods that can be used to extract this information.

Aspreviously stated, there are two components in the normal neutron noise, but only the correlated part is related to the reactor stability. This part is believed to be caused by perturbations in global reactivity, which are seen in the neutron density through the reactivity-to-power transfer function. The neutron noise, n(s), as seen by the in-core detectors is given, then, by the expression

$$n(s) = D(s)\ G(s)\ \rho(s)\ +\ v(s) \tag{5-1}$$

where

$\rho(s)$ = Reactivity perturbations

$G(s)$ = Reactivity-to-power transfer function

$D(s)$ = Detector transfer function

$v(s)$ = Uncorrelated noise introduced by the detection

processes and local effects.

Therefore, $n(s)$ contains not only the poles of $G(s)$ but also the ones of $\rho(s)$. The reactor stability is related to the position of the most unstable pole of $G(s)$; thus, if Equation (5-1) is valid, the necessary information is included in the neutron noise. The point that needs to be proven is that the correlated part of the neutron noise is caused by reactivity fluctuations, in which case Equation (5-1) holds.

Figures 5.3a and b show the PSD of an APRM signal of the Dresden-2 reactor[59,105] at two operating conditions: (a) full power and (b) 52.7% power and 38.9% flow, which corresponds to the minimum recirculation pump speed; this is the most unstable condition within the normal operation map. The estimated decay ratio (obtained from noise analysis) of condition (a) is 0.15, and the decay ratio of condition (b) is 0.35. Two main differences exist between the PSDs for these two conditions: first, the characteristic frequency dropped from 0.71 Hz in the full-power case to 0.42 Hz in the low-flow condition. Second, the peak at the characteristic frequency is more pronounced and looks sharper in the low-flow condition. The two above differences can be explained by a change in the system transfer function:

(a) Nominal conditions
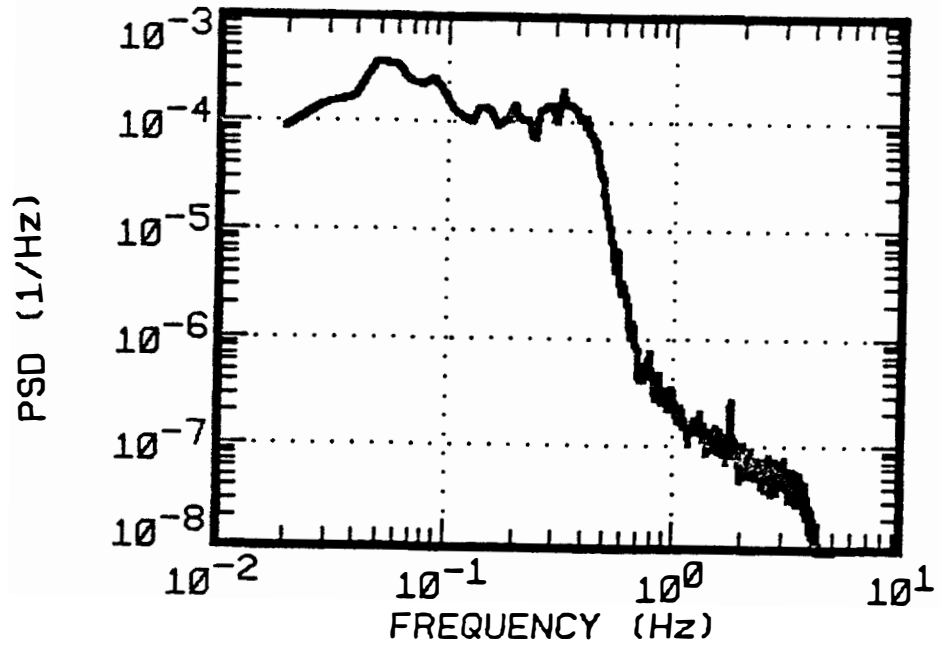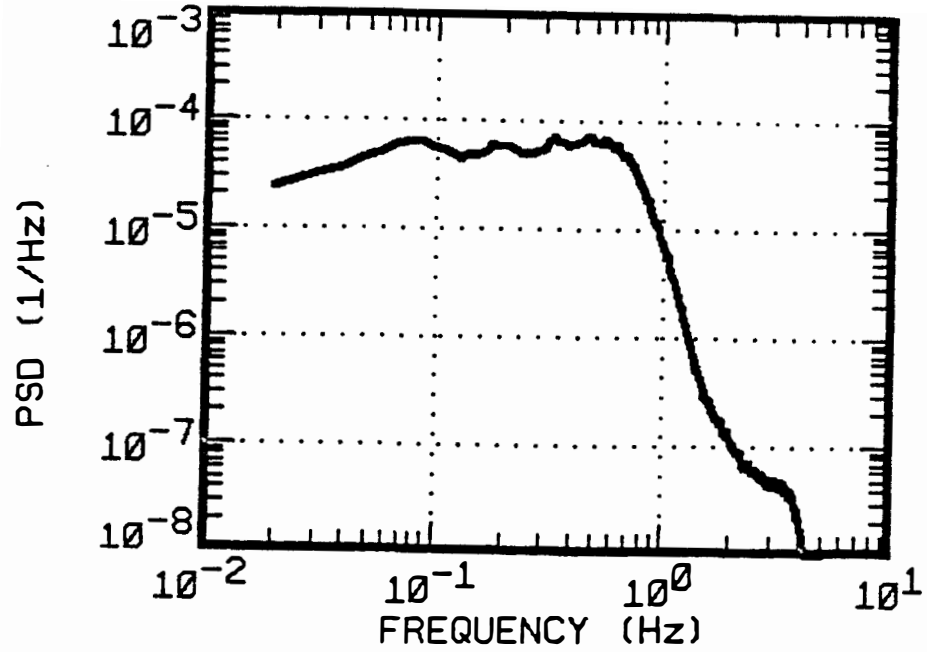
(b) Minimum recirculation pump speed

Figure 5.3 Power spectral density of a typical BWR APRM signal.

(a) It was established in Chapter 4 that the characteristic frequency of the reactivity-to-power transfer function is inversely proportional to the sweep time of the steam bubbles in the core. Therefore, a lower frequency is expected at the low-flow condition due to the reduced channel inlet velocity. The reduction in frequency as well as its actual value is in good agreement with computer simulations.

(b) If the peak observed in the neutron noise PSD corresponded to the peak in the reactivity-to-power transfer function, then it would have to be sharper at the low-flow condition due to the reduced stability. Furthermore, the numerical value of the decay ratio obtained from the neutron noise is in good agreement with computer simulations.[105,110]

Both of the above points imply that the peak in the neutron noise PSD is due to the characteristic peak of the reactivity-to-power transfer function; therefore, the correlated component of the neutron noise must be caused by global reactivity fluctuations and, thus, it contains stability information.

Once we have confirmed that Equation (5-1) is valid, identification of the poles of $n(s)$ can be used to evaluate the reactor stability. However, since we can measure only the output noise and not the transfer function directly, it is impossible to distinguish between oscillations caused by the reactivity-to-power transfer function and those caused by the fluctuations in the process variables

themselves. For instance, if the pressure control system were out of tune, and it were introducing a sinusoidal (DR = 1.0) perturbation in pressure (and consequently in reactivity), the neutron noise would show an undamped oscillation with DR = 1.0, but it would be due to the control system stability, not to the reactor thermal hydraulics. Therefore, by measuring the noise we can only measure the "output" stability; if an instability is found, it is up to the noise analyst to identify it with a physical phenomenon.

5.3 Methods to Measure Output Stability

In this section we discuss the characteristics of several methods to measure the output stability.

5.3.1 Autocorrelation Function

The autocorrelation function of a sampled signal, $x(t)$, is defined as

$$C(\tau) = \lim_{T \to \infty} \frac{1}{T} \int_0^T x(t)x(t-\tau) \, dt \quad , \tag{5-2}$$

where it has been assumed that the signal is zero before the start of the measurement at time $t = 0$. The correlation can be estimated from a sampled sequence, $x(t_i)$, as[66]

$$C(n\Delta T) \approx \frac{1}{N} \sum_{i=1}^{N} x(t_i) \, x(t_i - n\Delta T) \quad . \tag{5-3}$$

The standard deviation of the error made in the estimation of typical autocorrelations in BWRs is approximately constant for all

time lags.[79] The autocorrelation of stable systems decay

exponentially; therefore, the relative error in the estimation

increases exponentially. The more unstable the system, the slower the

autocorrelation decays and, therefore, the smaller the relative error;

the error in the estimation of the autocorrelation function thus

depends on the stability of the system itself.

The apparent decay ratio (see Chapter 3) can be measured

directly from the correlation function. The asymptotic decay ratio

can be estimated as the limit of the series formed by the decay ratios

between every two consecutive peaks. The apparent decay ratio is a

well-defined quantity and can be estimated using relatively small

lengths of data; however, the asymptotic decay ratio depends on the

values of the autocorrelation at large lags and, therefore, the preci-

sion of the estimation depends largely on the system stability.

Nevertheless, given enough time for the autocorrelation to converge,

this is the most accurate method of measuring the asymptotic decay

ratio because it does not rely on any assumption or model fitting.

5.3.2 Autoregressive Modeling

The autoregressive (AR) modeling technique[69,70,75] can be

thought of as an attempt to improve the convergence rate of the auto-

correlation function. A univariate AR model of order N for the time

series x(t) is of the form

$$x(t) = \sum_{k=1}^{N} A_k \, x(t-k\Delta t) + v(t) \quad , \tag{5-4}$$

where A_k are the AR parameters and $v(t)$ is a residual noise sequence whose variance is minimized by the model. Given a model order the model parameters, A_k, satisfy the Yule Walker equations[69,70,75] (see Appendix A for details).

$$C_i = \sum_{k=1}^{N} A_k \, C_{i-k} \quad , \tag{5-5}$$

where C_i is the autocorrelation function evaluated at time lag $\tau = i\Delta t$.

The AR parameters for a model of order N are defined if the correlation is known for the first N time lags. Then, as shown in Appendix A, the AR-consistent correlation function, C_i', is given by the expression

$$C_i' = \sum_{k=1}^{N} A_k \, C_{i-k}' \qquad ; \ i=N+1,\ldots \tag{5-6}$$

and

$$C_i' = C_i \qquad\qquad ; \ i=0,1,\ldots,N \quad . \tag{5-7}$$

In this manner we see that the AR model takes the first N lags in the correlation (the ones with smaller error) and predicts the rest in a consistent manner. The asymptotic decay ratio can be estimated from the AR-consistent correlation even if the real correlation is not converged enough to define it.

A serious problem with the AR modeling technique is the determination of the "optimal" model order: low orders use well-converged correlation lags, but they do not have enough parameters to describe the system accurately; high order models, on the other hand, have

enough parameters to describe the system, but they use unconverged correlation lags. Therefore, there is an "optimal" order that supposedly produces a minimal error. There are several criteria described in the literature to determine this optimal model order; the most commonly used is Akaike's information criterion[104] which takes into account the amount of data used, the model order, and the standard deviation of the residual sequence. The model order is chosen so that the joint probability of all the measurements (i.e., the likelihood function) is maximum. This is equivalent to minimizing the following function

$$\text{AIC} = M \ln(\sigma^2/C_0) + 2N \quad , \tag{5-8}$$

where

M = Total number of samples used

σ^2 = Variance of the residual noise

N = AR model order.

Assuming that the AR model order is determined, there are several methods of estimating the asymptotic decay ratio from the AR parameters:

(a) The first approach is to use the AR-consistent correlation function defined by Equation (5-6) and measure the decay ratio directly. This correlation is normally well behaved and the decay ratio between consecutive peaks converges rapidly to the asymptotic value. The apparent decay ratio of the autocorrelation can be measured directly.

(b) Another approach is to obtain the impulse response of the AR model. It is shown in Appendix A that the AR model assumes that the system has only one more pole than zeros (i.e., 1-zero/2-poles, 2-zeros/3-poles, etc.). The impulse response, h_i, of this type of systems is equal to an initial value response and can be calculated recursively according to the equation

$$h_i = \sum_{k=1}^{N} A_k \, h_{i-k} \quad , \tag{5-9}$$

with initial conditions $h_0 = 1$ and $h_{-i} = 0$.

A similar estimate can be obtained by using the properties of the cross-correlation function (see Appendix A)

$$h_i = C_i' - \sum_{k=1}^{N} A_k \, C_{i+k}' \quad , \tag{5-10}$$

where C_i' is the AR-consistent autocorrelation function.

(c) A third method is based in the frequency response of the AR model and determines the position of the most unstable pole of the model. The Fourier transform of Equation (5-4) is

$$X(\omega) = \sum A_k \, X(\omega) \, D^k + V(\omega) \quad , \tag{5-11}$$

where D is the backshift operator

$$D = e^{-2\pi i \omega \Delta t} \quad ; \tag{5-12}$$

therefore,

$$X(\omega) = \frac{V(\omega)}{1 - \sum\limits_{k=1}^{N} A_k \, D^k} \, . \qquad\qquad (5\text{-}13)$$

If $V(\omega)$ is close to a white noise, the poles of $X(\omega)$ are the zeros of the denominator, which is a polynomial in D. Note that there are N poles for $0 < \omega <$ Nyquist-frequency. The zeros of the denominator can be obtained by using Newton's method in the complex domain. Figure 5.4 shows the magnitude squared of equation (5-13) for a typical 30th order model. The figure has been plotted in three dimensions to display the whole complex or s-plane. The left vertical axis corresponds to the frequency axis; thus, the magnitude in this plane is the PSD.

Newton's method requires a starting point in the s-plane which is close to the solution. This first guess can be estimated in several ways. The most straightforward method would be to search for the frequency, ω, of the peak of interest in the PSD; then, the starting point would be $s = 0+j\omega$. This approach works most of the time, specially if the pole is close to the imaginary axis (decay ratio > 0.5), but the iterations might possibly converge to a different pole if it is far away as can be seen in Figure 5.4. The starting guess can be improved if an estimate of the real part of the pole can be obtained. Following a conformal mapping procedure in the closed-loop transfer function similar to the one used
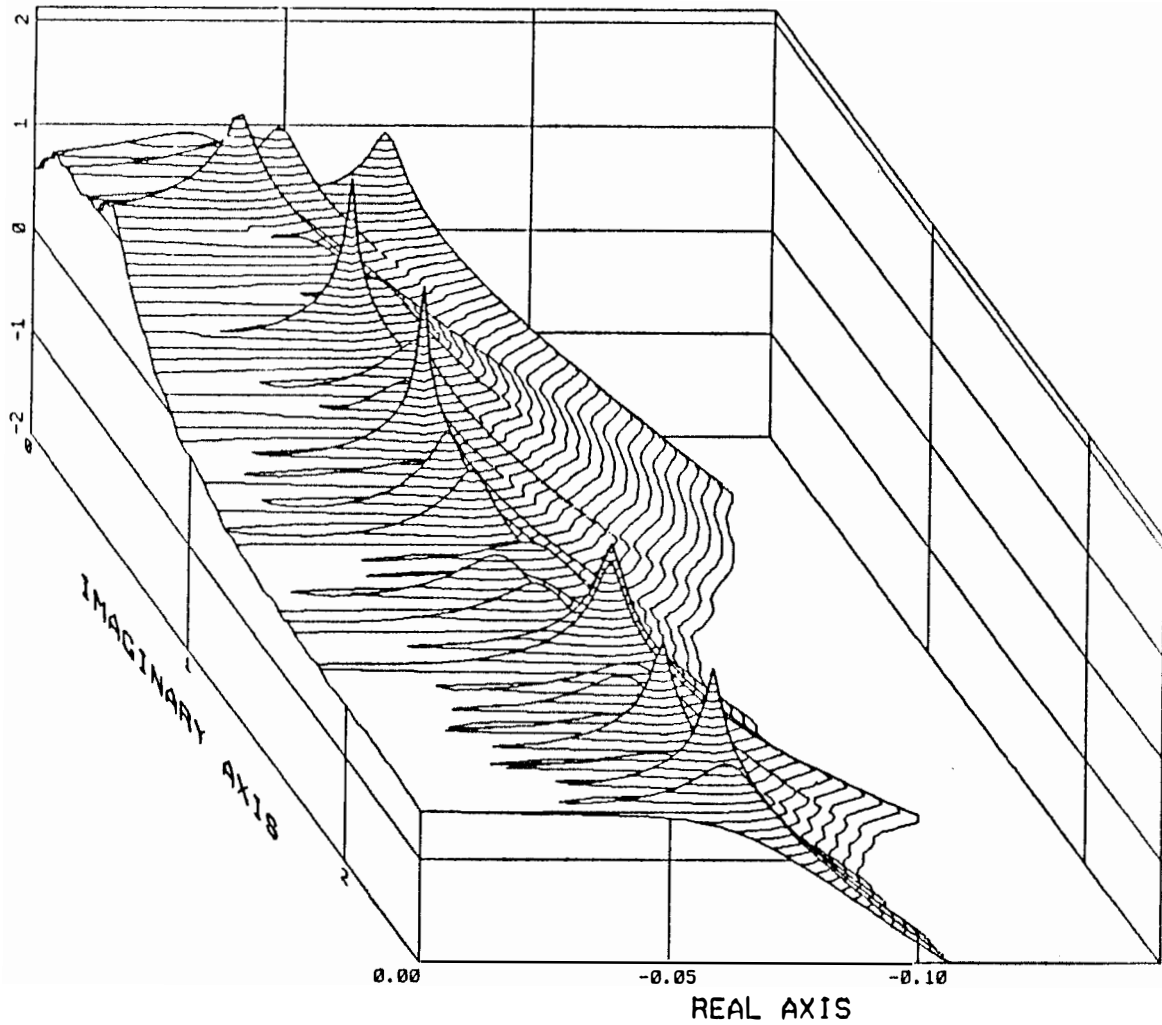
Figure 5.4   Magnitude square of a typical 30th order AR model in
the s-domain.

by Smith[101] in the open-loop, it can be shown that an estimate
of this real part is

$$\sigma = \frac{d\omega}{d\phi}\Big|_{\omega=\omega_{max}} \tag{5-14}$$

where ϕ is the phase of the transfer function (Equation
(5-13)) in radians and σ will have the units of ω. ω_{max}
corresponds to the frequency where the derivative of the phase
is a maximum. An improved starting guess is then $s = \sigma+i\omega$.

If a different starting point is used for each of
the peaks in the PSD, a set of AR poles is obtained. If the
one with the smallest real part is chosen, it should yield
the same asymptotic decay ratio as methods (a) and (b).
Note, however, that the decay ratio for the other peaks is
also available. This is the main advantage of this approach
compared to the other two.

5.3.3 Power Spectral Density Fit

A procedure similar to that in part (c) of the last section
cannot be applied to the measured PSD directly because the PSD is
only defined along the frequency axis. However, a nonlinear fit
with poles and zeros can be performed on this data and an asymptotic
decay ratio can be obtained from the fitted parameters.

The functional form of the PSD is a ratio of polynomials
with even powers in frequency. The fit should be weighted by the
inverse variance of the PSD, given by the expression[66]

$$\sigma^2 = \frac{PSD(\omega)}{N} , \tag{5-15}$$

where N is the number of blocks used in the fast Fourier transform (FFT) analysis to generate the PSD.

5.4 Validation of Noise Techniques Against
Computer-Generated Data

To demonstrate the validity of the methods described in Section 5.2, noise data for a system with 3 zeros and 4 poles were obtained from computer simulations by solving the corresponding fourth-order differential equation driven by a Gaussian white noise. The four poles were located at $s = -0.05 \pm 0.5i$ Hz and $s = -0.1 \pm 0.1i$ Hz. The zeros were located at $s = -0.05$ Hz and $s = -1 \pm i$ Hz. A time series of 3 hours of data was generated and AR models of orders 10, 30, and 50 were utilized to fit the generated data. The AR-consistent correlations are plotted versus the computer-generated correlations in Figure 5.5. The AR-PSDs for the three model orders are shown in Figure 5.6.

In addition, a nonlinear fitting routine was used to fit a second order system (only 2 poles) and a 3-zeros/4-poles model. The results of the fits are shown in Figures 5.7a and b.

After all these data were reduced, a stability analysis was performed using the techniques described previously. The results are shown in Table 5.1. It can be observed that all methods worked reliably except, as expected, the second order fit.
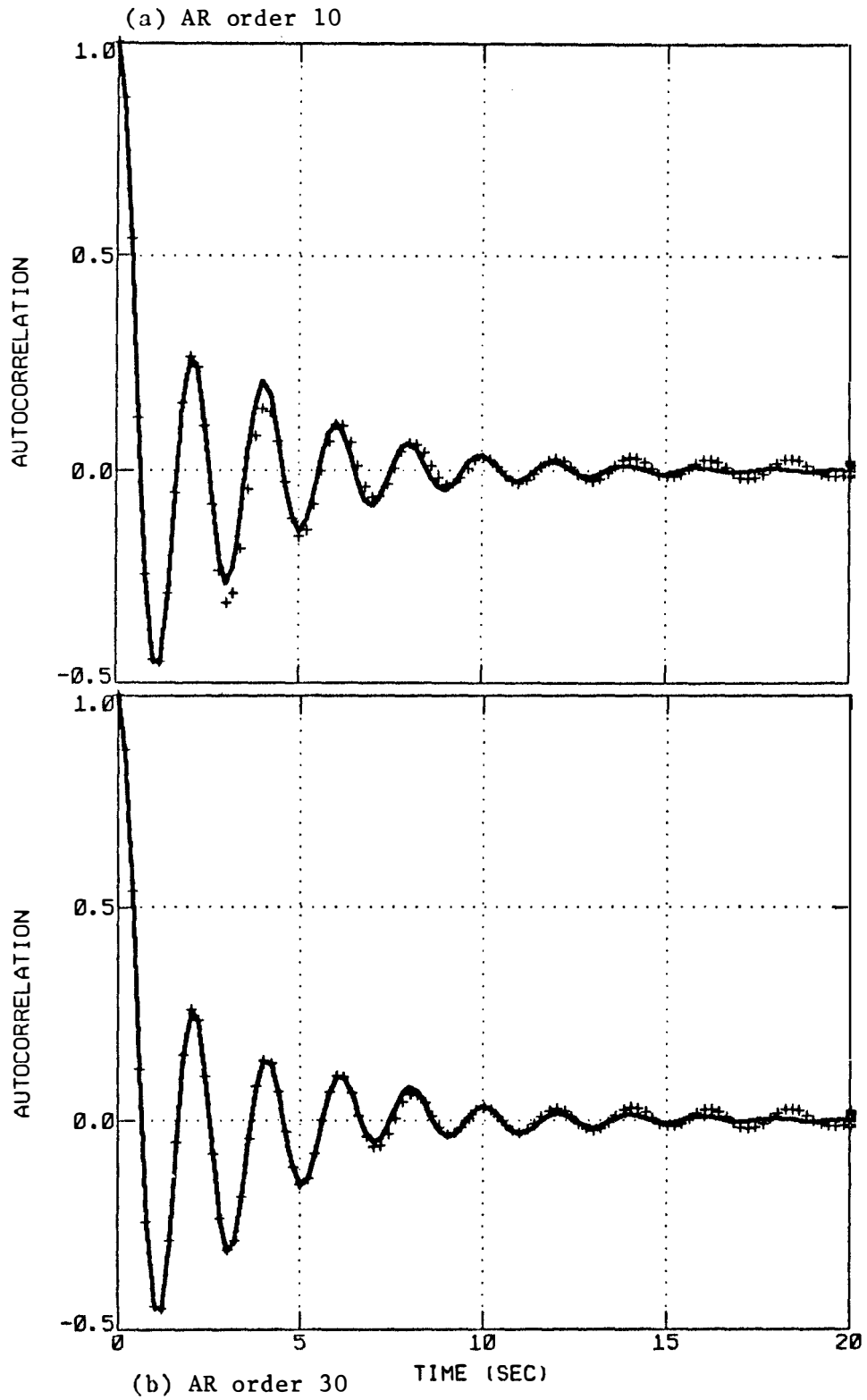
Figure 5.5 Comparison of AR-consistent versus measured
autocorrelation functions for computer generated data.
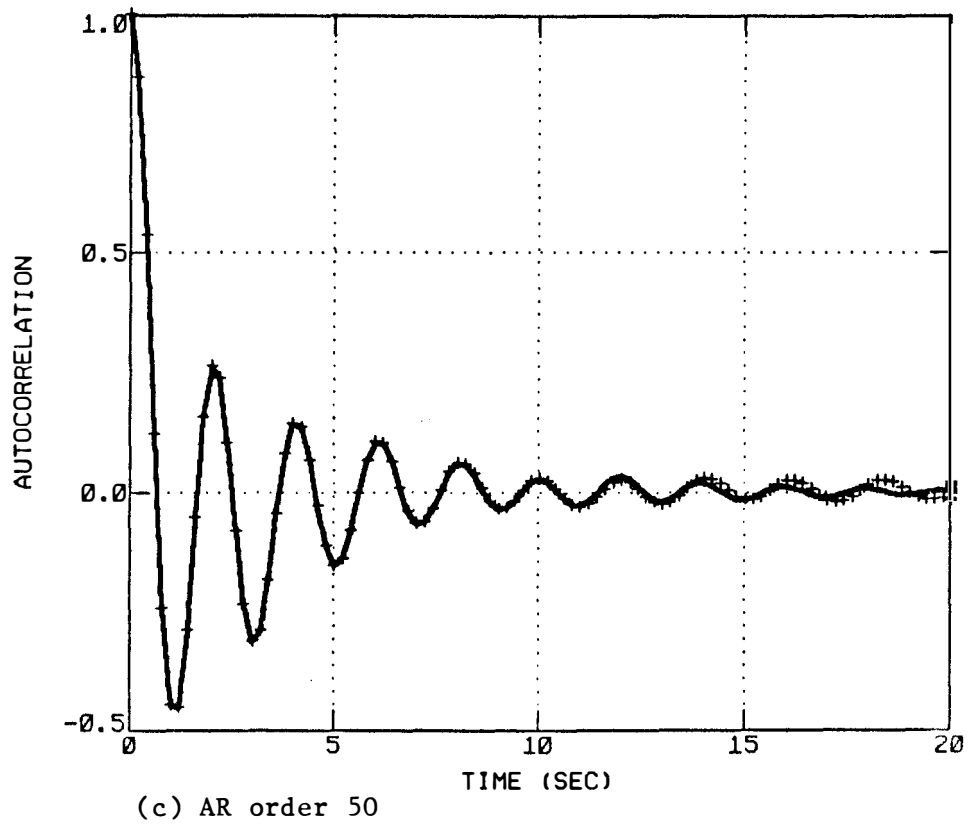
(c) AR order 50

Figure 5.5 (continued).

(a) AR order 10

(b) AR order 30
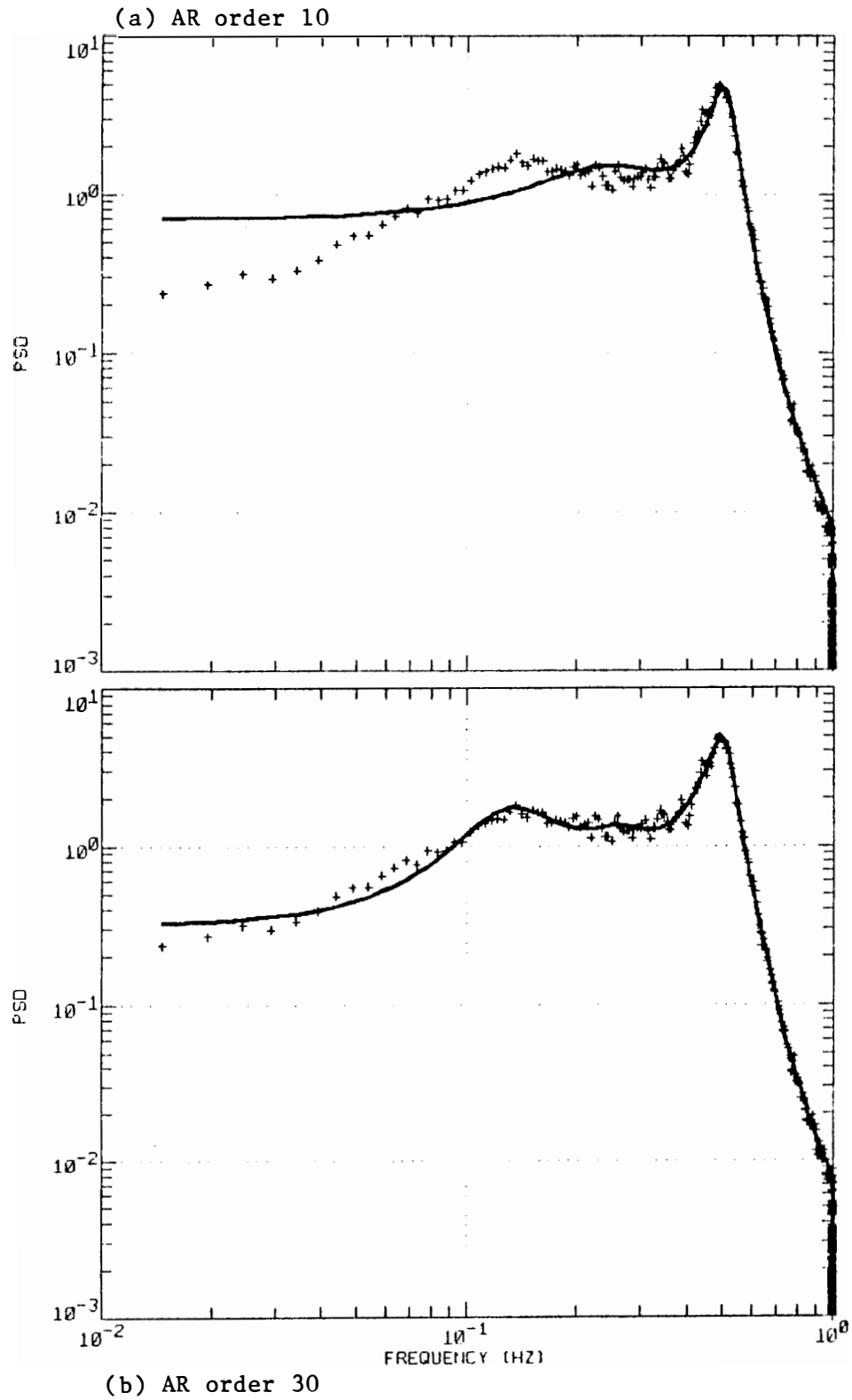
Figure 5.6 Comparison of PSD from AR model versus the fast Fourier transform method for computer generated data.
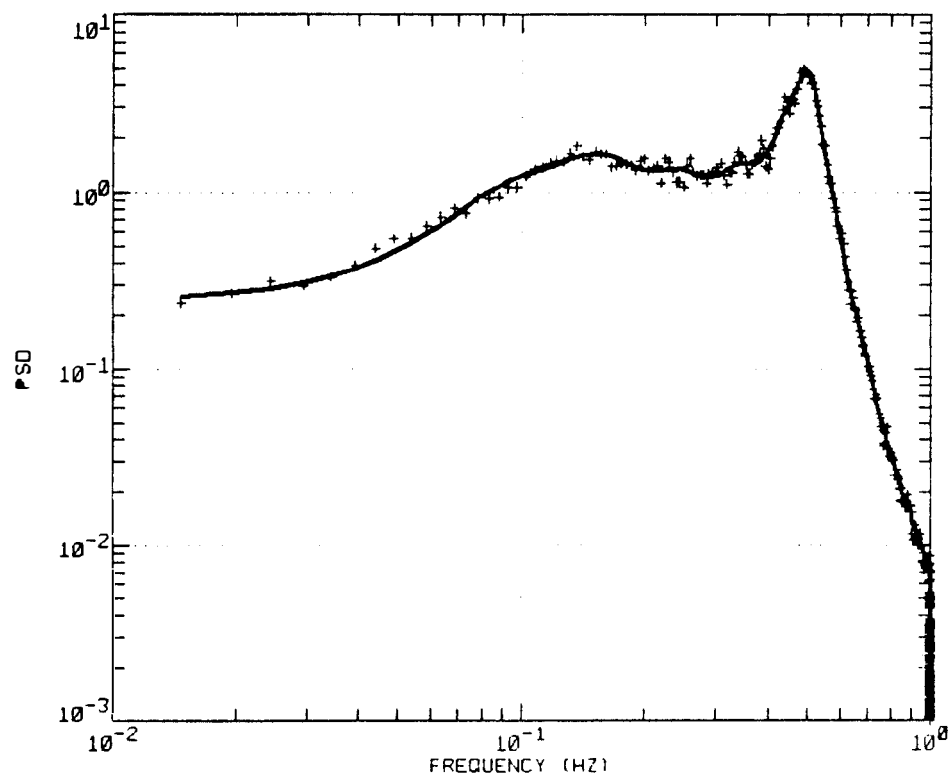
(c) AR order 50

Figure 5.6 (continued).

(a) Second order fit

(b) 3-zeros/4-poles fit
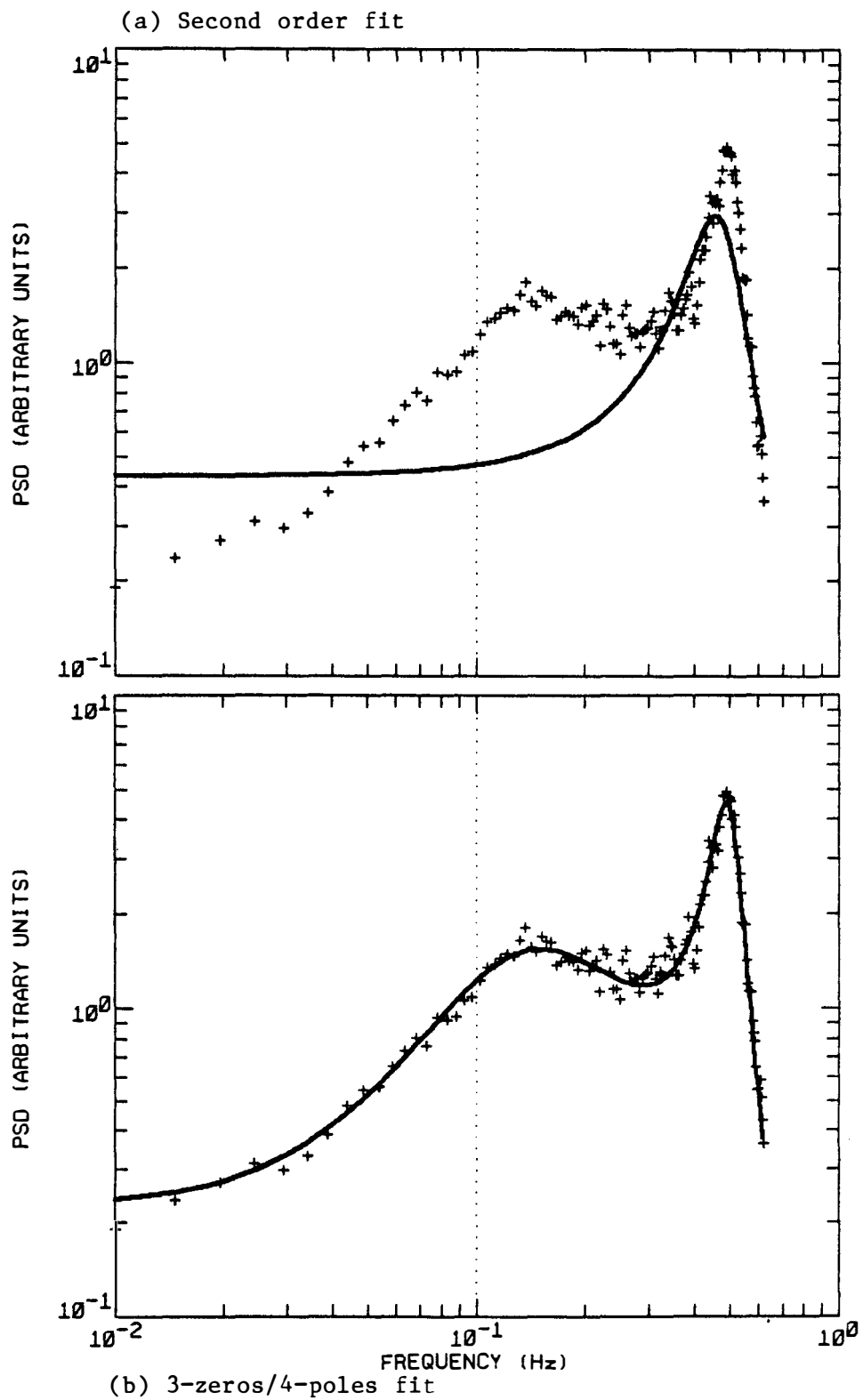
Figure 5.7 Comparison of non-linear fit with actual PSD for computer generated data.

Table 5.1

RESULTS OF THE STABILITY ANALYSIS OF
COMPUTER GENERATED DATA

Method	Asymptotic Decay Ratio	Apparent Decay Ratio Impulse Response	Apparent Decay Ratio Correlation
Exact	0.53	0.24	0.27
Correlation function	0.50	--	0.27
PSD fit			
0/2 order	0.28	--	--
3/4 order	0.54	--	--
AR model order 10			
Most unstable pole	0.49	--	--
Impulse response	0.49	0.37	--
Consistent correlation	0.49	--	0.37
AR model order 30			
Most unstable pole	0.55	--	--
Impulse response	0.55	0.25	--
Consistent correlation	0.55	--	0.26
AR model order 50			
Most unstable pole	0.54	--	--
Impulse response	0.54	0.24	--
Consistent correlation	0.54	--	0.27

5.5 The Algorithm to Estimate BWR Stability

from Noise Measurements

After consideration of the advantages and disadvantages of all the methods presented so far to estimate the BWR stability from noise measurements, an integrated algorithm has been developed which performs this task in an optimal manner. This algorithm has been implemented as a package of FORTRAN-77 subroutines, which are described in detail in Appendix B. In this section, only the highlights of the present methodology are presented. The main advantages of this algorithm with respect to the previously existing methodologies[70-76] are

(a) First, the asymptotic decay ratio of the reactivity-to-power transfer function is obtained. Whereas the result of previous methods was an apparent decay ratio of the whole system. In contrast, if more than one resonant frequency is present in the reactor response, the present method makes an effort to try to determine which one corresponds to the reactor transfer function.

(b) This algorithm is designed so that it can perform in an automated manner, without the need of an expert noise analyst to validate the results. Most of the effort during the calculation is directed towards validating these results by using a priori knowledge, which is provided as a series of heuristic rules, and a series of self-consistency checks.

Since the emphasis of the present technique is on having a reliable, completely automated system, considerable effort is spent on the determination of the confidence level of the estimate based on a priori knowledge and consistency checks. It has been our experience that any of the methods described in the previous sections will fail (i.e. yield the wrong decay ratio) given the appropriate combination of poles and zeros in the neutron PSD. For this reason, the DR is computed in three different ways for each set of data. A set of heuristic rules is built into the subroutines to check for self-consistency among the set of the estimated DRs. A best estimate DR is obtained and then checked again for consistency with available noise descriptors, such as the autocorrelation function or the AR impulse response. In addition, a priori knowledge about BWR behavior, such as the frequency of the charateristic resonance and the expected shape of the PSD, is tested. The three different DR estimates are:

(a) apparent DR of the correlation function, (b) asymptotic DR of the AR impulse response, and (c) DR estimated from the position of the most unstable pole of the frequency response equivalent of the AR model.

An estimate of the statistical precision is also obtained. For this purpose, the total data length is divided into a set of blocks and DRs are estimated for each block as well as for the average. In addition to the pure statistical error, the DR estimated from the average PSD compared to the individual results yields an estimate of the systematic or bias error.

This diagnostic technique has been implemented in the ORNL on-line surveillance system[103] and will be tested in a commercial BWR. We expect a rough estimate of the DR every ten minutes and a more accurate one, including the error estimate, approximately every hour.

5.5.1 Application of the Algorithm to Computer Generated Data

The present algorithm has been extensively validated against computer-generated data. For this purpose, autocorrelation functions corresponding to systems with different sets of poles and zeros were fed into the algorithm and the results are compared with the theoretical answers. The results of these comparisons were excellent. Some examples are presented in Tables 5.2 and 5.3. These tables contain the pole/zero configuration of the system along with the AR model order used and the results of the algorithm. These results are

Sampled Δt. This corresponds to the original sampling time used to calculate the autocorrelation function.

AR Δt. The algorithm has the capability of resampling the raw autocorrelation function in order to increase the effective sampling time for the AR model. This will occur if the original sampling time is too fast or if the algorithm senses that there are low-frequency components which need to be modeled in order to obtain an acceptable result. This resampling is performed if the maximum model order allowed for the calculation is not large enough to minimize Akaike's information criterion.[104]

Table 5.2

RESULTS OF THE NOISE STABILITY ALGORITHM
APPLIED TO COMPUTER GENERATED DATA
CASES (a) THROUGH (d)

	Case (a)	Case (b)	Case (c)	Case (d)
Poles (Hz)	−0.0441±0.4i	−0.0441±0.4i	−0.0441±0.4i	−0.0441±0.4i
Zeros (Hz)	––	−1.0	−0.1	−0.01
AR Order	10	10	20	30
Sampled Δt	0.1	0.1	0.1	0.1
AR Δt	0.1	0.1	0.2	0.2
Decay Ratio	0.50	0.50	0.54	0.51
Nat. Freq. (Hz)	0.40	0.40	0.40	0.42
Cofidence Level	7	7	6	4
DRC	0.50	0.50	0.54	0.51
FNC	0.40	0.40	0.40	0.42
IERC	0	0	0	0
DRS	0.50	0.50	0.54	0.51
FNS	0.40	0.40	0.40	0.42
IERS	0	0	−1	0
DRP	0.50	0.50	0.54	0.51
FNP	0.40	0.40	0.40	0.42
IERP	0	0	0	0

Table 5.3

RESULTS OF THE NOISE STABILITY ALGORITHM
APPLIED TO COMPUTER GENERATED DATA
CASES (e) THROUGH (h)

	Case (e)	Case (f)	Case (g)	Case (h)
Poles	$-0.0441\pm0.4i$	$-0.0067\pm0.4i$	$-0.147\pm0.4i$	-0.1
(Hz)	$-0.02\pm0.1i$	$-0.02\pm0.1i$	$-0.02\pm0.1i$	
Zeros	-0.1	-0.1	-0.1	---
(Hz)	$-0.5\pm0.5i$	$-0.5\pm0.5i$	$-0.5\pm0.5i$	
AR Order	20	20	15	10
Sampled Δt	0.1	0.1	0.1	0.1
AR Δt	0.4	0.4	0.4	0.1
Decay Ratio	0.51	0.90	0.19	-999
Nat. Freq. (Hz)	0.40	0.40	0.37	-999
Cofidence Level	3	4	2	-7
DRC	0.24	0.48	0.28	-999
FNC	0.40	0.40	0.40	-999
IERC	0	0	0	-1
DRS	0.30	0.90	0.32	-999
FNS	0.10	0.40	0.10	-999
IERS	0	-1	0	-2
DRP	0.51	0.90	0.19	-999
FNP	0.40	0.40	0.37	-999
IERP	0	0	-2	-3

Decay ratio. This is the best estimate of the asymptotic decay ratio produced by the algorithm and it is the main result.

Natural frequency. This is the natural frequency of oscillation.

Confidence level. This number represents the goodness of the fit. A level of +7 is maximum and a level of -7 is minimum. As long as the level is greater than 3, the estimate is probably good. If the level is negative, the estimate is worthless.

DRC. The apparent decay ratio of the autocorrelation function.

FNC. The apparent natural frequency of the autocorrelation function.

IERC. Error code generated by subroutine DRCORR while calculating the apparent decay ratio of the autocorrelation. See the subroutine listing for details.

DRS. The asymptotic decay ratio of the impulse response function generated from the AR model.

FNS. The asymptotic natural frequency of the impulse response.

IERS. Error code generated by subroutine DRTIME while calculating the asymptotic decay ratio of the impulse response. See the subroutine listing for details.

DRP. The asymptotic decay ratio estimated from the frequency domain pole search.

FNP. The asymptotic natural frequency estimated from the frequency domain pole search.

IERP. Error code generated by subroutine DRFREQ while calculating the asymptotic decay ratio from the frequency domain pole search. See the subroutine listing for details.

The first case in Table 5.2 corresponds to a pure second-order system with a decay ratio of exactly 0.5. We observe that for such a well behaved system, all methods work perfectly. Indeed, the confidence level is +7 and all estimates agree.

A major problem with AR models is their inherent difficulty with representing zeros. Indeed, AR models are formed just by poles in the frequency domain, and it takes an infinite number of poles to represent exactly a zero. Cases (b) through (d) in Table 5.2 confirm this idea, but they show that the present algorithm is capable of performing acceptably well even under adverse circumstances. For these three cases we conserve the same poles than for case (a) (i.e., decay ratio 0.5) but we introduce a zero at different frequencies. In theory, the AR model should have the greatest problem with the lowest frequency zero (case (d)). This is precisely what happens. We observe first that the AR model needed to double the sampling time and use model orders as high as 30 to represent case (d). The confidence level came down from +7 to +4, but the estimate of the decay ratio is still acceptable. Summarizing, the algorithm was smart enough to realize that this cases were more difficult to solve and it adjusted itself to obtain the best possible estimate of the solution.

Table 5.3 (cases (e) through (h)) show how the algorithm behaves under difficult conditions. Cases (e) through (g) highlight the problem of the low-frequency resonance. In this problem, a resonance exists at a frequency lower than that of the reactivity-to-power transfer function. The pole causing this resonance, having lower frequency, has a smaller real part and thus dominates the asymptotic behavior of the response, but the decay ratio of the pole of interest is larger. The low-frequency pole has a real part of -0.02 Hz and a decay ratio of 0.28.

The pole in case (e) has a decay ratio of 0.5. In this case we observe that the apparent decay ratio of the autocorrelation is very poor, the asymptotic decay ratio of the impulse response, however, yields the decay ratio of the low-frequency pole, due to its dominance of the asymptotic behavior. Fortunately, the frequency domain search is able to perform properly for this case and yield the correct answer which is the best estimate value taken by the algorithm. However, given the disparity between the results, the confidence level is marginal at a value of +3.

In case (f) the problem is better defined. The transfer function pole has now a smaller real part than the low-frequency pole, because the reactor decay ratio is now 0.9. In this case we observe that both the impulse response and the pole search methods work properly. The confidence is moderate, +4, because the sampling time had to be doubled twice to allow for the low-frequency modeling.

Case (g) represents the opposite case. Here, the low-frequency pole has a smaller real part and a larger decay ratio, because the reactor decay ratio is only 0.1. This is an extremely difficult problem to solve, but the pole search part of the algorithm is still able to locate the reactor pole. The estimate is of the order of magnitude, but it is not as accurate as when the decay ratios are high. Nevertheless, we should consider a success that the algorithm is capable of finding a solution even in a difficult case like this. Of course, the confidence level is low, +2, suggesting to the user that this reactor condition is a difficult one for a decay ratio estimation.

Finally, in case (h) we present a case in which no pole exists. This would correspond to a case in which the operator made a mistake, and the signal supplied was not from a BWR neutron detector, or a case in which a recording or processing problem of some sort had happened. We observe that the algorithm detects this problem inmediately and returns a confidence level of -7.

In summary, we can say that the algorithm works extremely well. It not only supplies the asymptotic decay ratio accurately, but it also is able to determine abnormal data sets and use the most appropriate method for the particular condition. We have found the confidence level estimate to be of great usefulness, especially when an on-line or automated diagnostics system has to be implemented.

5.5.2 Application of the Algoritm to BWR Data

During the Peach Bottom stability tests,[53-56] noise
(unperturbed) data was recorded in digital form immediately after or
before the pseudorandom binary signal (PRBS) perturbation tests. Some
of this noise data was made available to the Oak Ridge National
Laboratory by the Electric Power Research Institute (EPRI). The typi-
cal length of the noise recordings was 25 minutes; however, we
received only about 3 minutes worth and only for test 3PT3. We found
that this short time was inadequate to accurately define the decay
ratio (DR), at least for these relatively stable conditions
(DR < 0.5). Nevertheless, we present here the results of this analy-
sis to show that, even with these short record lengths, a "reasonable"
estimate can be obtained. We also received from EPRI the PRBS data
corresponding to the same 3PT3 test. In this case we had a full
25 minute recording. Even though this was not noise data, we treated
it as such, because the input perturbation is random in nature.
We estimated the "output" stability as if we did not know the input,
which is the case in the noise recordings. We show that we can obtain
a good estimate from this type of data without making use of the input
information.

The estimated DR from the noise recordings (3 minutes) was
DR = 0.39. The estimated DR from the PRBS data treated as noise was
DR = 0.53, which compares well with the published DR results using the
standard transfer function technique, DR = 0.50.[56] Table 5.4 and
figures 5.8 and 5.9 summarize the results of these analyses.

Table 5.4

RESULTS OF STABILITY ANALYSIS OF PEACH BOTTOM
TEST 3PT3 DATA

	Asymptotic Decay Ratio	Natural Frequency (Hz)
Noise data (3 minutes)	0.39	0.38
PRBS data (25 minutes)	0.53	0.38
Transfer function results*	0.50	0.41

*from Reference 56

A different set of data was also available to validate this technique. This data was obtained during the local stability tests performed at the Dresden-2 reactor.[59,105] For these tests, the reactor was operated at the intersection between the minimum recirculation pump speed and the 100% flow control line, which is the most unstable point of the normal operating range. This corresponded to 52.7% power and 38.9% flow. The decay ratio estimated for this condition was 0.35, whereas the decay ratio at full power was 0.15. Unfortunately, this number cannot be compared to transfer function measurements, simply because they were not performed. However, computer calculations, both by Exxon Nuclear Corporation and Oak Ridge National Laboratory predicted very similar decay ratios for both conditions.[105,110]

In summary, the present technique works properly with both computer-generated and real-world data.
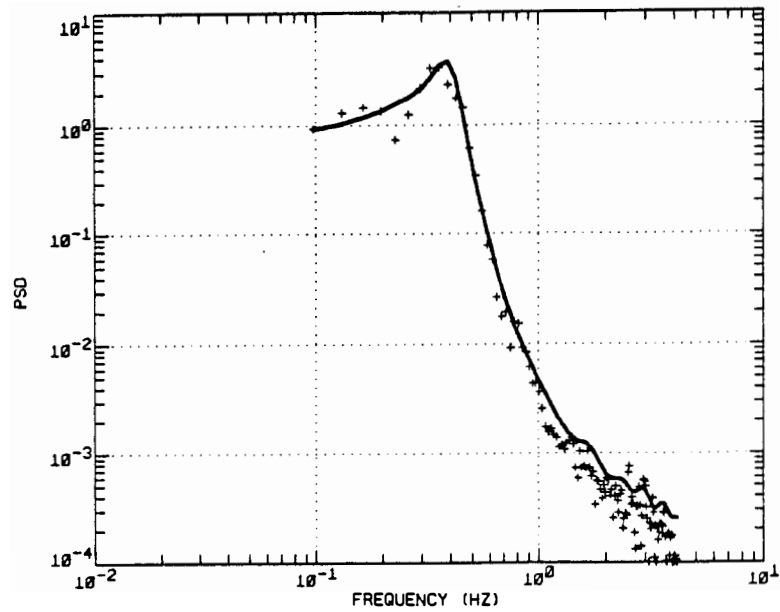
Figure 5.8   Comparison between measured and AR-model predicted
power spectral density for noise data.
(Peach Bottom, test 3PT3).
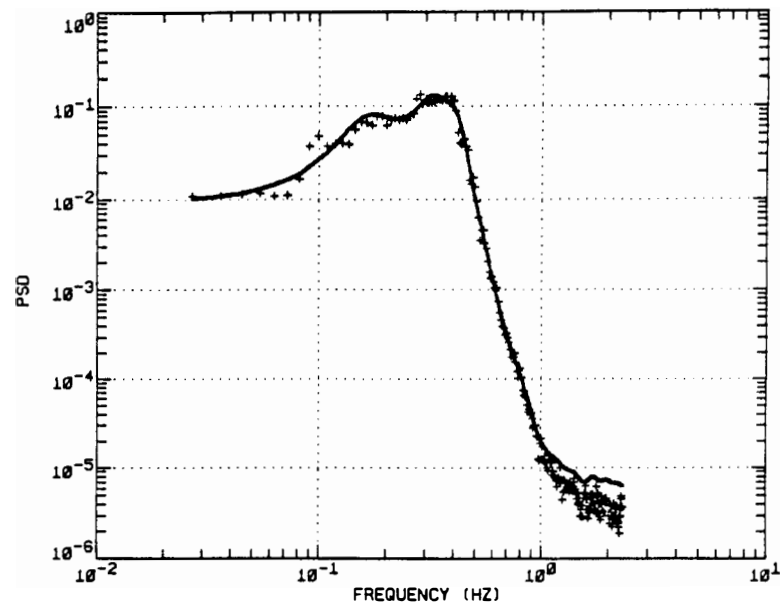
Figure 5.9   Comparison between measured and AR-model predicted
power spectral density for PRBS data treated as noise.
(Peach Bottom, test 3PT3).

CHAPTER 6

NONLINEAR SYSTEM DYNAMICS

The basis of linear system dynamics is the superposition principle, which states that if a system is linear the response is proportional to the input. In the same way, the response to the sum of two different inputs is equal to the sum of the responses to both inputs. This is the basis for the transfer function analysis used in linear dynamics because, as shown in Chapter 3, a transfer function completely defines the dynamics of a linear system. In nonlinear dynamics, however, the superposition principle does not apply and the response of a system to different inputs cannot be characterized by a single transfer function.

One characteristic of nonlinear systems is that they are able to excite frequencies in the output response which are not contained in the input signal. Normally, the frequencies excited are harmonics of the input frequencies, but this is not necessarily the case for all nonlinear systems. In particular, some systems could excite subharmonics (lower frequencies) of the input frequencies. A typical example of nonlinearity is a saturation. For instance, an electronic amplifier will perform linearly as long as the output signal is small, but if the output approaches the voltage of the power supply, the output signal becomes deformed. In this fashion, the output to a sine wave is chopped at the tops and it looks more like a square wave,

which contains higher frequencies (harmonics). The new frequencies
are harmonics of the original because the output signal must be
periodic with period equal to the one of the input signal. This
example shows one of the main characteristics of non-linear systems,
which is that they behave in a linear fashion as long as the output is
of small magnitude, and nonlinearities become important as the magnitude increases.

A nonlinear effect of particular importance is the appearance
of limit cycles. Within linear theory (see Chapter 3), if one of the
poles has a positive real part, the system becomes unstable and the
solution is a diverging exponential whose magnitude will grow indefinitely. However, many nonlinear systems under these conditions exhibit
periodic bounded solutions, which are called limit cycles. Limit
cycles are commonly represented as trajectories in phase space (the
space of the dependent variables) with time being a parameter of the
trajectory.

A typical example of the development of a limit cycle is
presented in Figure 6.1. The point (0,0) is an unstable equilibrium
point. Close to this point (see the insert in Figure 6.1) the trajectory behaves linearly and spirals away exponentially. However, if we
look at the whole picture, we observe that the trajectory stays
bounded and it eventually converges to a closed curve which defines
the limit cycle.
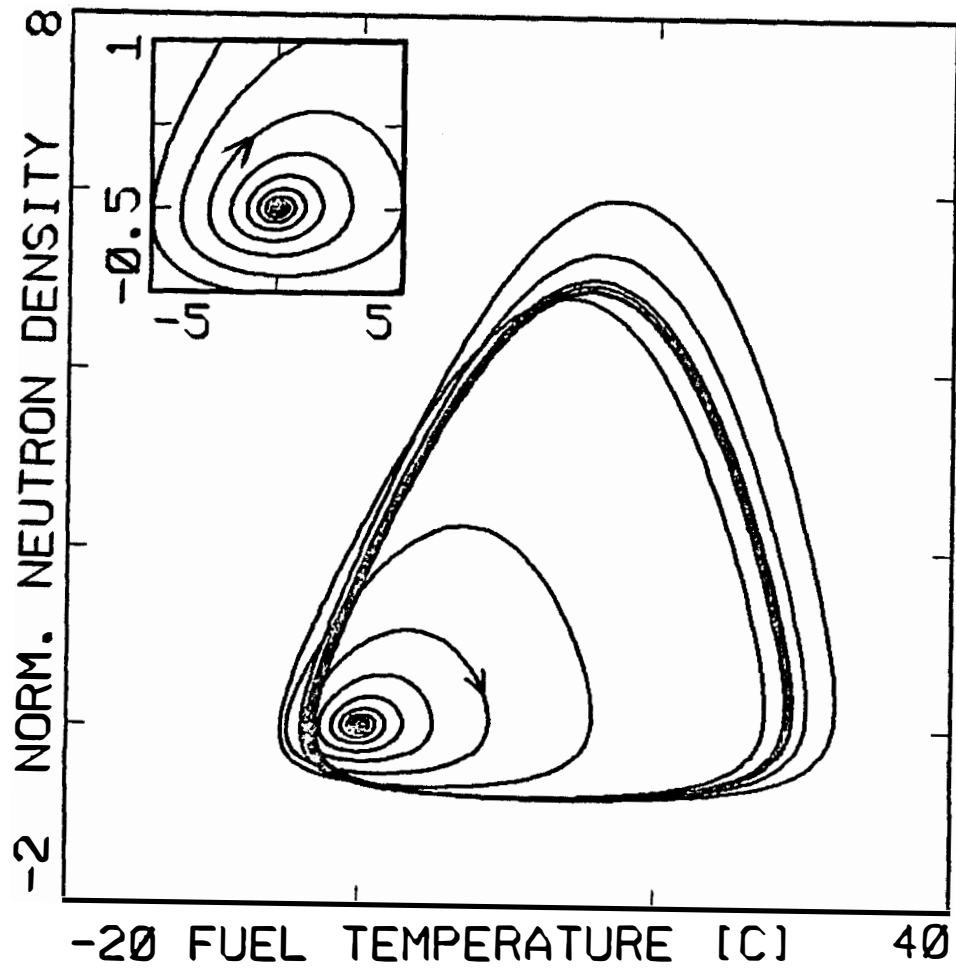
Figure 6.1 Development of a typical limit cycle in phase space.

The stability of linear systems is characterized by an asymptotic decay ratio (see Chapter 3). In nonlinear systems exhibiting limit cycles, however, the decay ratio is always equal to 1.0. A parameter of more interest in the nonlinear regime, then, is the amplitude of the oscillations.

6.1 Subharmonic Excitation: Feigenbaum's Theory for
the Transition to Aperiodicity

As stated before, the most common effect of nonlinearities is the excitation of higher harmonics of the input frequencies. However, there are some systems in which the nonlinearities excite lower harmonics (i.e., subharmonics) also. In some of these systems the excitation of subharmonics eventually produces aperiodic or chaotic behavior as a system parameter is changed. In these systems the solution is stable and converges to the equilibrium point as long as the value of a parameter, p, is less than a critical value, p_0. When p_0 is surpassed, the solution becomes unstable and a limit cycle of period T is reached. When p is made greater than p_1, a new critical value, the limit cycle becomes unstable and a new limit cycle of twice the period appears (i.e., the first subharmonic is excited). Further increases of the value of the parameter result in a cascade of instabilities with each successive limit cycle being unstable at a value of the parameter p_j. The successive periods are $2^j T$ and, thus, the jth subharmonic is excited. Each of the successive limit cycle instabilities is called a period-doubling pitchfork bifurcation.

Numerical experiments have shown that the critical values, p_j, at which the bifurcations occur converge geometrically to an accumulation point. In other words, the bifurcations occur closer and closer as j is increased. Furthermore, the convergence ratio, defined as

$$\delta_j = \frac{p_j - p_{j-1}}{p_{j+1} - p_j} \tag{6-1}$$

converges to a universal constant, $\delta = 4.6692...$, which was predicted by Feigenbaum's theory[82,83] and which has the same universal value regardless of the form or order of the system of equations. A second Feigenbaum constant, $\alpha = 2.5029...$, is called the pitchfork scaling parameter and relates the relative magnitudes of the new subharmonic frequencies.

Feigenbaum's theory was developed for one-dimensional noninvertible maps and then expanded to nonlinear differential equations based on properties of the Poincare map.[80]

A one-dimensional map is simply a function, F, or a recursion formula which relates the value of the dependent value, x_k, to itself one increment of time before. That is

$$x_k = F(x_{k-1}) . \tag{6-2}$$

For instance, the relation

$$x_k = 4b \ x_{k-1} \ (1-x_{k-1}) \tag{6-3}$$

is a one-dimensional map which is graphically represented in Figure 6.2a. This map is clearly noninvertible because there are two possible values of x_{k-1} which give the same x_k.
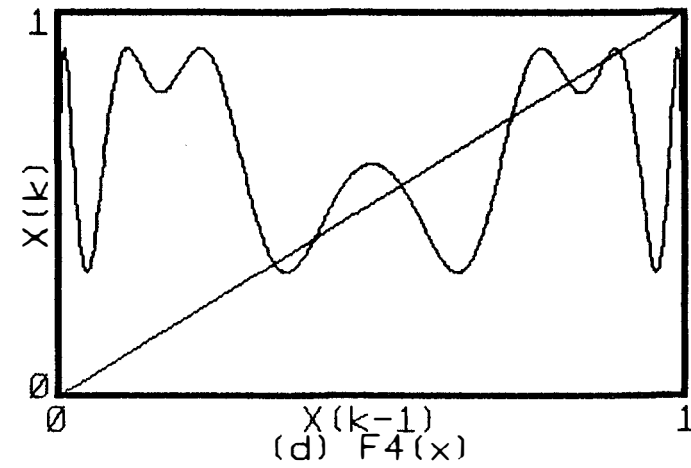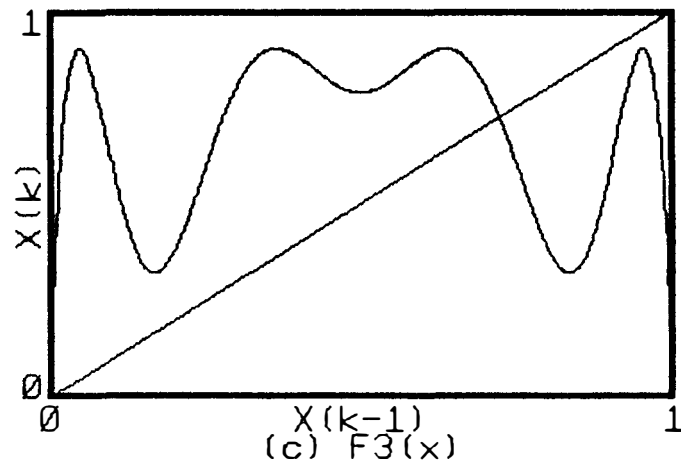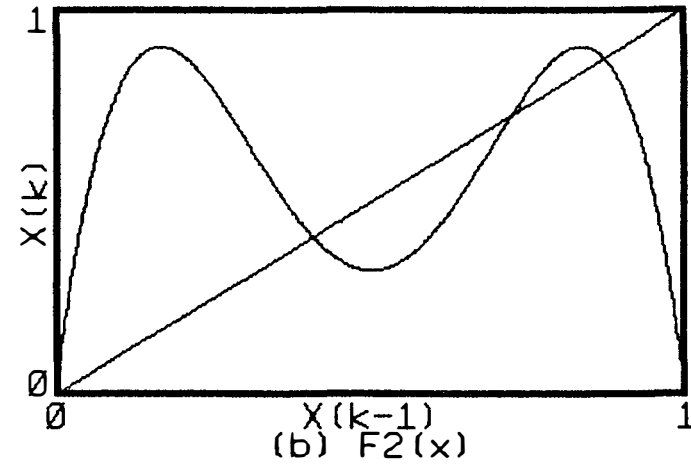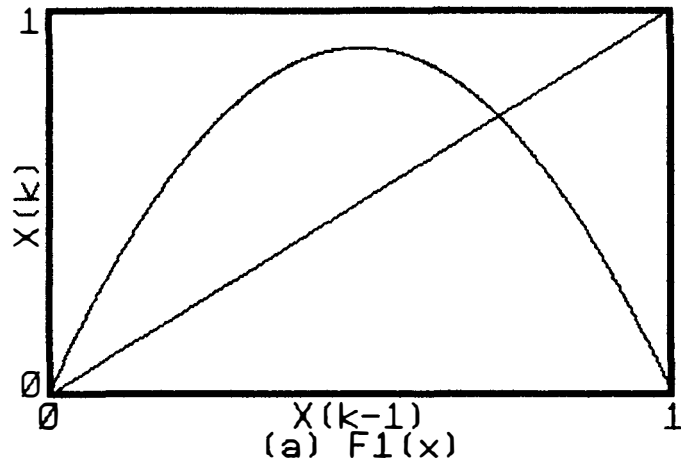
Figure 6.2 Graphical representation of the map $x_k = 4B \, x_{k-1} \, (1-x_{k-1})$
for the first four iterations.

The equilibrium points are found by the intersection of the map with the line $x_k = x_{k-1}$. These points are stable if the absolute value of the derivative of the map is less than 1.0. Otherwise they are unstable. By changing a parameter (for instance b of Equation (6-3)) we can make the equilibrium point unstable. Feigenbaum[80-83] shows that for maps with quadratic extrema (i.e., F and its derivative are continuous), whenever the equilibrium point of $F(x)$ becomes unstable, two new equilibrium points appear in the second iterate of F ($F_2(x)=F(F(x))$) and, thus, the solution of the map bifurcates (splits) into two points (Figure 6.2b). If we keep increasing the parameter b, then $F_2(x)$ becomes unstable, but four equilibrium points of $F_4(x)$ appear (Figure 6.2d). Thus $F_2(x)$ bifurcates and the process continues ad infinitum when $F_4(x)$ bifurcates and then $F_8(x)$ and so on. Note that there is only one equilibrium point for $F_3(x)$ and it is unstable.

A direct result of Feigenbaum's theory is that the values of b, at which the bifurcation occurs, converge to the accumulation point b_∞ in a geometric fashion, with a convergence rate $\delta = 4.6692...$, independently of the form of the map. For values of b greater than b_∞, the solution has bifurcated an infinite number of times; therefore, its period is infinite. That is, the solution is aperiodic.

The process through which the aperiodic or chaotic behavior is reached is called in the literature a "cascade of period-doubling pitchfork bifurcations." This process is represented in Figure 6.3
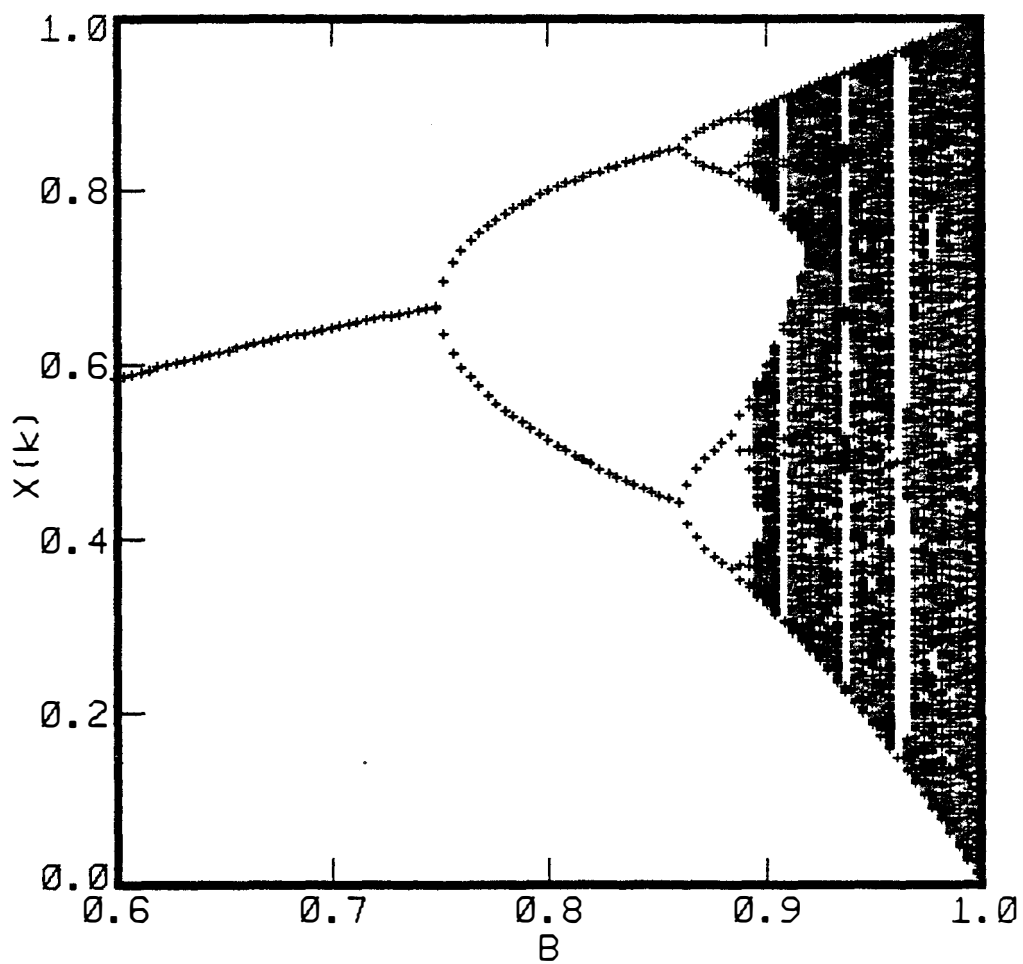
Figure 6.3 Bifurcation diagram for the map $x_k = 4B\ x_{k-1}\ (1-x_{k-1})$.

which shows the bifurcation diagram for the map of Equation (6-3).
In this diagram the equilibrium points of Equation (6-3) are plotted
as a function of the value of the parameter b. We observe that when b
< 0.75 the single equilibrium point is stable. For b = 0.75 the first
bifurcation occurs and two points appear. From then on the cascade of
bifurcations starts. The geometric convergence of the critical bifur-
cation values is also obvious in this figure.

A second universal constant defined by Feigenbaum is the
pitchfork scaling parameter, α_j, which scales the magnitude of the
successive bifurcations as shown in Figure 6.4. The value of α_j
converges to $\alpha = 2.5029...$ as the bifurcation number increases.

Some nonlinear differential equations exhibit a similar
universal behavior as the one described for 1-D noninvertible maps.
This effect is explained in the literature using Poincare maps.[80]
A Poincare map corresponding to a particular solution of a system of
equations can be constructed by intersecting the phase space trajec-
tory of the solution with a predetermined surface. In this way a set
of points is obtained every time the trajectory intercepts the surface
in a predetermined direction. The 1-D map can then be formed graphi-
cally by plotting the consecutive points obtained in this manner.
If this map has a quadratic maximum, then the Feigenbaum scenario is
applicable. In this event, the set of nonlinear differential
equations associated with such a Poincare map is expected to follow
this scenario.
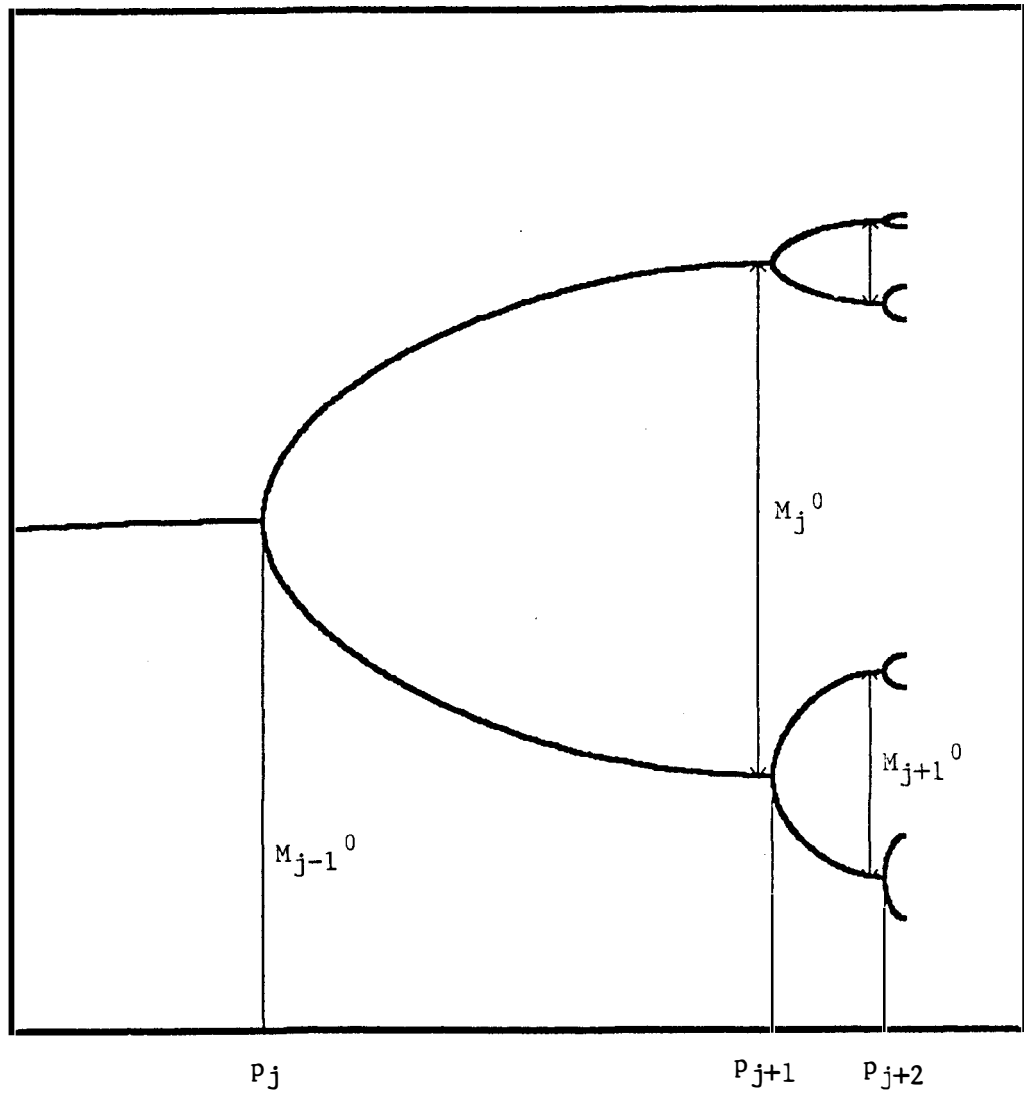
Figure 6.4 Typical bifurcation diagram showing the scaling process.

6.2 Period-Doubling Pitchfork Bifurcations in

Nonlinear Ordinary Differential Equations

In the previous section the transition to aperiodic behavior through a cascade of period-doubling bifurcations, which is experienced by some nonlinear ordinary differential equations, was explained a posteriori in terms of Poincare maps. That is, once the solution is found to undergo bifurcations numerically, the Poincare map is formed and the behavior explained by the non-inversibility of the map. In this section, however, we develop a procedure to gain some insight to the necessary conditions which have to be satisfied by a set of N coupled ordinary differential equations in order to exhibit period-doubling bifurcations.

Upon elimination of N-1 dependent variables, the resulting N^{th} order differential equation can be written in general form as

$$\sum_{n=0}^{N} B_n(x,t,p) \frac{d^n x(t)}{dt^n} = F(t,p) \quad , \qquad (6-4)$$

where B_n are nonlinear coefficients which in general depend on the value of the dependent variable, x, the independent variable, t, and a parameter, p, that can be varied externally. $F(t,p)$ is a forcing function. Arbitrarily, and without loss of generality, we set B_N to unity.

Let us assume that for a value of the parameter $p = p_1$ greater than p_0, the critical value, there is a periodic solution, $x_0(t)$, which defines a limit cycle with period T. We are interested at this

moment in the stability of the limit cycle with respect to changes in the parameter p. Following a small perturbation, we can expect that the new solution, $x(t)$, will be similar in form to $x_0(t)$ but with different magnitude. Let us thus make a change of variables such that

$$x(t) = (1+M(t)) \, x_0(t) \quad . \tag{6-5}$$

The new variable, $M(t)$, describes as a function of time the change in magnitude of the original limit cycle, $x_0(t)$, when the parameter is changed. Note that from Equation (6-5)

$$\frac{d^n x(t)}{dt^n} = \sum_{k=0}^{N} \binom{n}{k} \frac{d^k (1+M(t))}{dt^k} \frac{d^{(n-k)} x_0(t)}{dt^{(n-k)}} \quad . \tag{6-6}$$

We can introduce Equation (6-6) into (6-4) and, after some algebraic manipulations, we obtain

$$\sum_{n=0}^{N} B_n(x_0,t,p_1) \sum_{k=0}^{N} \binom{n}{k} \frac{d^k M(t)}{dt^k} \frac{d^{(n-k)} x_0(t)}{dt^{(n-k)}}$$

$$+ \sum_{n=0}^{N} B_n(x,t,p) \frac{d^n x_0(t)}{dt^n} = F(t,p) \quad . \tag{6-7}$$

Since $x_0(t)$ is a solution of the original equation for $p = p_1$, the following relation is satisfied

$$\sum_{n=0}^{N} B_n(x_0,t,p_1) \frac{d^n x_0(t)}{dt^n} = F(t,p_1) \quad . \tag{6-8}$$

Therefore, Equation (6-7) becomes

$$\sum_{k=0}^{N} \left[\sum_{n=0}^{N} \binom{n}{k} B_n(x,t,p) \frac{d^{(n-k)} x_0(t)}{dt^{(n-k)}} \right] \frac{d^k M(t)}{dt^k}$$

$$= - \sum_{n=0}^{N} \Delta B_n(M,t,p) \frac{d^n x_0(t)}{dt^n} \quad , \tag{6-9}$$

where the term ΔB_n is defined as

$$\Delta B_n(M,t,p) = B_n(x,t,p) - B_n(x_0,t,p_1) \qquad (6-10)$$

and, thus, it contains terms which depend on M and independent terms. Let us collect all the independent terms of ΔB_n in a variable called $I_n(t,p)$ and all M-dependent terms in another variable called $D_n(M,t,p)$. Then Equation (6-9) becomes

$$\sum_{k=0}^{N} C_k(M,t,p) \frac{d^k M(t)}{dt^k} = -\sum_{n=0}^{N} I_n(t,p) \frac{d^n x_0(t)}{dt^n} , \qquad (6-11)$$

where

$$C_k(M,t,p) = \sum_{n=0}^{N} \binom{n}{k} B_n(x,t,p) \frac{d^{(n-k)} x_0(t)}{d^{(n-k)}} \quad ; \ k=1,\ldots,N \quad (6-12)$$

and

$$C_0(M,t,p) = \sum_{n=0}^{N} \left[B_n(x,t,p) + \frac{D_n(x,t,p)}{M(t)} \right] \frac{d^n x_0(t)}{dt^n} . \qquad (6-13)$$

Note that Equation (6-11) is formally very similar to the original equation, (6-4). The only difference is that C_N is not equal to 1.0, since from equation (6-12), and for $k = N$

$$C_N(M,t,p) = C_N(t) = x_0(t) \qquad (6-14)$$

and, thus, C_N is a periodic function of period T. Furthermore, it does not depend on M or p.

Let us define a new set of coefficients, B_n^1, as

$$B_n^1(M,t,p) = \frac{C_n}{C_N}$$

$$= \sum_{k=n}^{N} \left[\binom{n}{k} B_n(x,t,p) + \frac{\delta_{k,0}}{M(t)} D_k(M,t,p) \right] \frac{1}{x_0(t)} \frac{d^k x_0(t)}{dt^k} \qquad (6-15)$$

$$F_1(t,p) = -\sum_{n=0}^{N} I_n(t,p) \frac{1}{x_0(t)} \frac{d^k x_0(t)}{dt^k} \quad . \tag{6-16}$$

Note that all the coefficients are independent of the amplitude of $x_0(t)$. The equation for the limit cycle amplitude becomes

$$\sum_{n=0}^{N} B_n^{\ 1}(M_1,t,p) \frac{d^n M_1(t)}{dt^n} = F_1(t,p) \quad , \tag{6-17}$$

where we have renamed $M(t)$ as $M_1(t)$.

This equation is formally equal to Equation (6-4), the original equation. We observe, however, that the effect of the original forcing function, F, has been completely lost because it was canceled by the relation in Equation (6-8). All of the new coefficients, $B_n^{\ 1}$, and forcing functions are now function only of $x_0(t)$ and the original coefficients, B_n. The form of the new coefficients is a ratio of periodic functions (see Equation (6-15)) and, thus, they are periodic functions. The period, however, is not necessarily the same as the one of the original limit cycle, $x_0(t)$. For instance, recall that

$$\tan(a/2) = \frac{1-\cos(a)}{\sin(a)} \quad ; \tag{6-18}$$

consequently, given the right conditions, the new coefficients, $B_n^{\ 1}$, could have a period twice as large as the original one (i.e., 2T). Note that these conditions will not be met in general for all N^{th} order differential equations. This is the reason why not all such equations bifurcate.

If we consider now only very small perturbations of the parameter p around the original value, p_1, then $M_1(t) \ll 1.0$, and

Equation (6-17) can be linearized. We obtain, then, a linear equation with periodic coefficients. This is a equation of the Floquet type,[98] which has been amply studied. Floquet theorem says that the solution, $M_1(t)$, will be periodic of the same period of the coefficients, which is twice as large as the period of the original limit cycle. There is a set of Floquet exponents which determine the stability of the solution, that is, the exponents determine whether $M_1(t)$ will follow a decaying or a diverging oscillation.

The oscillatory part of $M_1(t)$ represents a transient between two limit cycles of x(t) for parameters p_1 and $p_1 + \Delta p$. As long as $M_1(t)$ is stable, following a perturbation in p there will be a transient but x(t) will settle to a new limit cycle of different magnitude with the same period as the previous one. Note that Equation (6-17) is nonlinear and, thus, if $M_1(t)$ became unstable for a particular value of the parameter p, and if the right conditions exist, $M_1(t)$ could reach a limit cycle of its own, $M_1^0(t)$. For this value of the parameter the solution of the original equation is

$$x_1(t) = (1+M_1^0(t)) \, x_0(t) \quad , \qquad (6-19)$$

where $M_1^0(t)$ has period 2T. Therefore, the new solution, $x_1(t)$, has period 2T. Under these conditions, the equation has produced subharmonics of the original frequency in the solution. This process is also called a period-doubling bifurcation.

Equation (6-17) is exactly equal in form to the original Equation (6-4). Therefore, the same procedure used for x(t) can be

applied to $M_1(t)$. Assuming that $M_1(t)$ becomes unstable and reaches a limit cycle (periodic solution of period 2T) for a parameter value $p = p_2$, we make the change of variables

$$M_1(t) = (1+M_2(t)) M_1{}^0(t) \quad . \tag{6-20}$$

Following the same procedure outlined at the beginning of this section, we obtain

$$\sum_{n=0}^{N} B_n{}^2(M_2,t,p) \frac{d^n M_2(t)}{dt^n} = F_2(t,p) \quad . \tag{6-21}$$

with the coefficients, $B_n{}^2$, and the forcing function, F_2, defined by equations similar to Equations (6-12) through (6-16).

Again, C_N is a periodic function, but now its period is 2T (the period of $M_1{}^0(t)$). Thus, if the right conditions are met, the new coefficients, $B_n{}^2$ could have period 4T and, consequently, $M_1(t)$ would bifurcate. The period of the original function, $x(t)$, for this value of the parameter is, therefore, 4T.

The same procedure can now be applied to $M_2(t)$ and then to $M_3(t)$. In general, the equation representing the magnitude at the jth bifurcation is

$$\sum_{n=0}^{N} B_n{}^j(M_j,t,p) \frac{d^n M_j(t)}{dt^n} = F_j(t,p) \quad , \tag{6-22}$$

with

$$B_n{}^j(M_j,t,p) = \sum_{k=n}^{N} \left[\binom{n}{k} B_k(M_{j-1},t,p) \right.$$

$$+ \frac{\delta_{k,0}}{M_{j-1}(t)} \left. D_k{}^j(M_{j-1},t,p) \right] \frac{1}{M_{j-1}(t)} \frac{d^k M_{j-1}(t)}{dt^k} \tag{6-23}$$

and

$$F_j(t,p) = -\sum_{n=0}^{N} I_n^{\ j}(t,p) \frac{1}{M_{j-1}(t)} \frac{d^k M_{j-1}(t)}{dt^k} \quad . \qquad (6\text{-}24)$$

Note that given the appropriate conditions $M_j^{\ 0}(t)$ is a periodic

solution of period $2^j T$ and the original variable, $x(t)$, which is given

by the expression

$$x(t) = (1+M_j(t))(1+M_{j-1}^{\ 0}(t))\cdots(1+M_1^{\ 0}(t))x_0(t) \quad , \qquad (6\text{-}25)$$

is also periodic of period $2^j T$.

Equations (6-22) through (6-24) define a recurrence relation

for the magnitude of the successive bifurcations. Note that the

amplitude of $M_j(t)$ is equivalent to the magnitude of the bifurcation

in the bifurcation diagram as shown in Figure 6.4. It has been

observed numerically that these magnitudes scale down in successive

bifurcations according to the universal parameter α. Thus, the ampli-

tude of the successive $M_j(t)$ should scale accordingly.

The fact that the successive $M_j(t)$ and p_j scale in a universal

fashion is very hard to prove from first principles; however, the

recursive nature of these equations suggests that such universality is

possible because, once many bifurcations have taken place, the infor-

mation from the original equation has already been lost and the only

important feature is the way in which the recursion is produced.

In summary, in this section we have studied some necessary

conditions for period-doubling bifurcations in ordinary differential

equations. We showed that

(a) general nonlinear ordinary differential equations can sustain period-doubling pitchfork bifurcations of the original limit cycle oscillations.

(b) the underlying cause of these bifurcations is an instability of the magnitude, $M(t)$, of the limit cycle oscillations.

(c) $M(t)$ is a periodic function and may have a period twice as large as the original because the coefficients of its differential equation are formed by ratios of periodic functions.

(d) some generic arguments have been made supporting the fact that these equations may have universal scaling factors.

CHAPTER 7

A REDUCED ORDER NONLINEAR MODEL

Integrating a set of nonlinear differential equations is a difficult task. The solution of at least one nonlinear system of equations is involved at each time step. In addition, the stronger the nonlinearity, the longer the computation time to solve the problem. This is the reason we are interested in obtaining an approximate solution based on a low-order model with as few equations as possible. An additional advantage is that, having few equations, the results can be studied in more depth, and a physical understanding can be extracted from the numerical solution. In fact, some analytical work can be performed to increase the understanding of this solution.

The approach taken in this section is to solve numerically a low-order, nonlinear model of a boiling water reactor (BWR) and study the qualitative aspects of the solutions. A model that gives more accurate numerical solutions is described in Chapter 8.

7.1   The Reduced-Order Model

For the reasons expressed above, we want to use a model with the lowest possible order (number of equations) which would represent qualitatively the dynamic behavior of a BWR. We showed in Chapter 4 that the basic processes involved in BWR dynamics can be represented

by one-point reactor kinetics, a one-node representation of the heat

transfer process in the fuel, and a two-node representation of the

channel thermal-hydraulics to account for the void reactivity feed-

back. Based on these findings, we propose to use the following

reduced order model:

$$\frac{dn(t)}{dt} = \frac{\rho(t)-\beta}{\Lambda} \ n(t) \ + \ \lambda c \ + \ \frac{\rho}{\Lambda} \tag{7-1}$$

$$\frac{dc(t)}{dt} = \frac{\beta}{\Lambda} \ n(t) \ + \ \lambda c \tag{7-2}$$

$$\frac{dT(t)}{dt} = a_1 \ n(t) \ - \ a_2 \ T(t) \tag{7-3}$$

$$\frac{d^2\rho_\alpha(t)}{dt^2} + a_3 \ \frac{d\rho_\alpha(t)}{dt} + a_4 \ \rho_\alpha = k \ T(t) \tag{7-4}$$

$$\rho(t) = \rho_\alpha(t) + D \ T(t) \quad , \tag{7-5}$$

where $n(t)$ is the excess neutron population normalized to the steady

state neutron population; $c(t)$ is the excess delayed neutron precur-

sors concentration also normalized to the steady state neutron popula-

tion; $T(t)$ is the excess average fuel temperature; and $\rho_\alpha(t)$ is the

excess void reactivity feedback. Note that for this simple model,

the only nonlinear term appears in the neutronic equation through the

parametric feedback produced by the reactivity.

Parameters a_1 through a_4 are obtained from the fitting proce-

dure described in Section 4.2 so that the closed-loop poles and zeros

of this model correspond to the ones of the reactivity-to-power

transfer function of the reactor. Since we are interested in the

nonlinear region above the threshold for linear stability, the

parameters for the base case were obtained from a fit to the LAPUR

calculated transfer function for test 7N in Vermont Yankee,[58] for

which conditions a limit cycle was experimentally observed. The

value of the parameters are presented in Table 7.1.

Table 7.1

MODEL PARAMETERS FOR VERMONT YANKEE TEST 7N

Parameter	Value	Units
a_1	25.04	Ks^{-1}
a_2	0.23	s^{-1}
a_3	2.25	s^{-1}
a_4	6.82	s^{-2}
k_0	-7.50×10^{-4}	$K^{-1}s^{-2}$
D	-2.61×10^{-5}	K^{-1}
β	0.056	
Λ	4.00×10^{-5}	s^{-1}
λ	0.08	s^{-1}

The parameter k, which is directly related to the void reac-

tivity coefficient and the fuel heat transfer coefficient, controls

the gain of the feedback and,thus, defines the linear stability of

this reactor model. The value of k_0 given in table 7.1 is the criti-

cal value at which the model becomes unstable. By artificially

increasing the value of k above k_0 we can make the model unstable and,

under these conditions, we can study its behavior in the nonlinear

region.

An external source is needed to excite the model. The respon-

ses of the neutron signal to a reactivity step and to a step in heat

generation in the fuel are presented in Figures 7.1a and b respec-
tively. The reactivity step produces a very large first peak and then
it converges to the limit cycle. The heat generation step, however,
does not result in an overshoot, but it converges more smoothly to the
same limit cycle than in the reactivity step case. Based on these
observations, we applied the input source to the heat generation in
Equation (7-3) for the deterministic and stochastic analyses.

7.2 Model Qualification

The model parameters presented in Table 7.1 were obtained from
a frequency domain fit to the reactivity-to-power transfer function as
calculated by the linear code LAPUR. Therefore, for this fit, the
present model had to be linearized and then Laplace transformed.
Figure 7.2 presents a comparison between the transfer function for the
conditions of test case 7N of Vermont Yankee[58] calculated by LAPUR and
the transfer function of the linearized version of our simple model.
We proved in Chapter 4 that this model accurately represents the
linear dynamic behavior of the reactor when the appropriate parameters
are used.

Although the present model was proven in Chapter 4 to be based
on sound physical foundations, its validity is limited by the fact
that the thermohydraulic processes have been linearized. In view of
this approximation, the results of this model will be valid only for a
limited range of parameters around the nominal values presented in
Table 7.1 (see Chapter 8 for the results of a more accurate model).
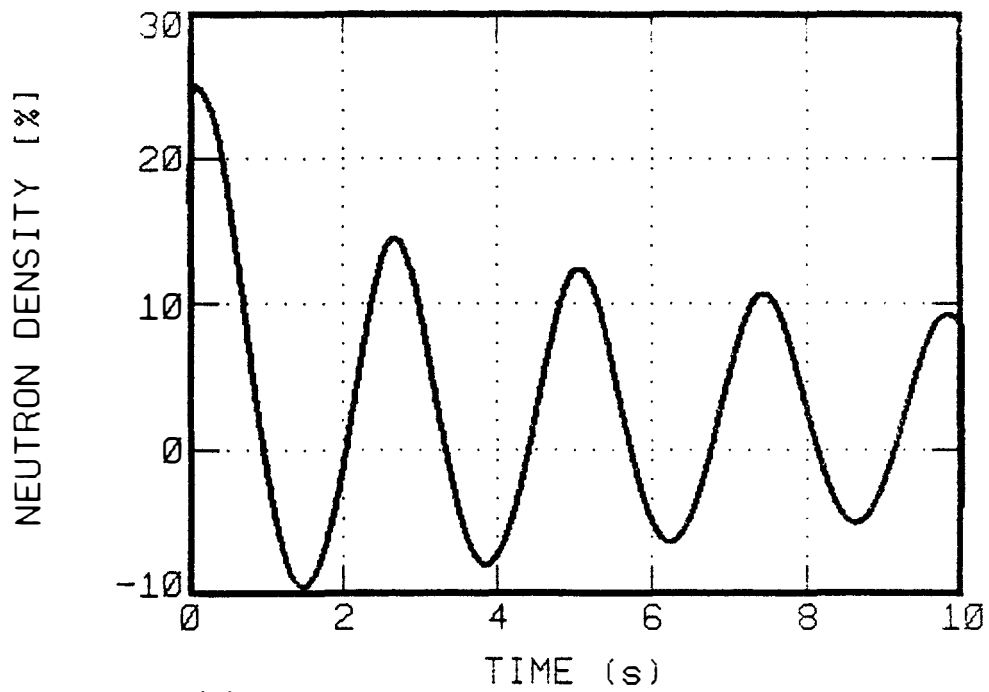
(a) Reactivity step perturbation
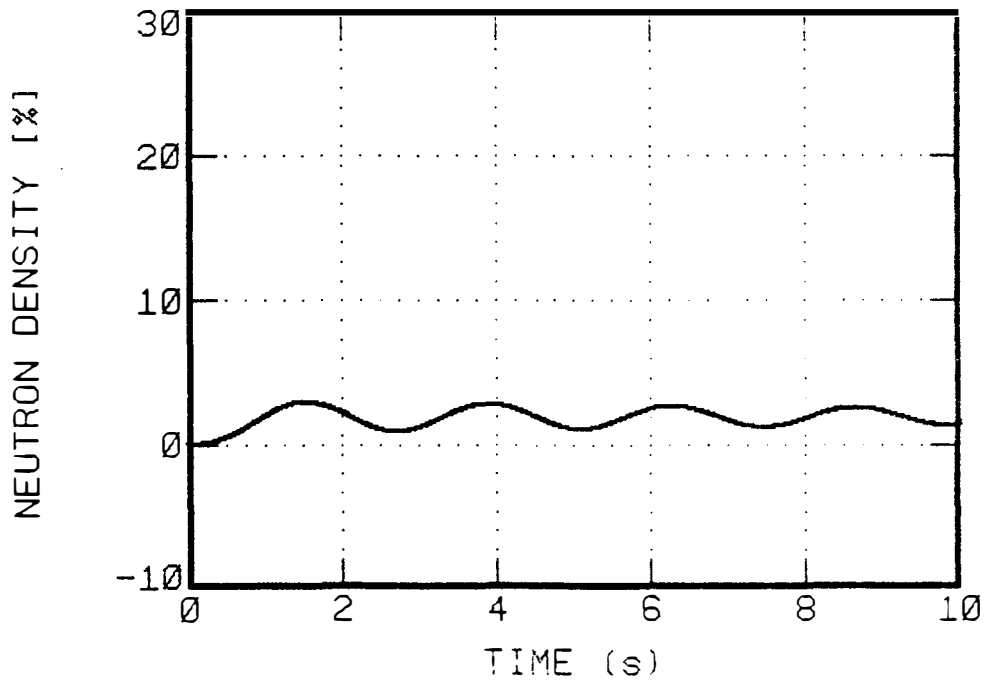
(b) Neutron density step perturbation

Figure 7.1 Step responses of the nonlinear model.
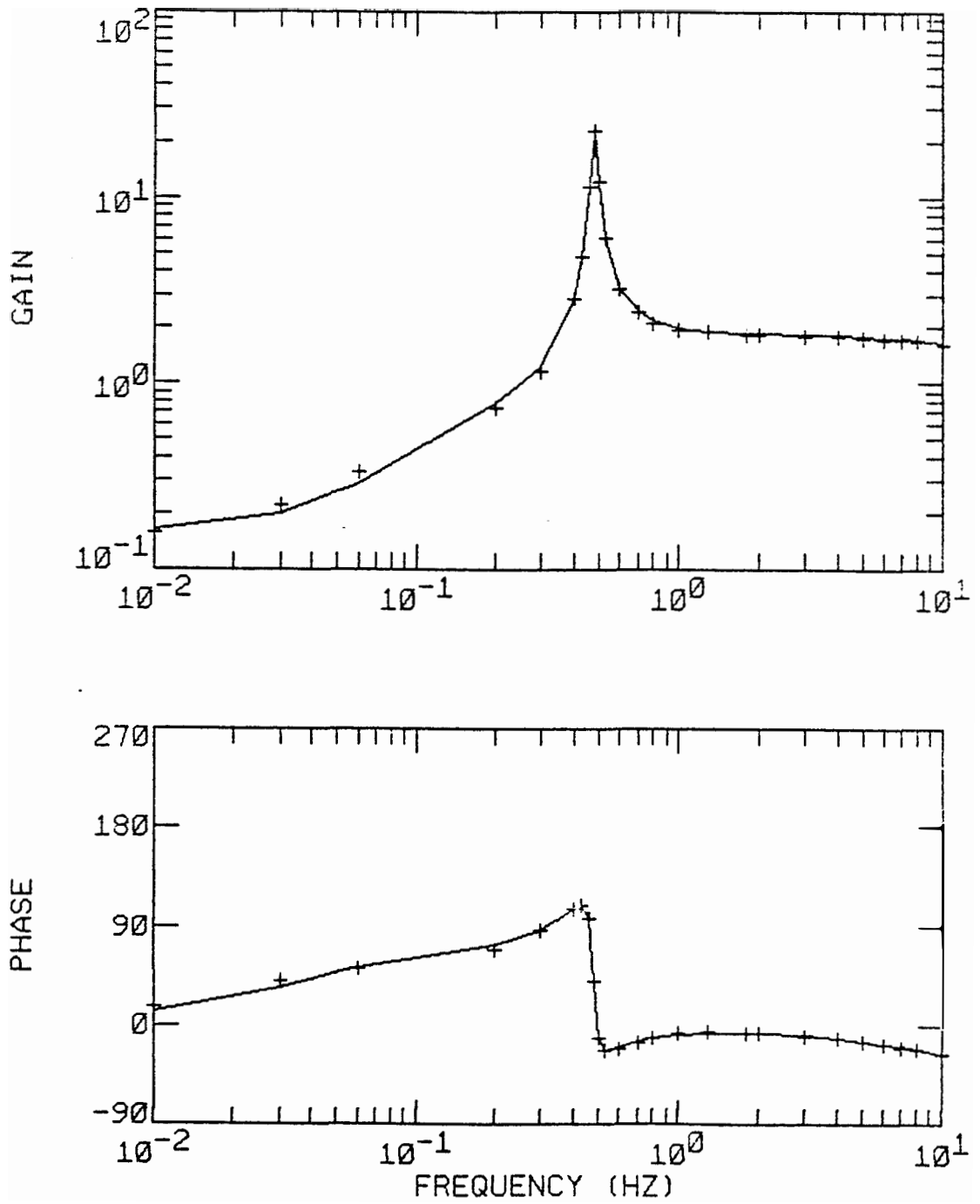
Figure 7.2   Comparison between the LAPUR transfer function and the
             linearized version of the nonlinear model.
             (Vermont Yankee reactor).

Despite this limitation, the model has proven to work remarkably well, for instance it predicted the appearance of the experimetally observed limit cycle.

7.3 Analytical Study

Given the low order of the model, we can perform some analytical studies on it to try to understand the behavior of the solution. The first thing we should be concerned with are the equilibrium points; these are obtained by setting the time derivatives in Equations (7-1) through (7-5) to zero after a step perturbation of magnitude Δ in the neutron population is applied in the heat generation term of Equation (7-3). Some straightforward algebra yields the only two equilibrium points, which we name A and B.

$$A = \left\{ n=-\Delta, \ c=-\beta\Delta/(\lambda\Lambda), \ T=0, \ \rho_\alpha=0 \right\} \tag{7-6}$$

and

$$B = \left\{ n=-1, \ c=-\beta/(\lambda\Lambda), \ T=a_1(\Delta-1)/a_2, \right.$$
$$\left. \rho_\alpha=ka_1(\Delta-1)/(a_2a_4) \right\} \ . \tag{7-7}$$

Equilibrium point A corresponds to the normal reactor operation, while point B describes a shutdown condition. Note that by definition $n = (N-N_0)/N_0$ and, consequently, $n = -1$ corresponds to $N = 0$, where N is the absolute neutron density.

We can now study the local (linear) stability of the system of equations describing the model. For this we linearize the equations around points A and B and compute the eigenvalues of the resulting

Jacobian matrix. By doing this, we find that point B is always

unstable for this type of perturbation and that point A is stable as

long as $k < k_0$. For $k > k_0$ both equilibrium points are unstable.

Let us have a closer look at Equation (7-1). If we neglect

the delayed neutrons effects, it becomes

$$\frac{dn}{dt} = \frac{\rho(n+1)}{\Lambda} \qquad . \qquad (7-8)$$

Therefore, no matter how negative the reactivity feedback, ρ, is,

dn/dt will always tend to zero when n approaches n = -1 due to the

fact that the reactivity feedback is introduced in the form of a

parameter in the equation. This phenomenon is known as parametric

excitation. Physically what this means is that the absolute neutron

population, N, cannot be negative. Therefore, the whole n = -1 line

in phase space, not just the point B, is unstable in the sense that it

repels the trajectories (solutions) if they get close to it.

The n = -1 line can not be crossed by the trajectories.

With the above findings in mind we can draw the phase-space

trajectories of the solutions of the present model. This is done in

the diagrams presented in Figures 7.3a and b, which correspond to

$k < k_0$ and $k > k_0$ respectively. It can be observed that case (a)

corresponds to a stable system. All the trajectories end up at

equilibrium point A; this situation represents normal BWR operation.

Once the linear stability threshold is crossed we have case (b) in

which the trajectory spirals away from eqilibrium point A, which has

become unstable. In both cases, the trajectories are parallel to the
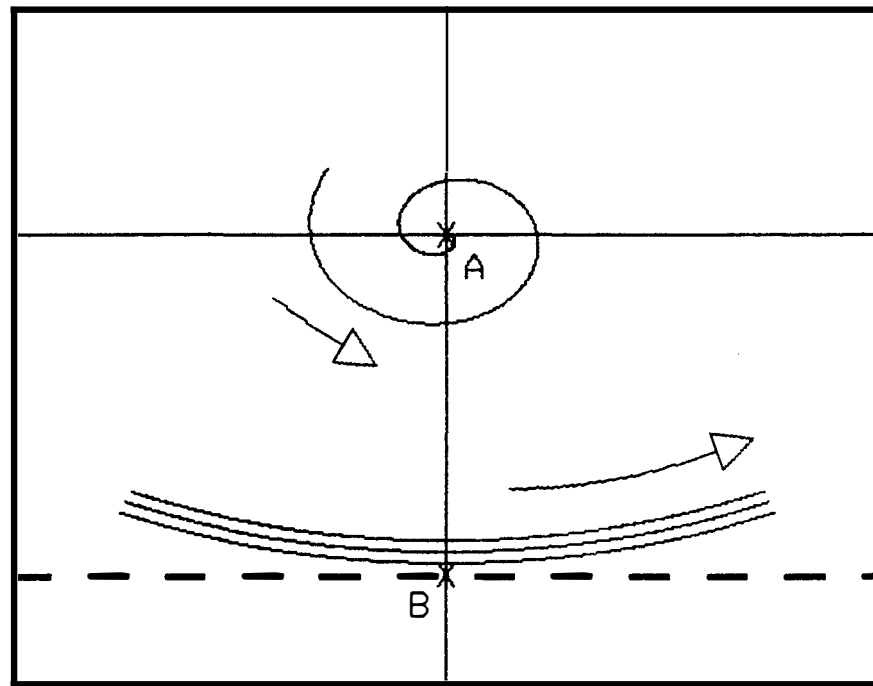
(a) k<k0
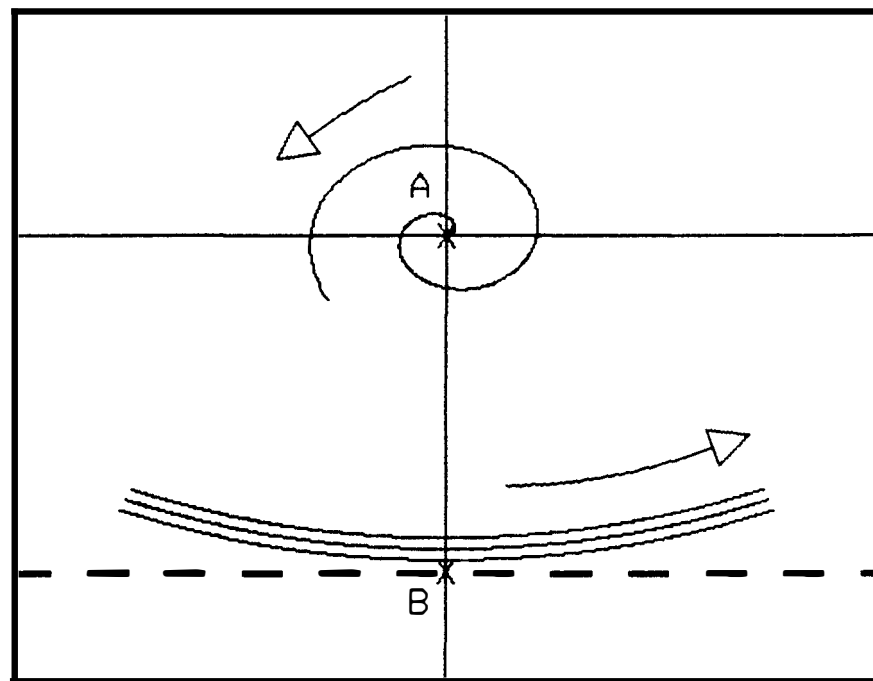
(b) k>k0

Figure 7.3   Phase space trajectory of the solution close to the two
equilibrium points.

n = -1 line as they approach it because they cannot cross it nor end at the unstable equilibrium point B.

Case (b) is in what we have called the non-linear region. We have already studied the local characteristics of the solution close to A and B. Globally, we can have several types of solutions. These types are graphically presented in Figures 7.4a through c. Three cases are considered here: in case (a), the system is globally unstable and the trajectory continually departs from point A. Case (b) describes a situation in which the trajectory departs from the equilibrium point, but due to the nonlinearities the system stabilizes itself and the trajectory remains bounded, eventually converging to a closed line. This line defines a limit cycle, which corresponds to a periodic solution of fixed magnitude. Case (c) is similar to case (b) in the sense that the trajectory is repelled by equilibrium point A and at the same time remained bounded due to the nonlinearities. The difference here is that no periodic solution (closed line) exists; therefore, the trajectory stays bounded within a region, but never converges to a closed curve or to an equilibrium point. This region is called a strange attractor[80] and the solution of a system of equations with a strange attractor is said to be aperiodic.

At this time it is hard to predict analytically which of the three cases shown here represents the behavior of our model in the nonlinear region, or even if all three are valid for different ranges of some parameters. To determine this we will have to resort to numerical solutions of the equations.

Figure 7.4 Possible types of solutions away from the equilibrium points.
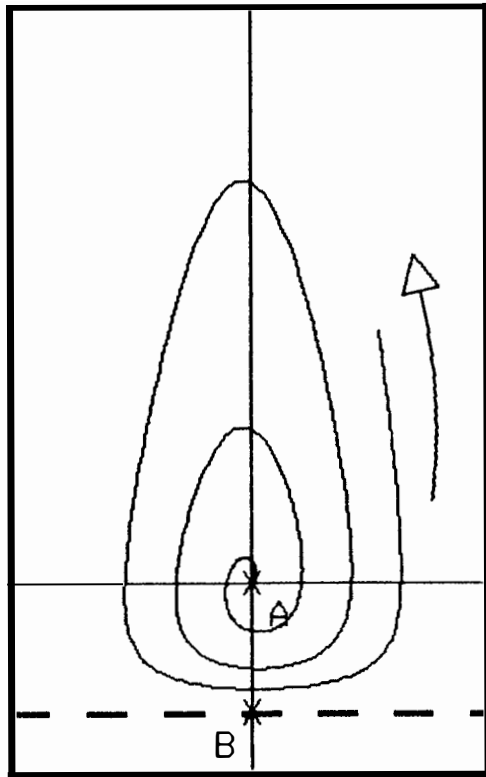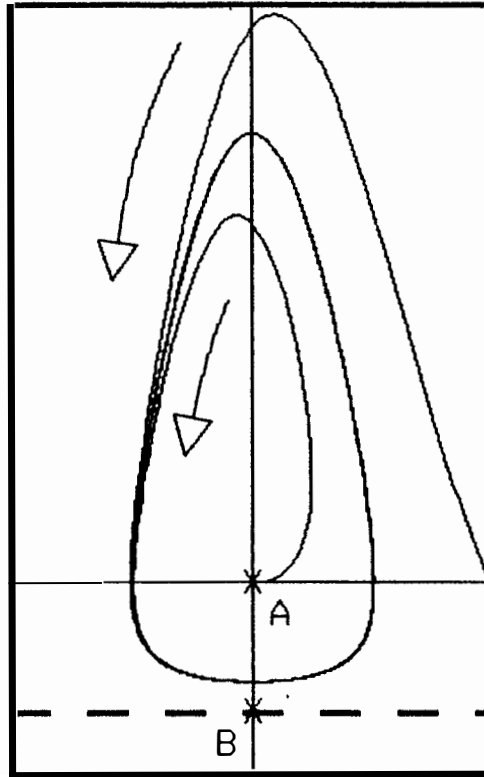
(a) Divergent   (b) Limit Cycle   (c) Strange Attractor

In summary we have seen that the present model has two equilibrium points. One of them, corresponding to a shutdown configuration, is always unstable. The other point becomes unstable when the feedback gain is increased above a critical value. When both equilibrium points are unstable, nonlinearities are important and fully define the dynamic behavior of the reactor. In the nonlinear region limit cycles (periodic solutions) or strange attractors (aperiodic solutions) are possible as a result of the interaction of the two unstable equilibrium points and the phase-space trajectories.

7.4 Deterministic Numerical Analysis

For the deterministic analysis, a step-type perturbation was used. As mentioned before, it was determined that the solution would converge faster and more smoothly to equilibrium if the perturbation were introduced in the heat generation term of the fuel Equation (7-3); therefore, for all these anlysis a 10% perturbation in the neutron population was introduced in the fuel equation at time $t = 0^+$, and then the solution was allowed to converge freely to its final state.

It is of great importance to select the appropriate numerical method for the solution of a nonlinear system of ordinary differential equations. In particular, the stability of the method used should be guaranteed for the specific set of equations to solve. Most ordinary differential equation solvers require that the eigenvalues of the

system lie within a stability region. For instance, the Runge-Kutta methods require that all the eigenvalues lie within a circle of radius $1/\Delta t$ with center in the negative real axis and which passes through the origin. The largest eigenvalue, then, sets the sampling time, Δt; if a problem has eigenvalues of different magnitudes, it is said to be stiff, because the slow eigenvalue solution (often the most interesting) has to be obtained with the fast eigenvalue sampling time, even if the fast transient has already died away. In our particular problem, we will show that the nonlinearities introduce an important higher harmonic contamination. In fact, all the harmonics up to infinite frequency are present. If we used a Runge-Kutta method, we would be forced to use a sampling time of zero to satisfy the stability requirements. Fortunately, there are the so-called implicit methods for ordinary differential equation solutions which are absolutely or A-stable regardless of the time step. These methods are stable regardless of the magnitude of the eigenvalues as long as the real part is negative. The time step in these methods controls the accuracy of the solution but not its stability. We used the A-stable method in the LSODE package[106] which has time step control for the accuracy of the solution. As a general rule, we solved the equations with a relative accuracy of 10^{-4} per time step.

7.4.1 The Appearance of Limit Cycles

The numerical solution of the model shows that when the feedback gain, k, is made greater than the critical value, k_0, limit

cycles appear. A limit cycle corresponds to a periodic and bounded solution of a system of equations which describes a closed trajectory in phase space. For the present model, phase space has five dimensions which are: n, the neutron density; c, the delayed neutrons precursors concentration; T, the fuel temperature; ρ_α, the void reactivity feedback; and $d\rho_\alpha/dt$, its derivative. Time is the parameter of the trajectory in phase space. The figures in this thesis can only represent, however, two-dimensional projections of the true five-dimensional limit cycle against some planes. Another type of representation is the time trace of a particular signal. For instance, the neutron density time trace presented in Figure 7.5a shows the development of a typical limit cycle following a 10% step perturbation when the system was originally in the unstable equilibrium point. A detail of the time trace once the limit cycle has been reached (Figure 7.5b) shows that the signal is periodic and that the amplitude of the oscillation is about ±15%, which is of the same order of magnitude as the experimentally observed limit cycle for test condition 7N at Vermont Yankee.[58] This fact gives more credibility to the model and confirms that it is representing the general dynamic behavior of BWRs not only in the linear domain but also in the nonlinear region.

Further increases in the feedback gain have the effect of making the reactor more unstable in the linear or local sense. However, we found that in the nonlinear regime, this increase has the effect of modifying the amplitude of the resulting oscillation.
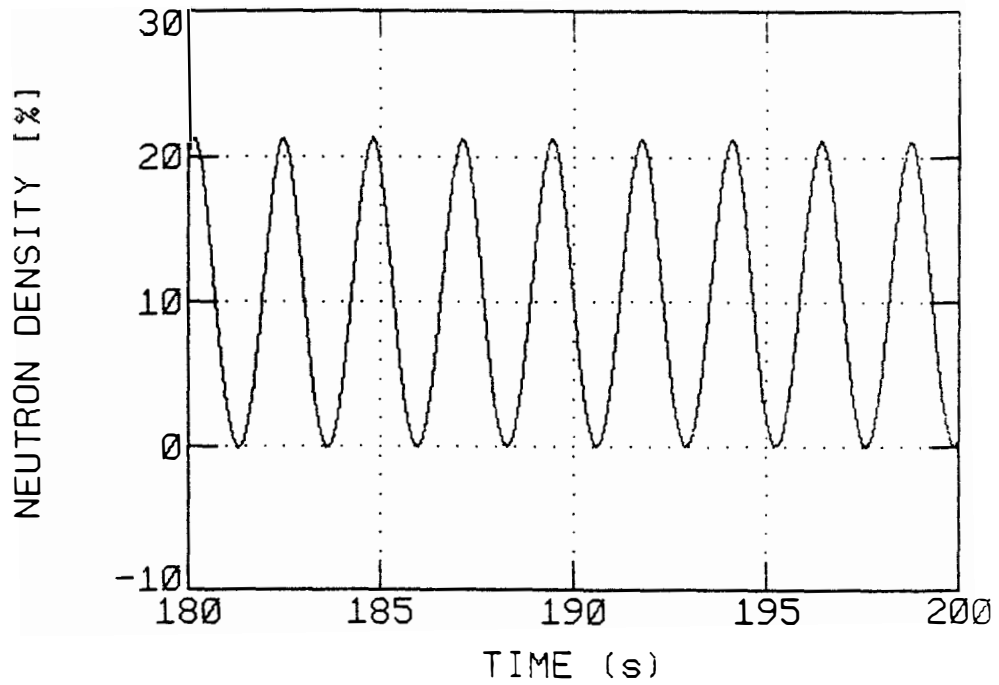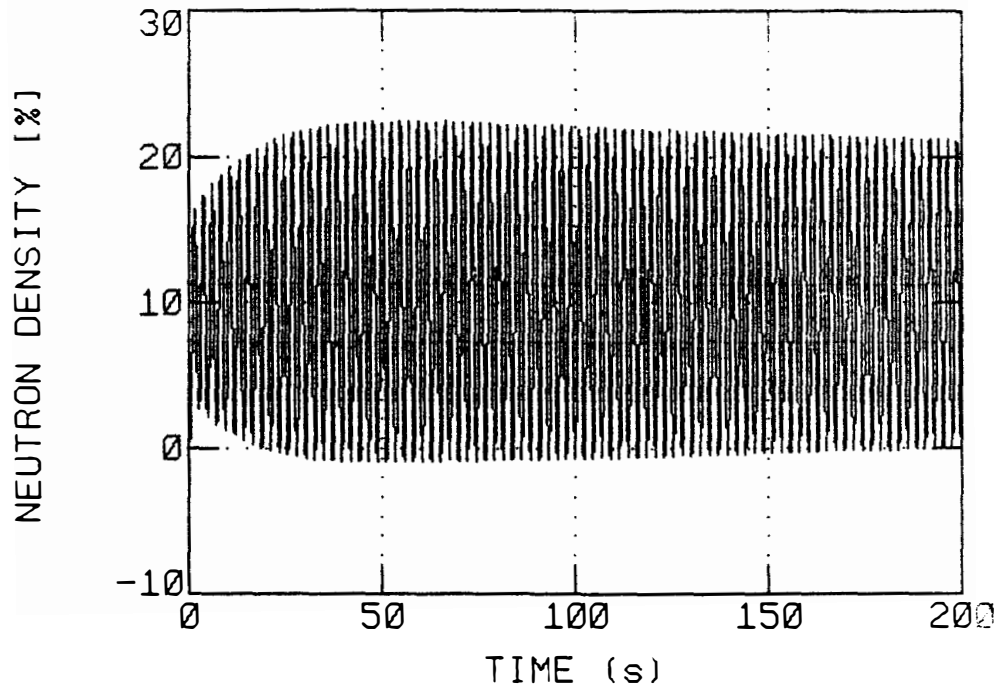
Figure 7.5   The development of a typical limit cycle in the
             time domain.

A limit cycle of large amplitude is presented in Figures 7.6 and 7.7 seen from different perspectives. The main characteristics of this limit cycle can be studied in the time trace plot (Figure 7.6):

(a) The neutron density shows a periodic series of pulses of large magnitude. Between the pulses, the neutron population remains at a low level close to the unstable shutdown condition (equilibrium point B of Section 7.3).

(b) The main feedback path is the channel void reactivity feedback. The average void fraction plotted in Figure 7.6 has been obtained as the ratio between the void reactivity and the void reactivity coefficient. It shows a slightly deformed sinusoidal behavior, which defines the frequency of the neutronic pulses. The pulses appear during the negative part of the void fraction oscillation.

(c) The fuel temperature rises sharply during the neutron pulse, and then it decays exponentially according to the fuel time constant. The temperature oscillation around equilibrium becomes negative and, thus, produces a positive reactivity feedback before the void reactivity does, but it is clear from the timing of the neutron pulses that they are caused by the void reactivity feedback rather than by temperature effects. It can also be observed that the fuel temperature oscillation merely follows the neutronics, whereas the void fraction oscillation (determined by Equation (7-4)) dictates the frequency of the pulses.
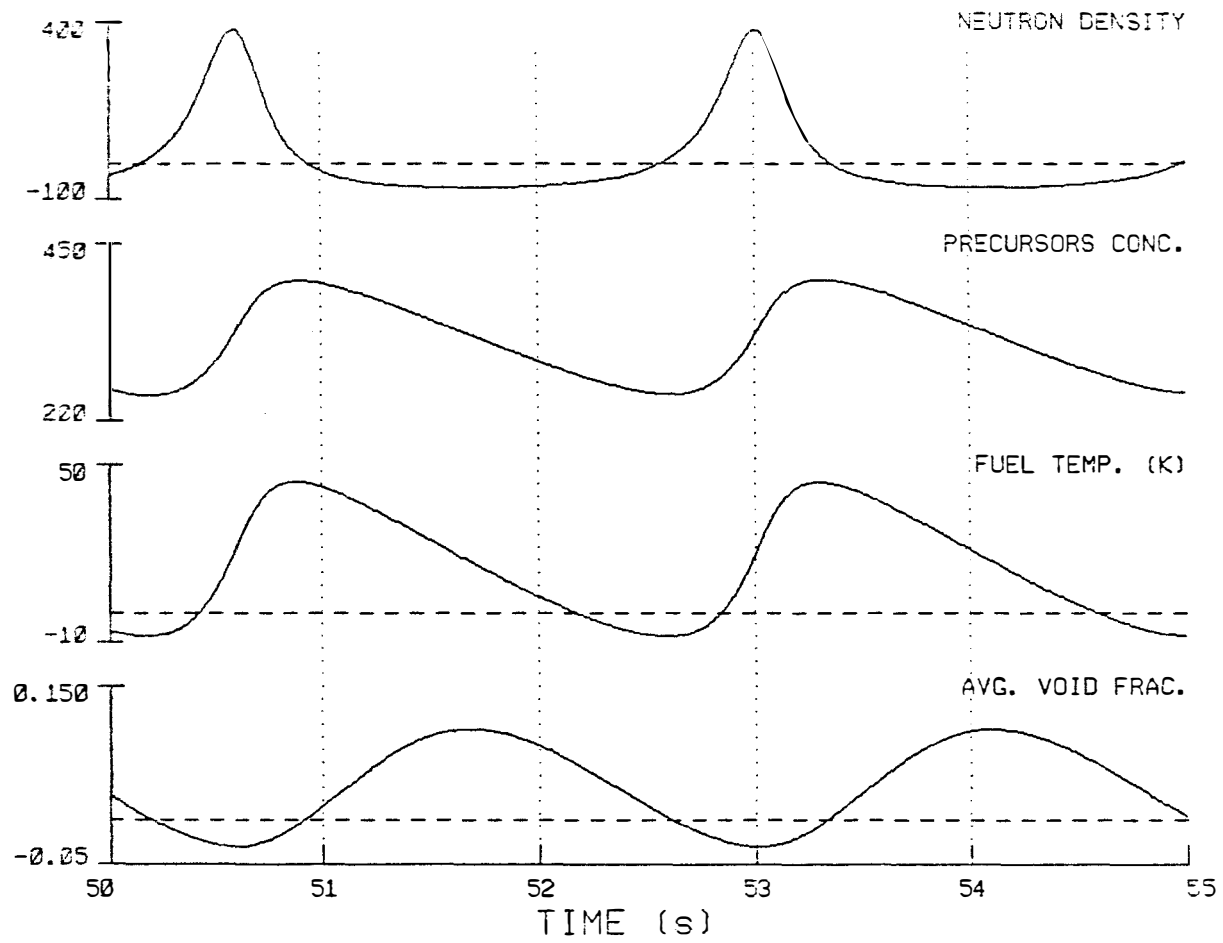
Figure 7.6 Time domain representation of a limit cycle.
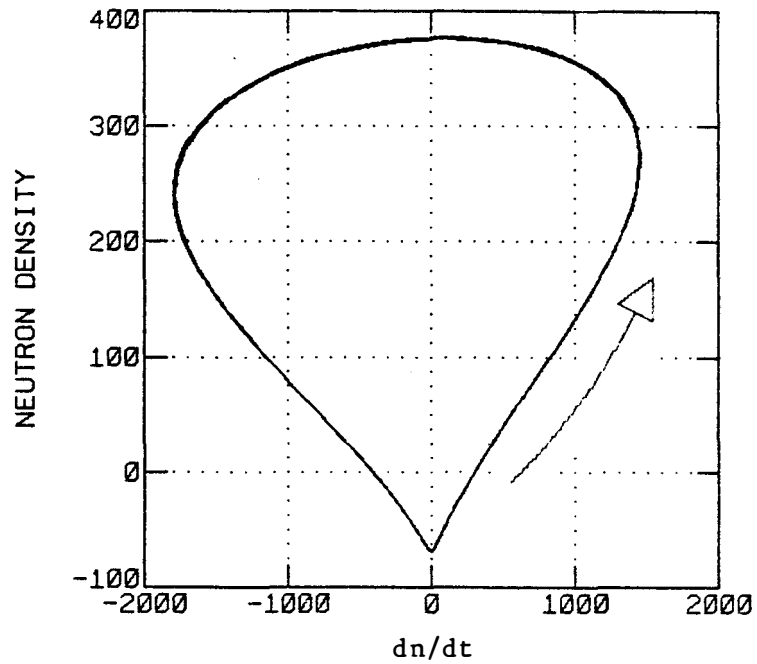
(a) Neutron density versus dn/dt.
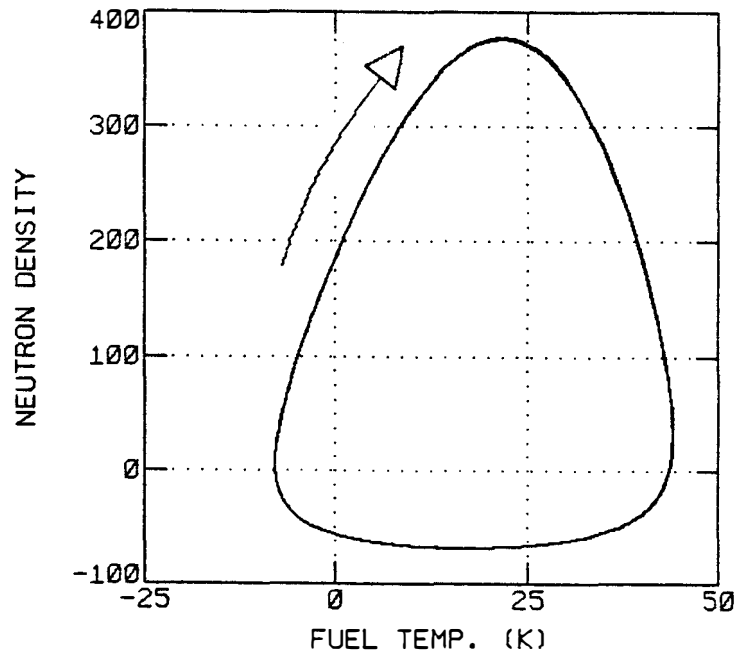
(b) Neutron density versus fuel temperature.

Figure 7.7 Phase space representation of a limit cycle.
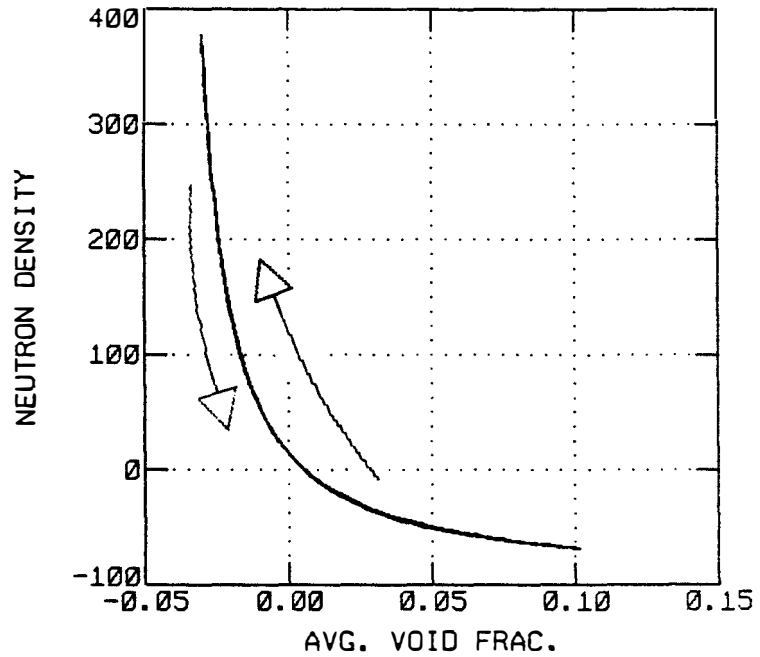
(c) Neutron density versus average void fraction.
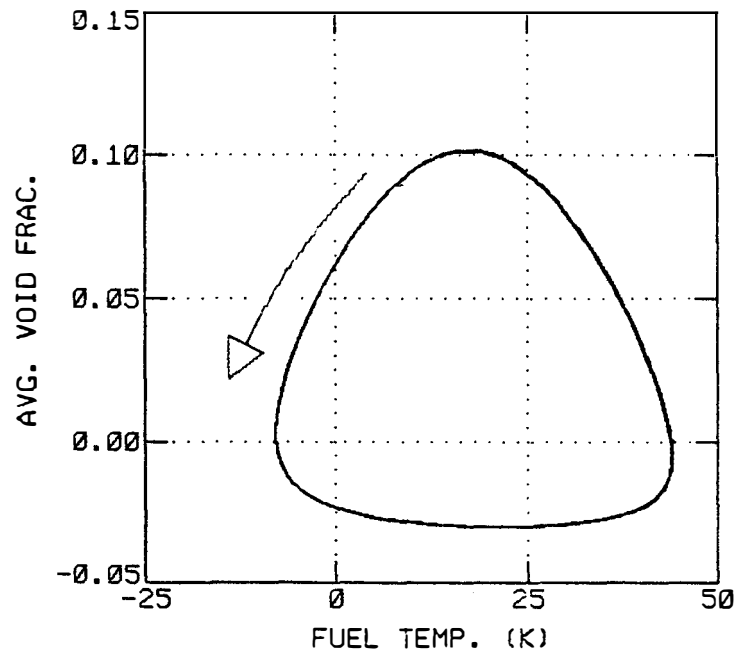
(d) Average void fraction versus fuel temperature.

Figure 7.7 (continued).

(d) Similarly to the case of the temperature oscillation, the
delayed neutrons precursor concentration seems to play a minor
role in the nonlinear reactor dynamics. It rises during the
pulse and then slowly decays according to its own time
constant. The precursors concentration has a major impact,
however, in determining the width of the pulse because the
precursors limit the rising and decaying rates for the neutron
density.[107]

So far we have described the limit cycle in the time domain.
In phase space we can represent the relationship between different
variables by projecting the five-dimensional limit cycle against
two-dimensional planes (Figure 7.7). Some interesting observations
can be made:

(a) The plot of neutron density versus its derivative
(Figure 7.7a) shows a slight asymmetry which implies that the
pulse rises more sharply than it decays due to the effect of
the delayed neutrons. There is also a fast change of the
derivative on the upper part of the plot, which corresponds to
the summit of the pulse. Then the derivative tends to zero as
the neutron density decays towards its minimum value.

(b) The neutron density versus fuel temperature plot (Figure 7.7b)
is even more interesting. The temperature rises as long as
the neutron density is positive. When the neutron population
reaches its minimum, the fuel slowly cools down. The delay

between the time at which the temperature oscillation becomes negative and the time at which the sharp neutron pulse starts can be easily recognized.

(c) Figure 7.7c shows how strongly correlated the neutron density and the void fraction are because in the scale shown in the figure it becomes impossible to resolve the width of the limit cycle. When the void fraction is negative (positive reactivity feedback), the power rises. At this moment the void fraction increases, producing a negative reactivity effect, which causes the neutron density to decrease. The void fraction remains high as long as there is energy stored in the fuel.

(d) As seen in Figure 7.7d (fuel temperature versus average void fraction), the fuel temperature rises sharply, and then it starts to decay. While it decays, the fuel transfers energy to the coolant, and the void fraction increases. Finally, the coolant convection takes over, and the void fraction starts decreasing too. At this moment, a new cycle is started.

As postulated before, the dominant contribution to the overall reactor dynamic behavior in the nonlinear regime is the void reactivity feedback. The effect of the fuel temperature (Doppler) feedback can be further studied in Figures 7.8 and 7.9 which contain neutron density time traces for the base case (Figures 7.8a and 7.9a) and with zero Doppler feedback (Figures 7.8b and 7.9b). Even though
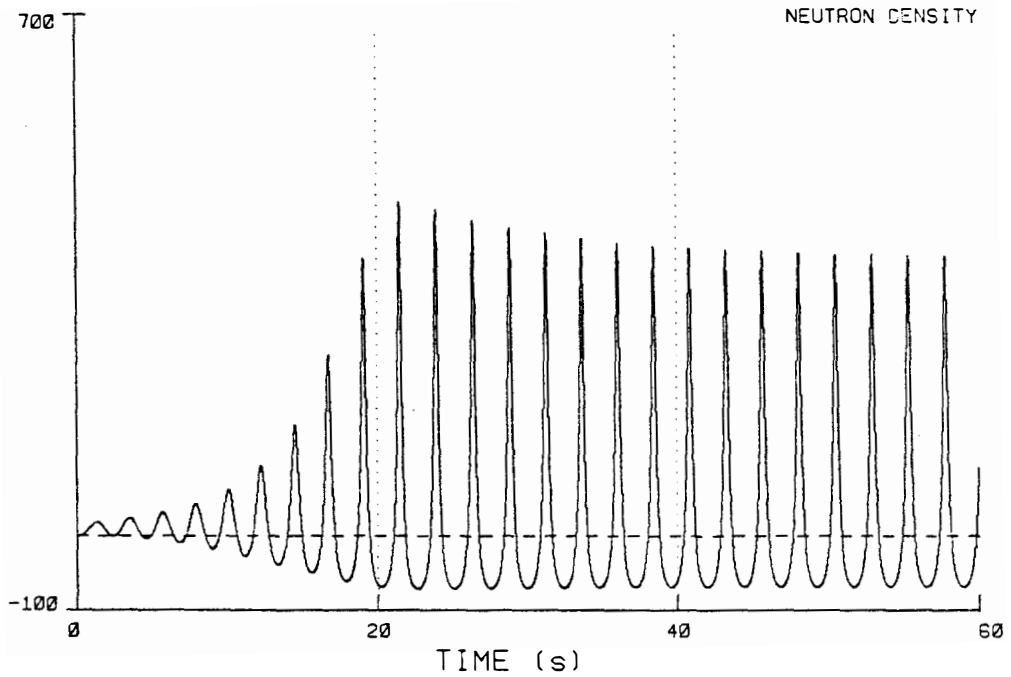
(a) With Doppler feedback
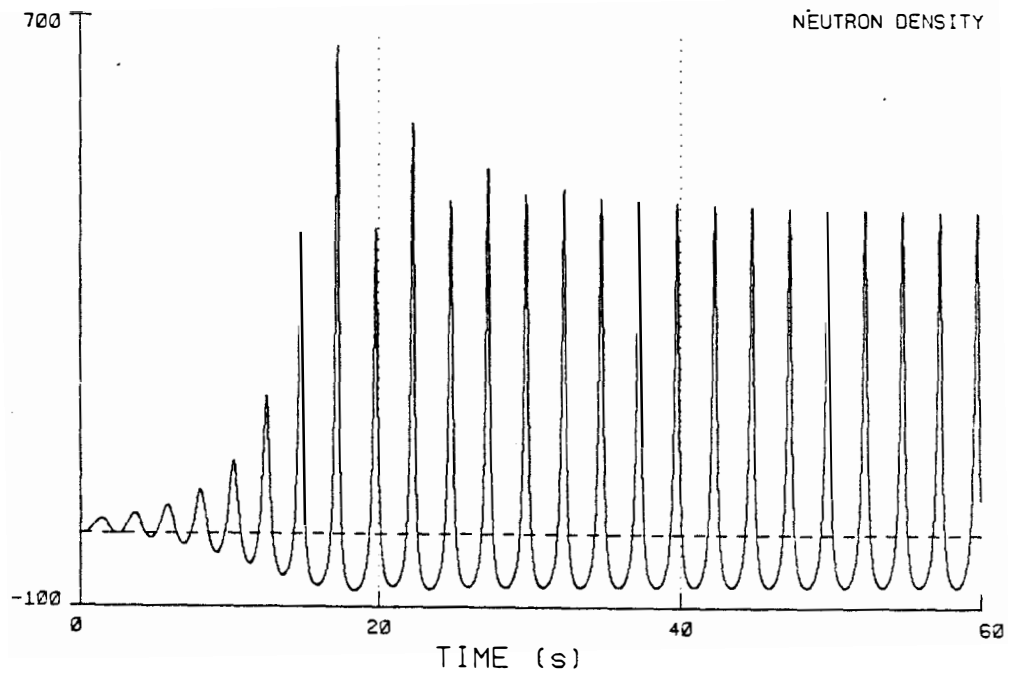
(b) Without Doppler feedback

Figure 7.8 Effect of Doppler feedback on the step response of
the model.

(a) With Doppler feedback   TIME (s)

(b) Without Doppler feedback TIME (s)
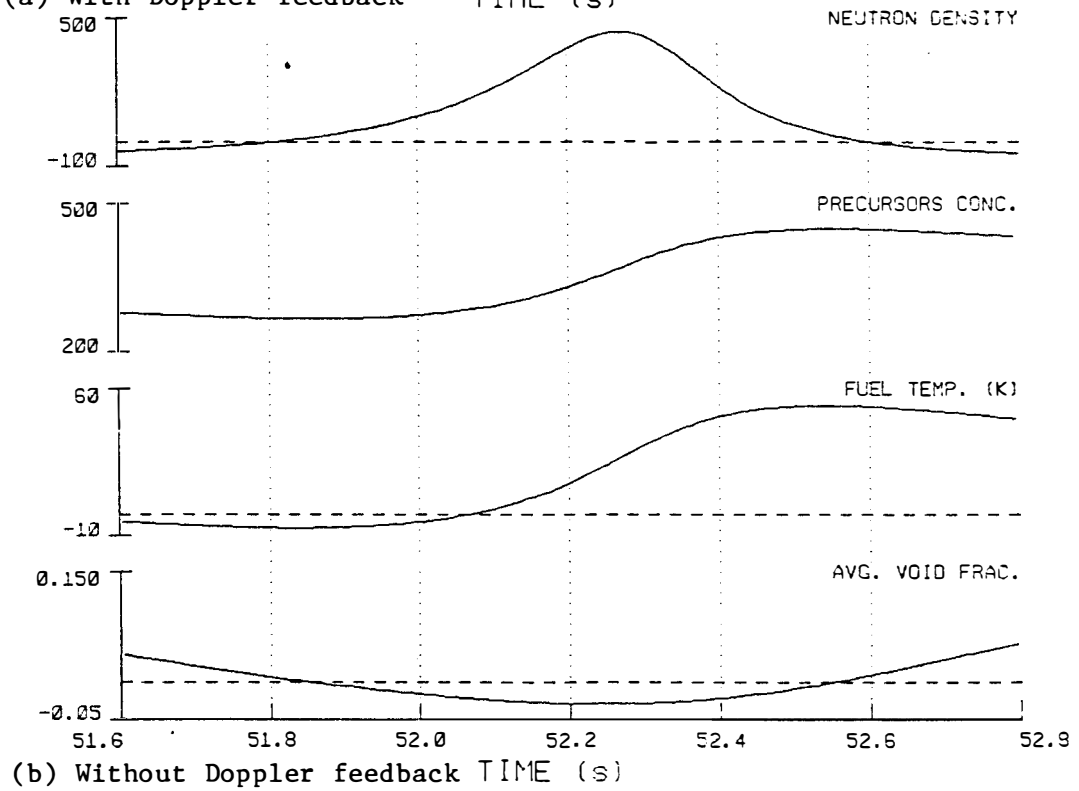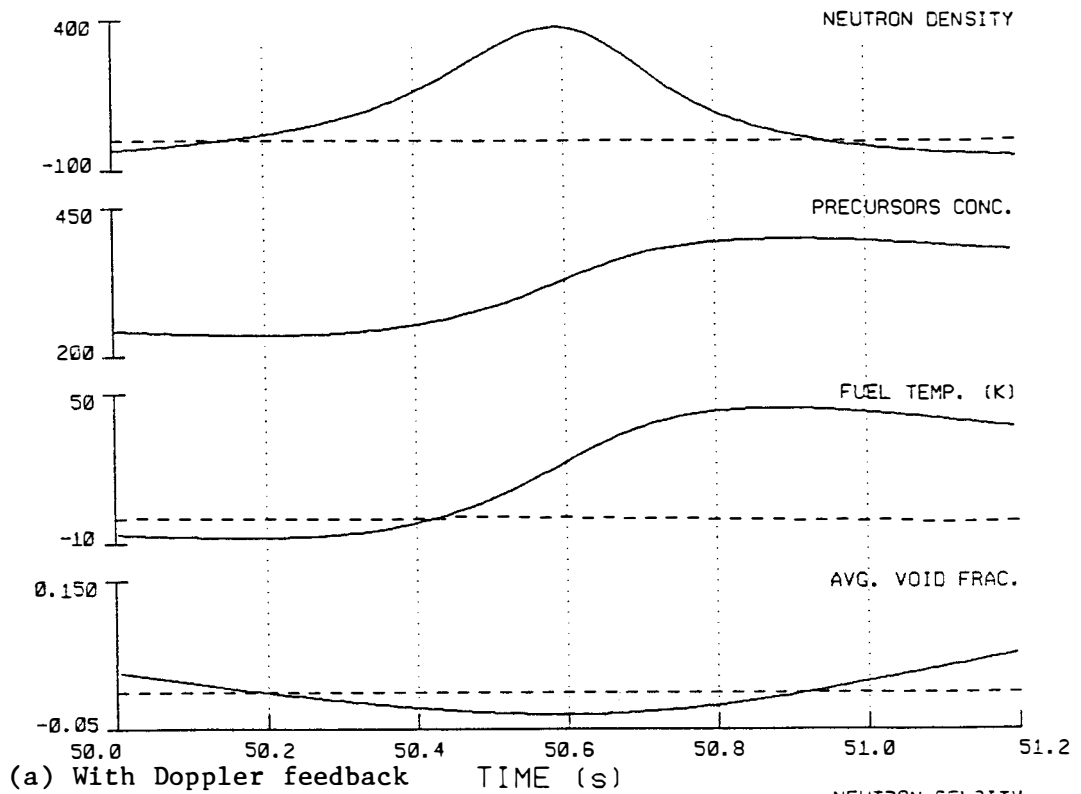
Figure 7.9   Effect of Doppler feedback on the limit cycle
oscillation.

the Doppler effect increases the magnitude of the feedback, we observe

that its presence actually stabilizes the reactor (the limit cycle

amplitude is smaller when the Doppler reactivity coefficient is taken

into account). This apparent paradox is due to the fact that this

type of feedback is practically in phase with the neutron dynamics

and, therefore, does not contribute to the oscillatory (out-of-phase)

type of instability. A second observation is that the time traces

have the same general behavior and pulse frequency with or without

Doppler feedback. This confirms the fact that the pulsed behavior of

the neutronics is indeed produced by the channel thermal-hydraulics

and the corresponding void reactivity feedback.

7.4.2 Limit Cycle Stability: Bifurcations

In Chapter 3 we defined the concept of stability for a linear

system. Here we will expand that concept to the nonlinear regime.

In the linear regime, the system is said to be stable if

following a perturbation the resulting oscillation is eventually

damped and the solution converges to the equilibrium state. For

instance, Figure 7.1b in page 141 contains a typical step response of

the present model in the linear region, when $k < k_0$ and consequently

the reactor is stable. When the system becomes unstable, however, the

solution does not converge to an equilibrium point, but to a new

equilibrium state defined by a limit cycle.

In the linear region the stability of the system is quantified

in terms of an asymptotic decay ratio (DR) defined in Chapter 3;

however, in the nonlinear region, the asymptotic DR is always equal to 1.0, due to the appearance of limit cycles. Therefore, the DR is not a good parameter to describe the dynamic state of the reactor in this region. A better dynamic parameter in this region is the amplitude of the limit cycle oscillations. Consequently, the concern in the nonlinear region is the stability of the amplitude of the oscillations, and not the oscillation themselves. To clarify this point let's look at figures 7.10a, b, and c. These figures show the development of the limit cycle for three different values of the feedback gain: (a) k = 1.2, (b) k = 1.4, and (c) k = 1.5. We observe a clear difference among the way the limit cycle is reached in the three cases. In case (a) the amplitude of the oscillation (which is equal to the maximum value of the pulses, i.e., the signal envelope) follows a smooth curve and promptly converges to the final amplitude. In case (b) the amplitude oscillates around the final value but eventually converges to it. In case (c), however, the amplitude oscillates, but it never converges; it describes an undamped periodic oscillation. This effect can be seen more clearly in Figures 7.10c, d, and e, where the oscillation amplitude has been highlighted. In essence the amplitude of the limit cycle has become unstable and is following a new limit cycle of its own with twice the original period. This causes the original signal to periodically exhibit two pulses of different magnitude. This process is known in physics by the name of a "pitchfork bifurcation."[80]
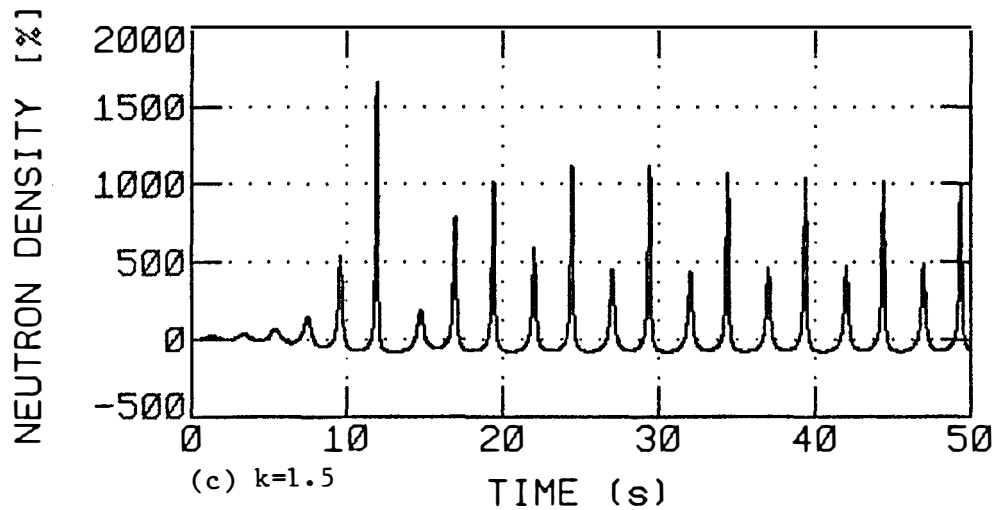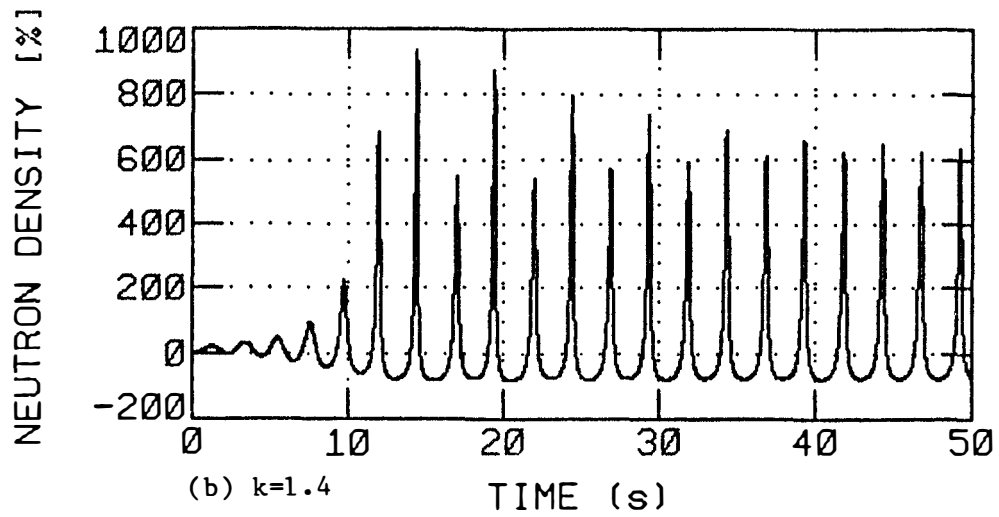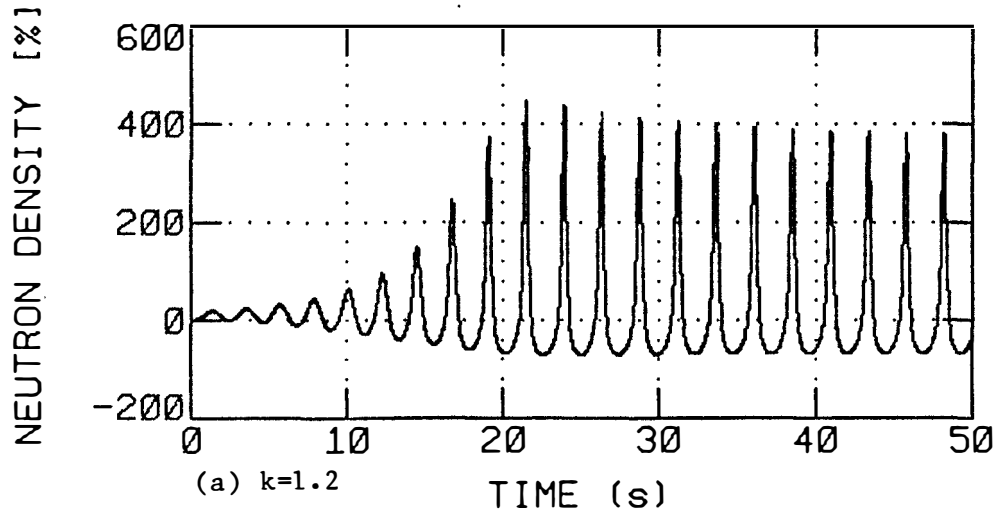
Figure 7.10   Development of an instability of the limit cycle
amplitude.

162

(d) Oscillation amplitude. k=1.2

(e) Oscillation amplitude. k=1.4

(f) Oscillation Amplitude. k=1.5
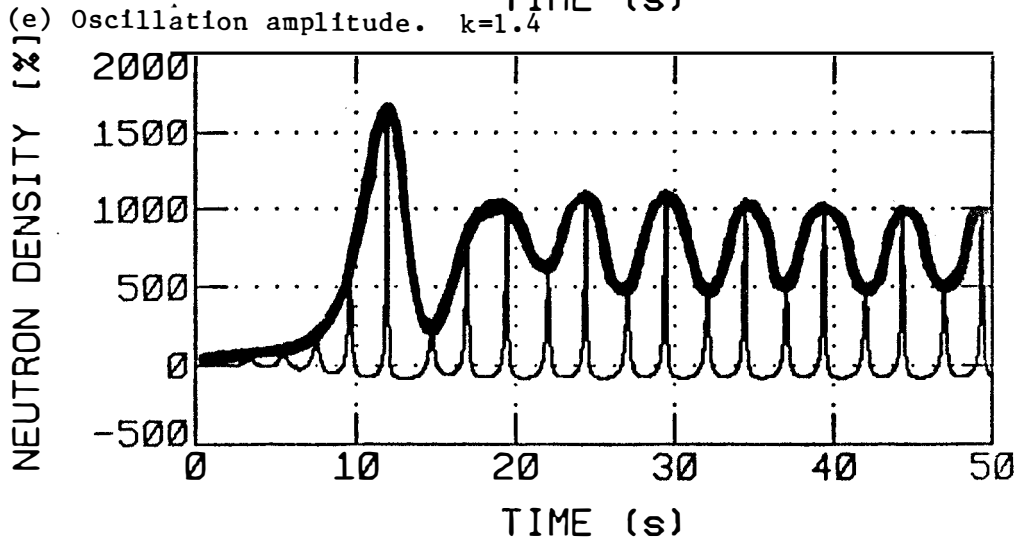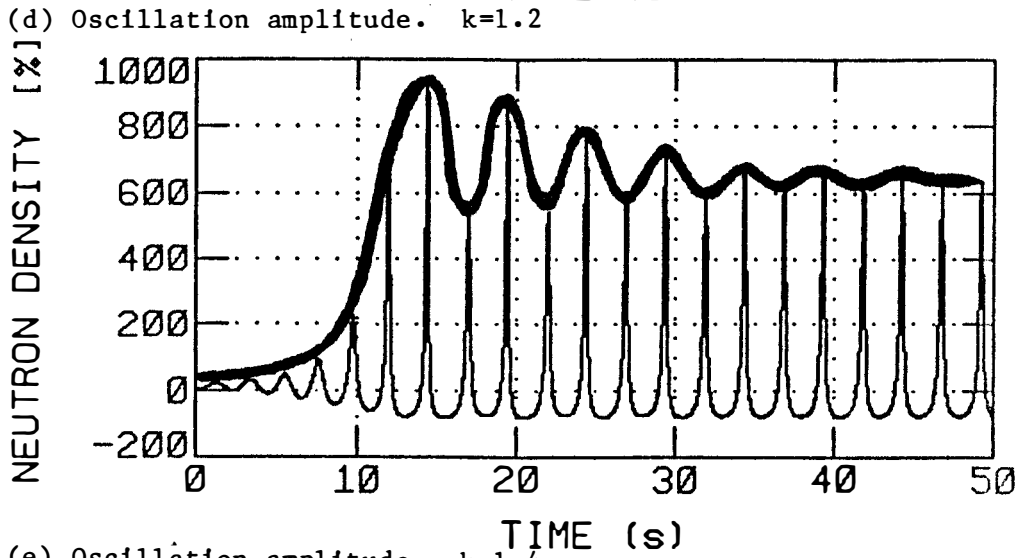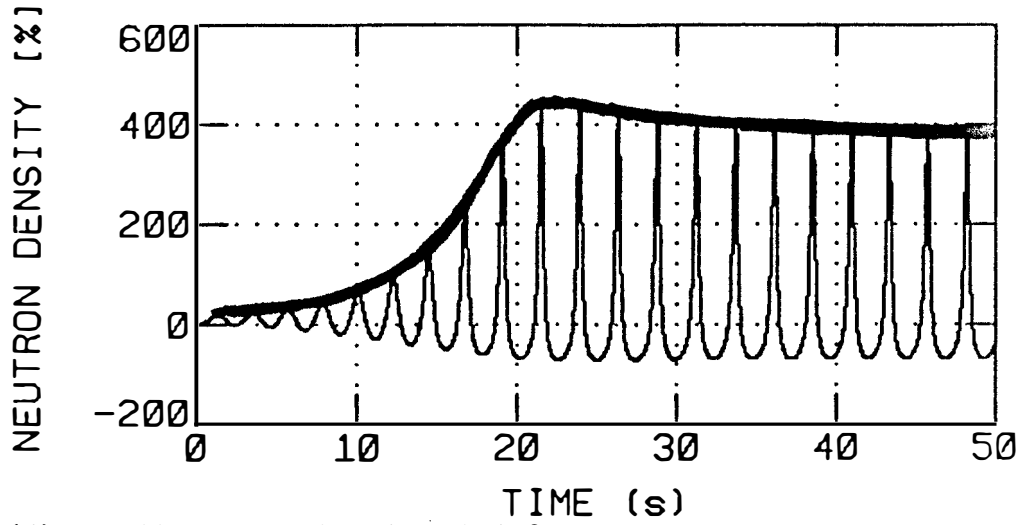
Figure 7.10 (continued).

The bifurcation process can be seen more clearly in phase space. For instance in the neutron-density/fuel-temperature plane (Figure 7.11), the original limit cycle splits (bifurcates) into two. The trajectory described by the solution follows first the inside curve and then the outside curve. The cycle is then repeated in a periodic fashion. The small and large curves in Figure 7.11 correspond to the small and large pulses in Figure 7.10c.

The bifurcation of the limit cycle implies that the basic period of the oscillation approximately doubles because now the trajectory has to make two turns to complete the cycle. This is the reason why this process is also called "period-doubling" bifurcations.

Summarizing, the bifurcation process can be understood as an instability of the limit cycle amplitude when a parameter, k, is increased. For a critical value of the parameter, the amplitude becomes unstable and it oscillates, but it remains bounded due to nonlinearities which force the appearance of a new limit cycle.

At this moment, the process can be complicated if the amplitude of this new limit cycle becomes unstable at some value of k and itself describes another limit cycle (in this case this would be a limit cycle of the amplitude of the limit cycle described by the amplitude of the original limit cycle). In fact, the equations representing our model have this behavior and produce this new bifurcation. The time traces and neutron-density/fuel-temperature phase-space plots are shown in Figure 7.12 for the three cases considered so far.
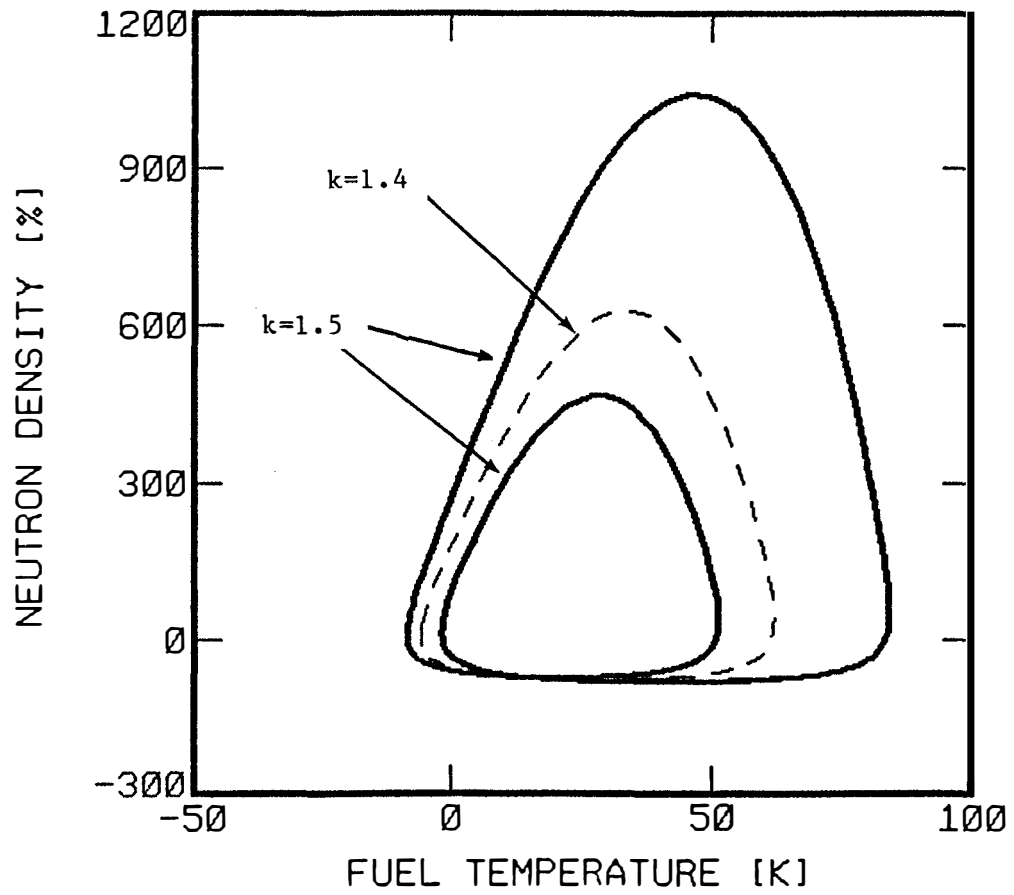
Figure 7.11   Illustration of a period-doubling bifurcation in phase space.
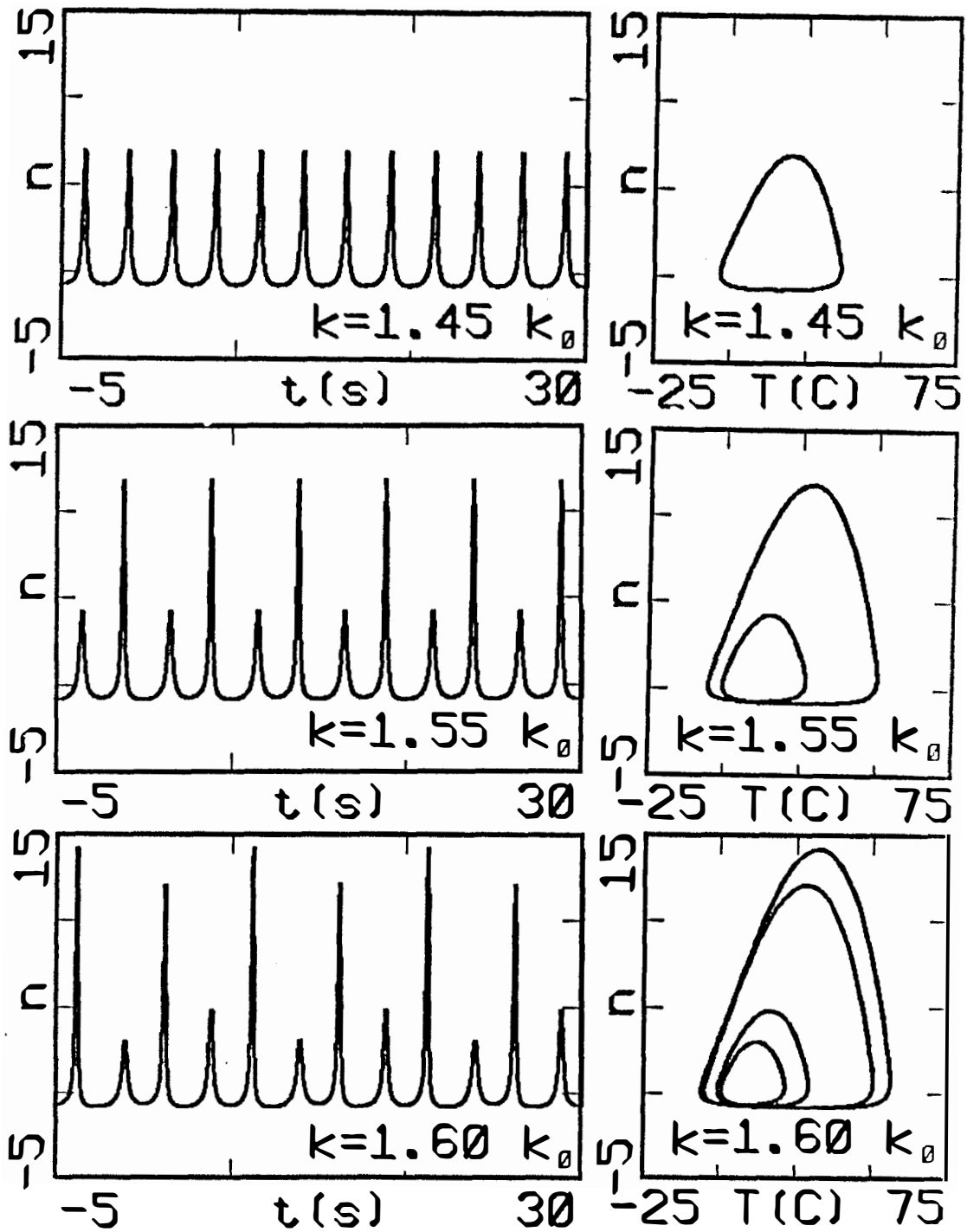
Figure 7.12 Cascade of period-doubling bifurcation as the feedback gain is increased.

We observe that for the latter type of limit cycle, the trajectory describes four full turns in phase space before it closes on itself, thus, it has a period four times as large as the original.

7.4.3 Aperiodicity: Strange Attractors

The bifurcation process described in Section 7.4.2 continues as the value of k is increased further. The amplitude of each new limit cycle becomes unstable at critical values of the feedback gain, k_j, and a new cycle of twice the period appears. This effect is described in the literature as a "cascade of period-doubling pitchfork bifurcations" and it can be visualized in the bifurcation diagram presented in Figure 7.13. In this diagram the maxima and minima of the oscillation are plotted for several values k.

In the region $k < k_0$ the model is stable and the maxima and minima coincide with the equilibrium point. Between k_0 and k_1 the equilibrium point is unstable but a limit cycle exists with amplitude defined by the maxima and minima of the oscillation. When k equal k_1 the amplitude of the original limit cycle becomes unstable and a new limit cycle with two maxima and two minima appears (see Figure 7.12). The two maxima are represented in Figure 7.13 by the two points in the upper branch of the diagram. When the value of k is increased over k_2, this two-turns limit cycle becomes unstable (i.e., its magnitude does) and a new four-turns limit cycle appears.

In this region ($k_2 < k < k_3$) there are four maxima, represented by four points in the bifurcation diagram. In essence, at
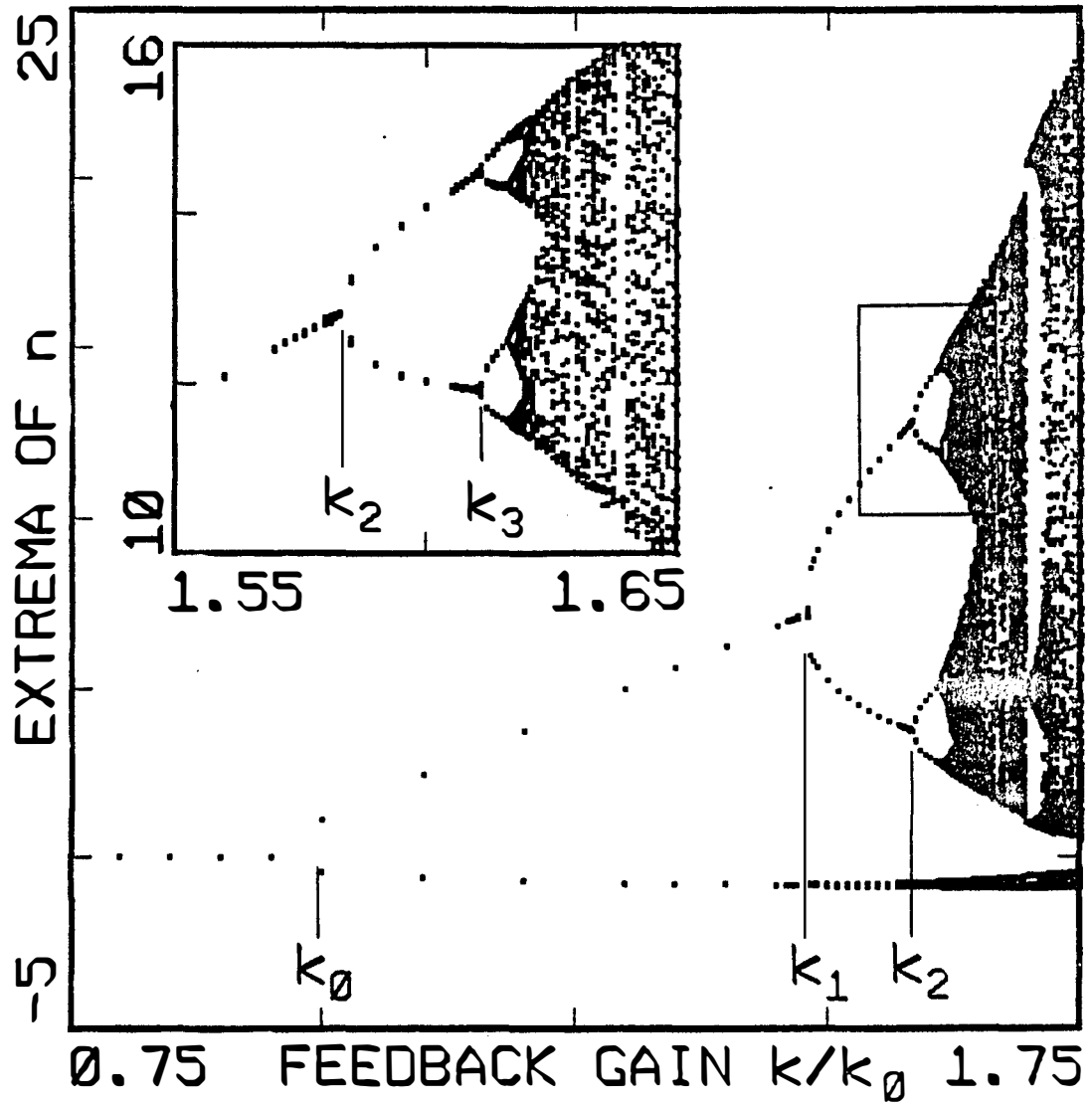
Figure 7.13 Bifurcation diagram showing the accumulation of the critical bifurcation values and the onset of aperiodicity.

every critical value, k_j, each of the branches in the diagram splits (bifurcates) into two new branches.

As seen in the insert in Figure 7.13, the critical values, k_j, occur at shorter intervals as j increases. In fact, we observe a geometric rate of convergence; that is, the distance between two consecutive critical values is decreased by a factor δ_j

$$\frac{k_j - k_{j-1}}{k_{j+1} - k_j} = \delta_j \quad . \qquad (7-9)$$

This implies that the critical values, k_j, converge to an accumulation point, k_∞, beyond which the limit cycle has bifurcated an infinite number of times and, thus, its period is infinite. The solution of the system of equations representing the present BWR model is, consequently, aperiodic (nonperiodic) for values of k greater than the accumulation point, k_∞.

In the aperiodic regime ($k > k_\infty$) the trajectory described by the solution stays bounded within a region of phase space, but it never converges to a closed curve or to an equilibrium point. This type of solution is called in the literature a "strange attractor"[80] because this region in phase space attracts the trajectories towards it but once inside, all the trajectories repel each other so that there is not a final closed curve that could maintain an equilibrium condition. A situation similar to this has been postulated to explain turbulence phenomena.[81]

In the bifurcation diagram of Figure 7.13 the aperiodic region starts at a k slightly larger than k_3. Periodic windows can be identified in the middle of this region. For instance, there is a window of periodicity at about $k = 1.7k_0$. This window corresponds to a limit cycle of period 3 (i.e., it makes three turns before closing on itself). This windows degenerate into aperiodicity through the same cascade of period-doubling bifurcations process described before. Limit cycles of period 3×2^j appear after each successive bifurcation.

The bifurcation process can also be observed in the frequency domain. Figures 7.14a through f present the power spectral densities of the fully developed limit cycles at different levels of the bifurcation. Figure 7.14a corresponds to the first limit cycle with the base period. Only the main resonant frequency can be observed. As the gain of the feedback is increased, the first bifurcation occurs; this appears as the first subharmonic in the frequency domain (Figure 7.14b). Limit cycles of period 2^2, 2^3, and 2^4 are presented in Figures 7.14c, d, and e respectively. Finally, Figure 7.14f contains the PSD of a limit cylce in one of the windows of periodicity. In this case, this condition corresponds to a period of 3×2^2 times the original period.

7.4.4 Universal Nonlinear Behavior in BWRs

Within the precision of the present numerical analysis, seven bifurcations were observed before the aperiodic region was found. The last periodic solution identified made $2^7 = 128$ turns in phase space

(a) Period T

(b) Period 2T

(c) Period 4T

(d) Period 8T

(e) Period 16T
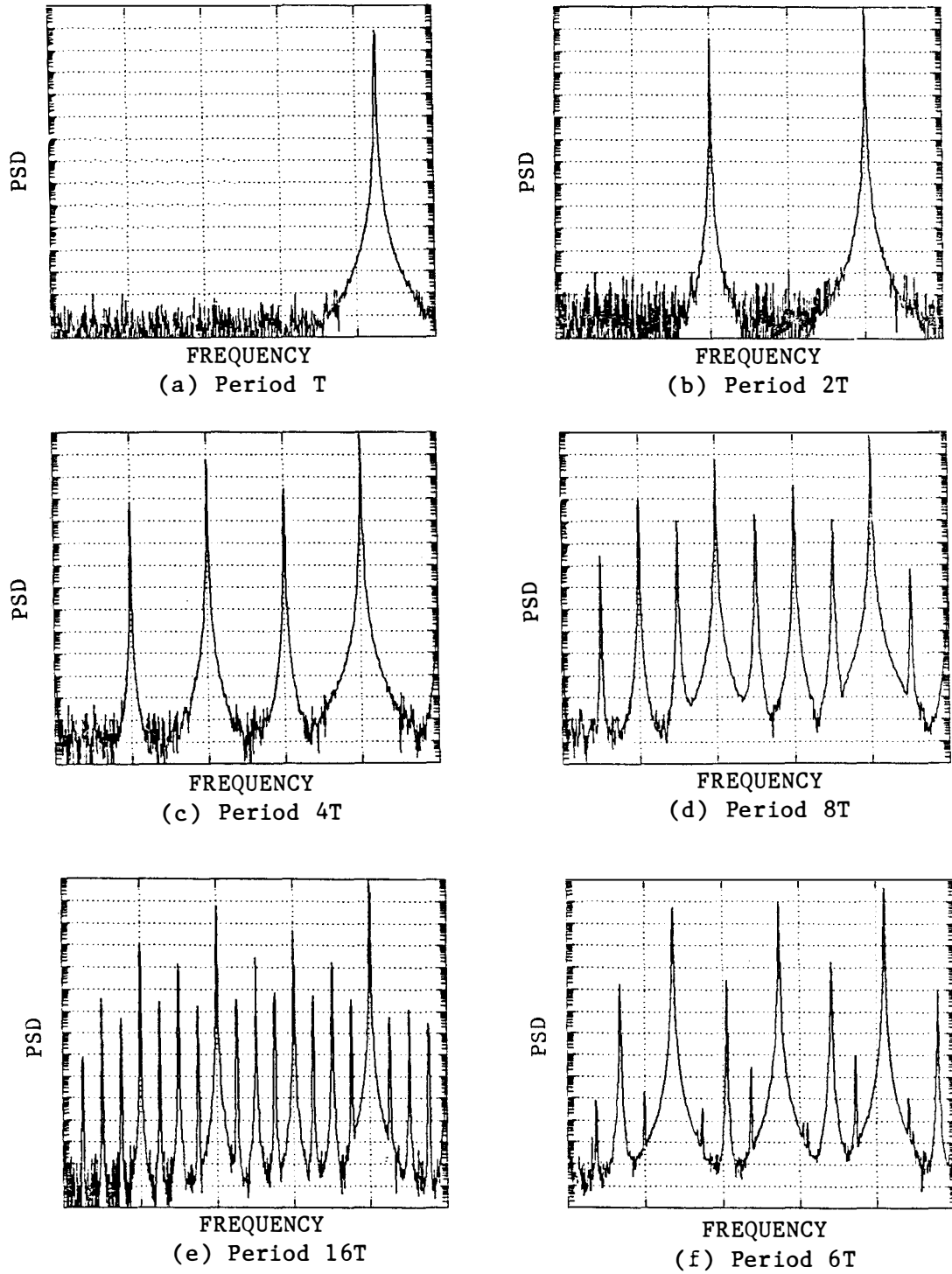
(f) Period 6T

Figure 7.14 The cascade of period-doubling bifurcations
in the frequency domain.

before closing itself. This corresponds to 128 different peak heights in the time trace. The extrapolated value of k_∞ was $k_\infty = 1.61811k_0$. Table 7.2 presents the calculated critical values of k_j/k_0 along with the constants δ_j and α_j defined in Chapter 7.

Table 7.2

ESTIMATION OF THE UNIVERSAL CONSTANTS δ AND α

Cycle j	Period 2^j	Critical bifurcation values k_j/k_0	δ_j	α_j
0	1	1.000000	–	–
1	2	1.470000 ± 0.002	–	–
2	4	1.584000 ± 0.001	4.123 ± 0.126	1.095
3	8	1.610300 ± 0.0001	4.335 ± 0.295	2.207 ± 0.097
4	16	1.616500 ± 0.0001	4.242 ± 0.314	2.391 ± 0.332
5	32	1.617750 ± 0.00001	4.960 ± 0.596	2.465 ± 0.154
6	64	1.618025 ± 0.000005	4.545 ± 0.648	2.517 ± 0.094
∞	∞	1.61811[a]	4.6692...[b]	2.5029...[b]

[a] extrapolated
[b] Reference 82

Feigenbaum's theory[82,83] predicts that as j tends to infinity, δ_j and α_j tend to the universal constants $\delta = 4.6692...$ (the convergence ratio) and $\alpha = 2.5029...$ (the pitchfork scaling parameter). These values are given together with the value (calculated by extrapolation) of the accumulation point k_∞, in the last row of Table 7.2. Overall, there is good agreement between these theoretically predicted values for δ and α, and the calculated

δ_j and α_j. This fact indicates that the present model of a BWR behaves in the nonlinear domain in the universal fashion predicted by Feigenbaum's theory. Many other physical systems have the same universal behavior; among them are: weather equations, transition from laminar to turbulent flow, nonlinear oscillators, nonlinear optics, etc.

7.4.5 The Poincare Map

Feigenbaum's theory[80,83] was developed for noninvertible unidimensional maps of the form

$$x_{k+1} = F(x_k) \tag{7-10}$$

for which the value of the dependent variable, x, depends only on the value of itself in the previous time step. For instance, as shown in Chapter 6, the map

$$x_{k+1} = 4bx_k(1-x_k) \tag{7-11}$$

has the same universal behavior as the BWR model we are studying. The first bifurcation in this simple map occurs at $b_0 = 0.75$. For values greater than b_0 there is a cascade of period-doubling bifurcations. The critical points converge to the accumulation point $b_\infty = 0.892$, and for values of b greater than b_∞ the solution is aperiodic. The convergence ratio, δ_j, and the pitchfork scaling parameter, α_j, converge to the same universal constants δ and α as our BWR model when operating in the nonlinear region.

A way to relate the solution of a system of equations with a one-dimensional map is to create the so-called Poincare map.[80] For

this purpose the trajectory in phase space described by the solution of the set of differential equations is intersected with a particular surface. In this way a set of points are obtained every time the trajectory intersects the surface in a predetermined direction.

A Poincare map for the solution of our model can be obtained by selecting the consecutive maxima of the neutron density time trace as the series of points defined by the map. This is equivalent to selecting as Poincare surface of intersection

$$\frac{dn}{dt} = 0 \qquad\qquad (7-12)$$

which defines a four-dimensional hypersurface (in five-dimensional phase space). The maxima and minima of n(t) lie on this surface.

Once the series of consecutive maxima has been obtained, the Poincare map can be formed graphically by plotting each maximum of n versus the previous maximum. In this way we are drawing the function $F(x_k)$ of Equation (7-7). The results of this process are shown in Figures 7.15a through c for three different values of k: (a) $k = 1.61803k_0$, which corresponds to a periodic solution with $2^7 = 128$ turns; (b) $k = 1.65k_0$, which is at the beginning of the aperiodic region; and (c) $k = 1.8k_0$, which is in the well-developed aperiodic region.

In case (a) the solution is periodic; therefore, there are only 128 different magnitudes of the pulses in the neutron time trace and, consequently, only 128 points appear in the Poincare map. The whole map cannot be defined. Nevertheless, it is clear that the line
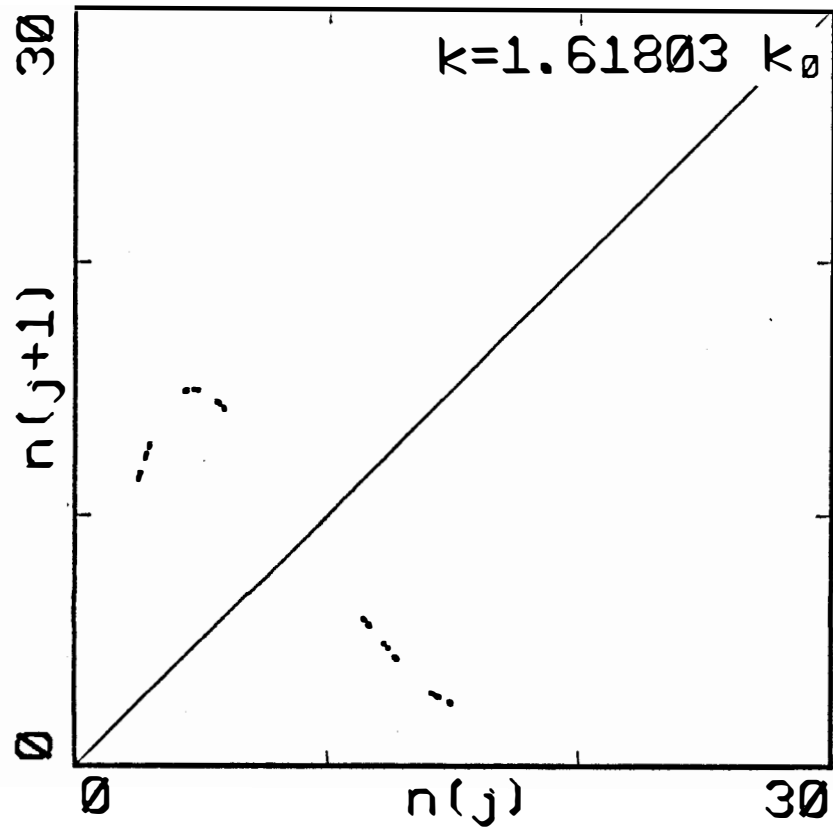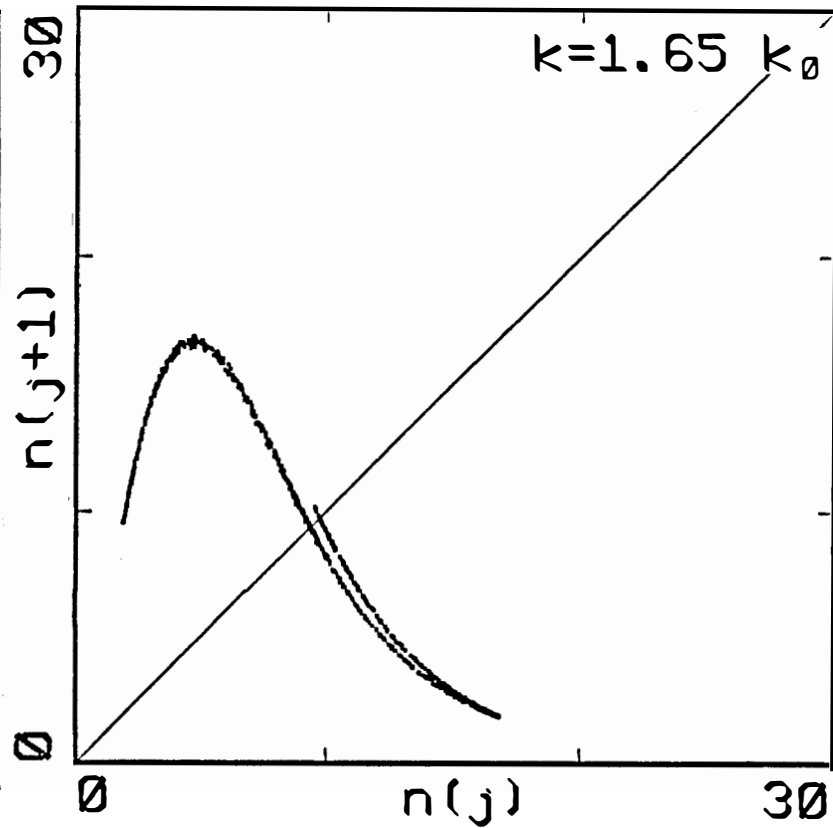
Figure 7.15 Poincare maps of our model showing the quadratic extrema.
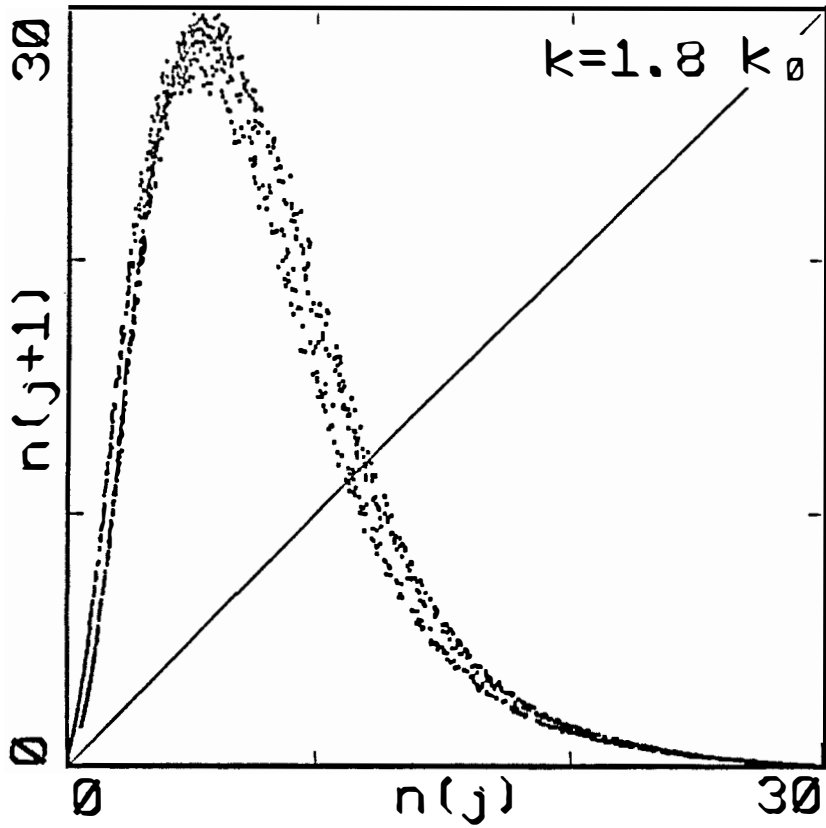
(c) k=1.80 k0

Figure 7.15 (continued).

on which these points lie is smooth and has a quadratic extremum (an extremum is quadratic if the function and its derivative are continuous). The map is clearly noninvertible because there are two possible values of $n(j)$ given $n(j+1)$. Furthermore, it has the stretching and folding capabilities described in Reference 80 necessary for chaotic (or aperiodic) behavior. The fact that the Poincare map of the solution of the present model satisfies the above conditions is the reason why Feigenbaum's universality properties are satisfied.

In the aperiodic region, however, the Poincare map is not as simple (see Figures 7.15b and c). It displays unexpected foldings, indicating that a double-valued relation might exist between successive maxima. This double valuedness, though, is illusory: an examination of the dynamic evolution of the relation between successive maxima reveals the existance of hysteresis, in that the solution evolves either on the lower or on the upper branch according to whether the magnitudes of preceding maxima form an increasing or a decreasing sequence. This hysteresis indicates that a many-term recursion relation is needed to represent the behavior of this BWR model in the aperiodic region.

7.5 Stochastic Numerical Analysis: Nonlinear Noise in BWRs

Section 7.4 has dealt exclusively with a deterministic analysis of the nonlinear behavior of BWRs. In this section we study the

effect of nonlinearities on the behavior of the reactor under stochastic (random) excitations (sources). For this purpose, the model was externally driven with a band-limited Gaussian white noise, and the equations were solved numerically in the time domain using an A-stable routine.[100]

Two parameters were varied: the feedback gain, k, and the variance of the driving noise source. The generated power traces, n(t), were Fast Fourier transformed to obtain power spectral densities (PSDs). The development of the limit cycle in the time domain is shown in Figure 7.16, where the envelopes (maxima and minima) of the oscillation are plotted as a function of time for three different values of the noise-source variance. For this figure, the system was held originally at the unstable equilibrium point. At time t = 0 a zero-mean white noise was applied. The amplitude of the oscillations increased initially until it reached a limit cycle. The amplitude of these limit cycle oscillations is independent of the magnitude of the driving noise variance as expected.

The study of the effects of the feedback gain on system behavior shows that for stable systems, $k < k_0$, the neutron PSD exhibits a single peak at the reactor characteristic frequency of oscillation as predicted by linear studies (see Chapter 4). However, as k approaches k_0 while maintaining the driving-source variance constant, the PSD develops peaks at the harmonics of this fundamental frequency. For $k > k_0$, the power oscillations increase in time and eventually reach
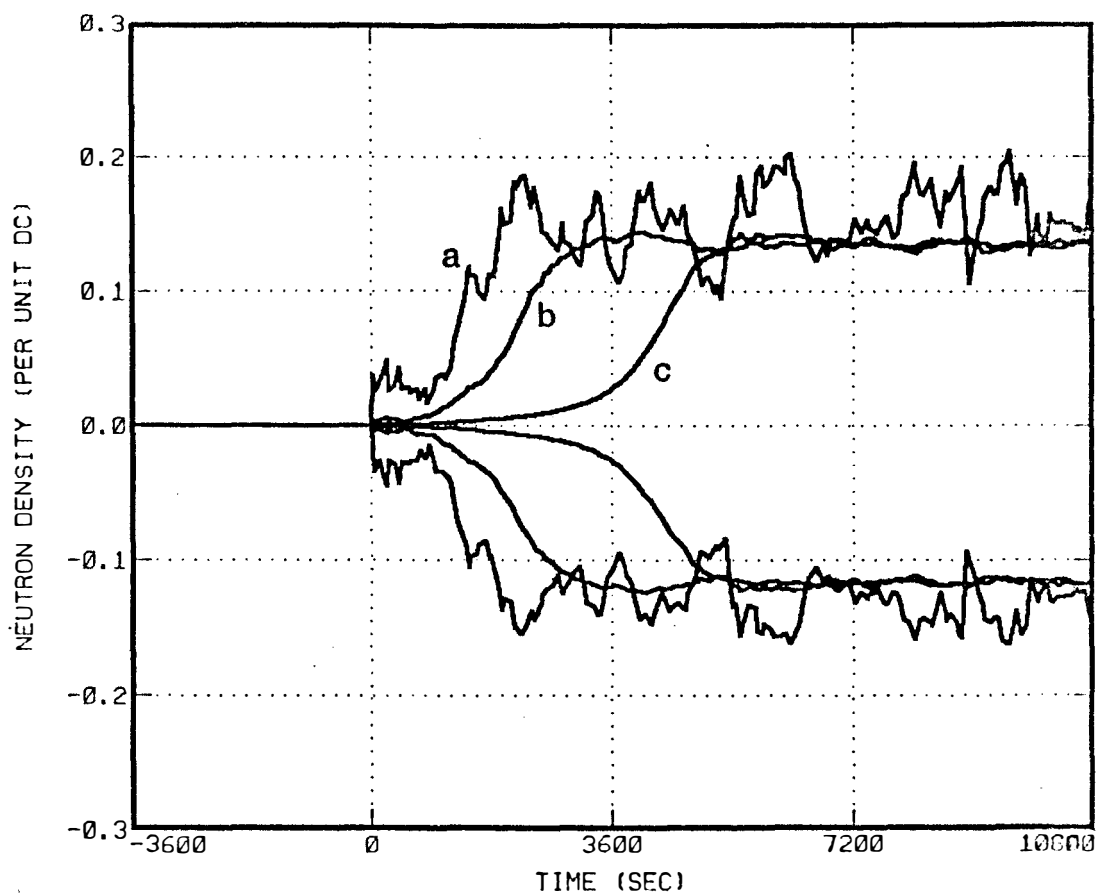
Figure 7.16   Envelopes of the development of a limit cycle in the
presence of noise.   (a) 10^{-1} noise variance.
(b) 10^{-3} noise variance.   (c) 10^{-5} noise variance.

a limit cycle, with an enhancement of the harmonic components of the PSD as seen in Figure 7.17. This figure shows three PSDs for different values of k. In case (a) the model is barely stable and only the fundamental peak is clearly discernible at about 0.4 Hz. Case (b) represents a small amplitude limit cycle, for which the value of k is only slightly above the critical value, k_0. Case (c) corresponds to a fully developed large amplitude limit cycle. The main difference between the stable and the unstable PSDs is the appearance of higher harmonics. These harmonics have a strong magnitude and they should be measurable in real-life experiments in which measurement and process noise are present.

One of the consequences of the appearance of a limit cycle in a reactor is an increase of the variance (noisiness) of the neutron density as seen by the in-core neutron detectors. However, an increase in neutron noise variance could also be due to an increase in the noise of other variables, flow for instance, which in turn drives the neutronics. It is of interest to be able to distinguish between these two kinds of noise increases, because different corrective action might need to be taken depending on the cause. For example, if the increase in noise is due to an instability (i.e., the appearance of a limit cycle), the reactor can be made more stable by increasing the flow rate through the core and, thus, eliminate the extra noise. However, if the increase in noise is due to a pump malfunction, which causes flow noise, an increase in flow would only make the problem worse.
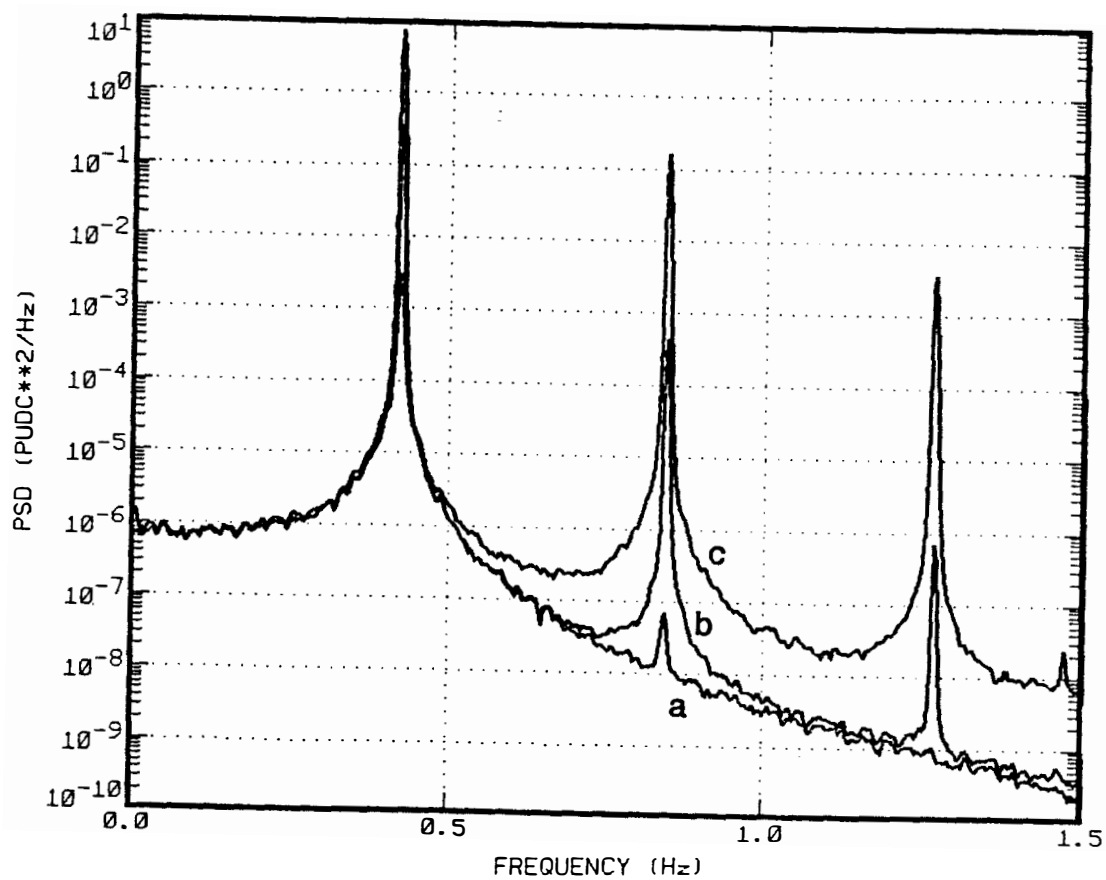
Figure 7.17   Power spectral densities before and after the
development of a limit cycle. (a) Slightly stable.
(b) Slightly unstable. (c) Fully developed
limit cycle.

This problem is a present concern with regard to BWR single loop operation. When some reactors are operated with a single pump and above 40% of rated flow, there is an increase in neutron noise, which at the moment is unexplained. It is necessary to develop a technique to differentiate between the two scenarios previously presented.

The present model was used to generate neutron density time traces for a stable and an unstable condition. The variance of the imput noise source was adjusted in order that the variance of the output neutron noise be of the same order of magnitude in both cases. The resulting time traces are presented in Figures 7.18a and b. Although there are obvious differences between the unstable reactor condition (a) and the stable one (b), it is not easy to determine if case (a) is really a limit cycle or not. A simple, more sensitive technique is required to differentiate the mode of operation. Figures 7.19a and b contain the PSDs of the time traces for the two above cases. Here the differences are more obvious. In case (a), where the reactor is unstable, the characteristic peak at about 0.4 Hz is very sharp. The main difference, however, is the appearance of high harmonics in the neutron PSD. Case (b), the stable condition, also has harmonic contamination, due to the large amplitude of the noise, but since the peak is wider, this contamination is not as obvious as in case (a). Furthermore, in a real-life measurement it would be impossible to distinguish harmonic contamination from the

(a) Limit cycle

(b) Externally induced noise
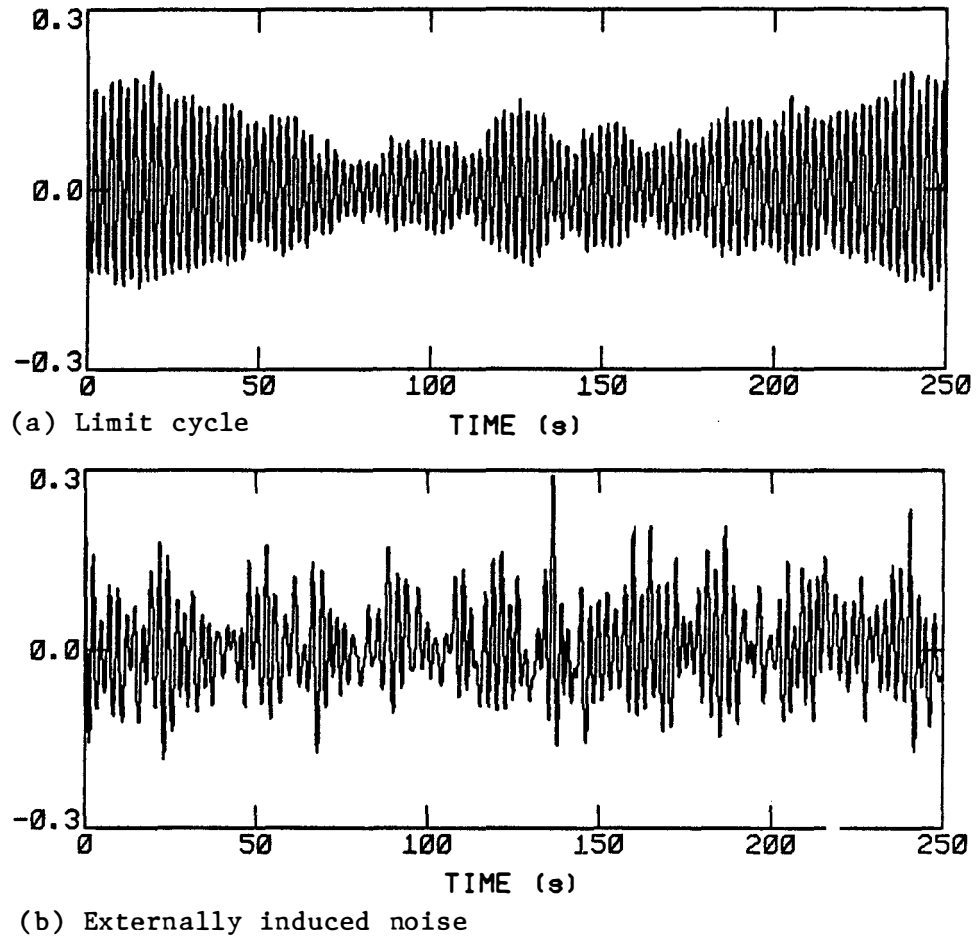
Figure 7.18 Comparison between limit cycle oscillations and externally-induced noise in the time domain.

(a) Externally-induced noise
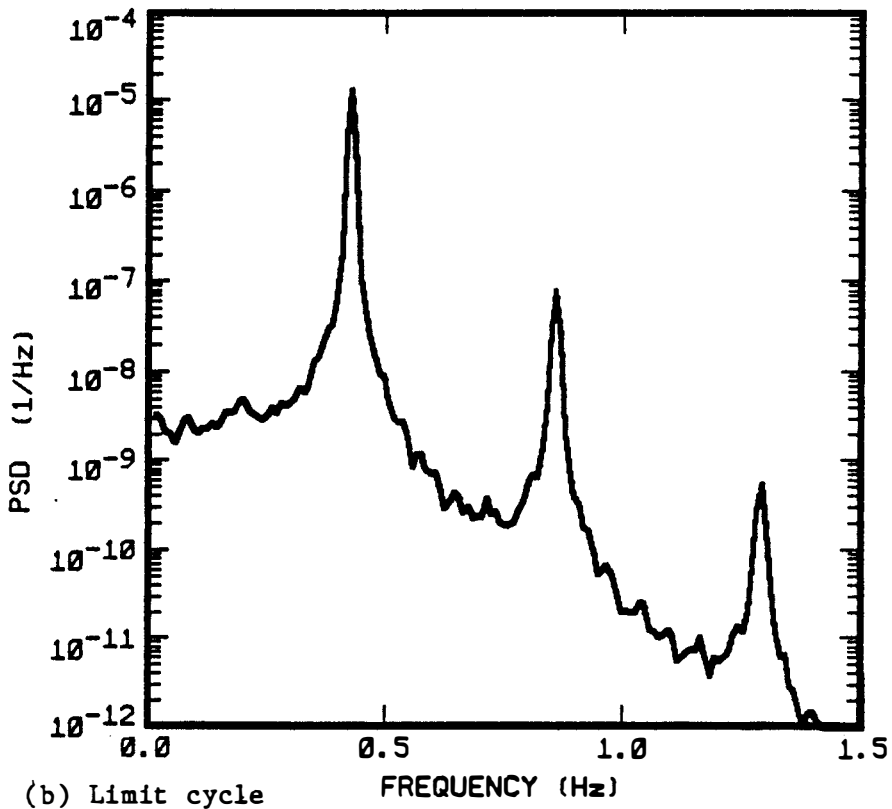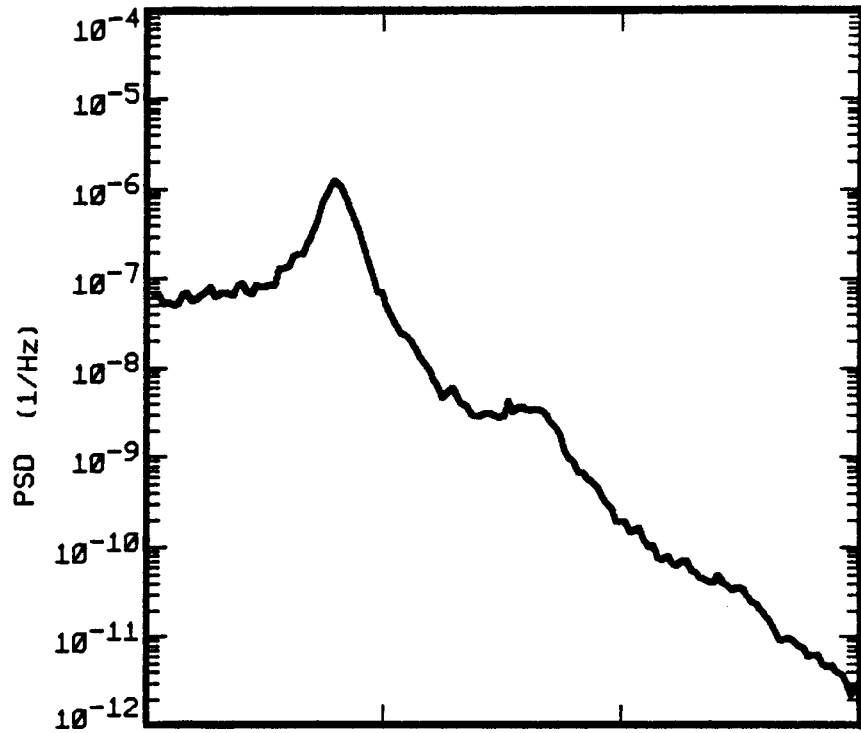
(b) Limit cycle

Figure 7.19 Comparison between limit cycle oscillations and
externally-induced noise in the frequency domain.

measurement and process noise. In summary, we have seen that if large amplitude oscillations occur, nonlinearities appear as harmonic contamination in the neutron PSD. If the oscillations are due to the appearance of limit cycles, the harmonic contamination is easily distinguishable from the background noise and it shows as sharp peaks at harmonic frequencies of the characteristic peak. This technique, then, can be used to differentiate between limit cycle oscillations and increased noise source variance.

CHAPTER 8

A MORE COMPLETE NONLINEAR MODEL

The model used in Chapter 7 was simple enough to allow a detailed parametric study and even some analytical results. However, it had some serious restrictions; the most important one being the fact that we assumed a linear thermohydraulic feedback. This assumption is based on the fact that the fuel element filters out most of the heat generation changes at the frequencies of interest. Indeed, only about 10% of the heat generated in the fuel is transfered to the coolant at 0.5 Hz; and at that same frequency, only about 3% of the generated heat produces changes in the average void fraction in the core.

The feedback parameters used in the model of Chapter 7 were obtained from a fit to the results of a LAPUR[48] run for which the reactor was close to instability. This model is supposed to work properly only for small changes of these parameters. In this chapter we will develop a nonlinear model for BWR dynamics which is entirely based on first principles and which maintains all nonlinearities. The model will then be applied to a reactor with conditions close to test 7N of the Vermont Yankee low flow stability tests[58] with the goal of obtaining quantitative results.

8.1 The Model

8.1.1 The Neutron Field

The point-kinetics approximation was used to represent the neutron dynamics

$$\frac{dN}{dt} = \frac{\rho - \beta}{\Lambda} N + \lambda C \tag{8-1}$$

$$\frac{dC}{dt} = \frac{\beta}{\Lambda} N - \lambda C \quad . \tag{8-2}$$

We define normalized neutron density and precursor concentrations as

$$n(t) = (N(t) - N_0)/N_0 \tag{8-3}$$

$$c(t) = (C(t) - C_0)/N_0 \quad . \tag{8-4}$$

The equations become

$$\frac{dn}{dt} = \frac{\rho - \beta}{\Lambda} n + \lambda c + \frac{\rho}{\Lambda} \tag{8-5}$$

$$\frac{dc}{dt} = \frac{\beta}{\Lambda} n - \lambda c \quad . \tag{8-6}$$

Note that the reactivity feedback appears as a parameter in Equation (8-5); since the dynamic reactivity is a function on $n(t)$, this equation is nonlinear. This was the only nonlinearity considered in Chapter 7 and caused all the effects presented previously. In this model, however, we will consider the feedback nonlinearities also.

8.1.2 The Fuel

Assuming only radial conduction, the equation governing the temperature distribution within the rod is

$$\rho c_p \frac{\partial T}{\partial t} = Q + \frac{1}{r} \frac{\delta}{\partial r} \left(rk \frac{\partial T}{\partial r} \right) \quad . \tag{8-7}$$

This equation is linear in T except for the dependence of the conductance, K, on the temperature. Assuming that K is constant, we can apply a procedure similar to the one used in Section 4.4.1 to define average fuel and clad temperatures, which we name T_f and T_c respectively. The resulting equations are

$$\langle \rho c_p \rangle_F \frac{dT_F}{dt} = \langle Q \rangle_F + \frac{2}{r_F} k \left. \frac{\partial T}{\partial r} \right|_{r=r_F} \tag{8-8}$$

$$\langle \rho c_p \rangle_c \frac{dT_c}{dt} = \frac{2r_c}{r_c^2 - r_F^2} k \left. \frac{\partial T}{\partial r} \right|_{r=r_c} - \frac{2r_c}{r_c^2 - r_F^2} k \left. \frac{\partial T}{\partial r} \right|_{r=r_F} \, , \tag{8-9}$$

with boundary conditions

$$k \left. \frac{\partial T}{\partial r} \right|_{r=r_F} = - h(T_F - T_c) \tag{8-10}$$

$$k \left. \frac{\partial T}{\partial r} \right|_{r=r_c} = - U(T_c - T_s) \, , \tag{8-11}$$

where h is the effective fuel-to-clad gap heat transfer coefficient and U is the effective overall heat transfer coefficient between clad and coolant. T_s is the coolant temperature, which we will assume constant and equal to the saturation temperature all along the channel. The final equations are

$$\frac{dT_F}{dt} = \frac{\langle Q \rangle_F}{\langle \rho c_p \rangle_F} - \frac{2h}{r_F \langle \rho c_p \rangle_F} (T_F - T_c) \tag{8-12}$$

$$\frac{dT_c}{dt} = \frac{2h r_F}{(r_c^2 - r_F^2) \langle \rho c_p \rangle_c} (T_F - T_c) - \frac{2U r_c}{(r_c^2 - r_F^2) \langle \rho c_p \rangle_c} (T_c - T_s) . \tag{8-13}$$

Typical values of the constants involved are shown in Table 8.1. The values of h and U are obtained from a fit to the LAPUR fuel transfer function calculated for the conditions of interest.

Table 8.1

TYPICAL VALUES OF THE CONSTANTS IN THE FUEL EQUATIONS

Constant	Value	Units
r_f	0.00503	m
r_c	0.00601	m
$\langle \rho c_p \rangle_f$	3.125×10^6	J/m^3K
$\langle \rho c_p \rangle_c$	15.18×10^6	J/m^3K
h	1.827×10^3	J/m^2Ks
U	0.103×10^3	J/m^2Ks

The heat generation within the fuel is proportional to $n(t)$

$$\langle Q \rangle_F = P \, n(t) \quad , \tag{8-14}$$

where the coefficient P depends on the steady state power N_0. For the values of the constants in Table 8.1 and the parameters of the Vermont Yankee reactor, the total heat transferred to the coolant per unit length of fuel is given by the expression

$$\Delta Q'(J/ms) = 8.876 \times 10^8 \; \Delta T_c \; (K) \; . \tag{8-15}$$

The above two expressions provide the coupling between the fuel and neutronics and thermal-hydraulics equations respectively.

8.1.3 The Channel Thermal-Hydraulics

Neglecting acoustic phenomena,[29] the one-dimensional channel equations (mass, energy, and momentum) can be written as[45]

$$\frac{\partial}{\partial t} \left[(1-\alpha)\rho_1 + \alpha\rho_g \right] + \frac{\partial G}{\partial z} = 0 \tag{8-16}$$

$$\frac{\partial}{\partial t} \left[(1-\alpha)\rho_1 h_1 + \alpha\rho_g h_g \right] + \frac{\partial}{\partial z} \left[(1-x)h_1 G + x h_g G \right] = Q' \tag{8-17}$$

$$\frac{\partial p}{\partial t} = -\frac{\partial G}{\partial t} - \frac{1}{2}\frac{\partial}{\partial z}\left[\frac{(1-x)^2 G^2}{\rho_l(1-\alpha)} + \frac{x^2 G^2}{\rho_g \alpha}\right] - \left[(1-\alpha)\rho_l + \alpha\rho_g\right]g_c$$

$$-f\frac{\phi^2 \Omega G^2}{2\rho_l D} - \sum \frac{Ki\phi^2\Omega G^2}{2\rho_l}\,\delta(z-z_i) \quad , \tag{8-18}$$

where

G = Coolant mas flux in Kg/m2s

α = Void fraction

x = Steam quality

p = Pressure in N/m^2

ρ_g = Saturated steam density in Kg/m^3

ρ_l = Saturated liquid density in Kg/m^3

h_g = Saturated steam enthalpy in J/Kg

h_l = Saturated liquid enthalpy in J/Kg

f = Single phase friction coefficient

ϕ^2 = Martinelli-Nelson correction factor

for two-phase flow pressure drop

Ω = Jones correction factor

D = Equivalent flow diameter in m

Ki = Concentrated pressure loss coefficients

Q' = Heat transferred per unit length of fuel

x, Φ^2, and Ω are functions of the void fraction, the operating pressure, and the flow rate.

Equations (8-16) through (8-18) are a set of partial differential equations. We will eliminate the space dependence by integrating over finite nodes. In this way a set of ordinary differential

equations for the nodal variables is obtained. For this purpose, we apply the operator

$$\frac{1}{\Delta_i} \int_{z_{i-1}}^{z_i} dz \qquad\qquad (8\text{-}19)$$

to Equations (8-16) through (8-18). Define the nodal variables as

$$G_i(t) = \frac{1}{\Delta_i} \int_{z_{i-1}}^{z_i} G(z,t)\ dz \qquad\qquad (8\text{-}20)$$

$$\alpha_i(t) = \frac{1}{\Delta_i} \int_{z_{i-1}}^{z_i} \alpha(z,t)\ dz \qquad\qquad (8\text{-}21)$$

$$Q_i'(t) = \frac{1}{\Delta_i} \int_{z_{i-1}}^{z_i} Q'(z,t)\ dz \qquad\qquad (8\text{-}22)$$

$$p_i(t) = \frac{1}{\Delta_i} \int_{z_{i-1}}^{z_i} p(z,t)\ dz \qquad\qquad (8\text{-}23)$$

Equation (8-16) becomes

$$G(z_i,t) = G(z_{i-1},t) + \Delta_i(\rho_1 - \rho_g)\ \frac{\delta\alpha_i(t)}{\delta t} \qquad\qquad (8\text{-}24)$$

using Equation (8-24), Equation (8-17) becomes

$$\frac{\partial\alpha_i}{\partial t} = \frac{Q_i' - (h_1 - h_g)\ G(z_{i-1})\ [x(z_i) - x(z_{i-1})]/\Delta_i}{[\rho_g h_g - \rho_1 h_1] + [h_1(1 - x(z_i)) + h_g x(z_i)](\rho_1 - \rho_g)} \qquad\qquad (8\text{-}25)$$

The momentum equation becomes

$$p(z_{i-1}) = p(z_i) + \Delta_i\frac{\partial G_i}{\partial t} + \left[EC(z_i) - EC(z_{i-1})\right]$$

$$+ \Delta_i\left[(1-\alpha_i)\rho_1 + \alpha_i\rho_g\right] + \Delta_i FR_i + \frac{K_0}{2\rho_1}\ G_0{}^2\delta(i,1)\quad, \qquad\qquad (8\text{-}26)$$

where $EC(z)$ and FR_i are the kinetic and friction terms respectively, and given by the expressions

$$EC(z_i) = \frac{1}{2} \left[\frac{(1-x)^2 G^2}{\rho_1(1-\alpha)} + \frac{x^2 G^2}{\rho_g \alpha} \right] \tag{8-27}$$

$$FR_i = \frac{1}{\Delta_i} \int_{z_{i-1}}^{z_i} f\frac{\Phi^2}{2\rho_1} \frac{\Omega}{D} G^2 dz \simeq f\frac{\Phi^2_i}{2\Delta_i\rho_1 D} \frac{\Omega}{D} \left[\frac{G(z_i)+G(z_{i-1})}{2} \right]^2. \tag{8-28}$$

Boundary conditions are

$$p(z=H,t) = p_0 \text{ ,} \tag{8-29}$$

because the upper plenum pressure is held constant by the pressure regulator.

$$\alpha(z=0,t) = 0 \text{ .} \tag{8-30}$$

The coolant at the inlet of the channel is assumed at saturation temperature, but all in liquid phase. The last boundary condition is

$$G(z=0,t) = G_0(t) \text{ ,} \tag{8-31}$$

where $G_0(t)$ is determined by the recirculation loop dynamics (see Section 8.1.4).

The initial conditions are

$$\alpha(z,t=0) = \alpha_0(z) \tag{8-32}$$

$$G(z,t=0) = G_0 \tag{8-33}$$

$$p(z,t=0) = p_0(z) \tag{8-34}$$

and they are determined by the steady state equations.

The numerical procedure of solution of these equations is:

(a) At each time step Equation (8-25) is solved to obtain the new value of the nodal void fraction.

(b) The new mass fluxes are calculated from Equation (8-24)

(c) The pressure distribution along the channel is calculated using Equation (8-26) starting from the boundary condition

$p(H) = p_0$. This procedure yields the lower plenum pressure, which determines the pressure drop across the channel. This pressure drop is the input to the recirculation loop equations which in turn determine the inlet mass flux rate, $G_0(t)$.

8.1.4 The Recirculation Loop

The recirculation loop is formed by the upper plenum, the steam separators and dryers, downcomer, jet pumps, and lower plenum. A complete model of this system would be extremely complex. We will treat it as a single path of fluid with variable flow areas but constant mass flow rate (i.e., incompressible flow). It is convenient to rewrite momentum equation in terms of flow areas, $A(z)$, and a constant flow rate, ω, instead of mass flux, G.

$$-\frac{\partial p}{\partial t} = \frac{\partial}{\partial t}\left(\frac{\omega}{A}\right) + \frac{1}{A}\frac{\partial}{\partial z}\left(\frac{\omega^2}{\rho_1 A}\right) + \rho_1 g + \frac{\partial}{\partial z}\left(\Delta P_f\right) \quad , \qquad (8-35)$$

where ΔP_f represents the integrated friction losses and can be considered proportional to ω^2. Note that A depends on z but not on t, whereas ω depends on t but not on z. Integrating over the path of the recirculation loop we obtain

$$P_1 - P_2 = \left(\frac{L}{A}\right)\frac{\partial \omega}{\partial t} + \frac{\omega^2}{2\rho_1}\left(\frac{1}{A_2^2} - \frac{1}{A_1^2}\right) + \rho_1 g(z_2 - z_1) + \Delta P_f \qquad (8-36)$$

where we define

$$\left(\frac{L}{A}\right) = \int_1^2 \frac{1}{A(z)}\, dz \quad . \qquad (8-37)$$

Defining

$$H = z_1 - z_2 \qquad (8-38)$$

and

$$\Delta P_f = k\omega^2 \qquad\qquad (8\text{-}39)$$

the equation becomes

$$\left(\frac{L}{A}\right)\frac{\partial\omega}{\partial t} = \left[(p_1-p_2)+\rho gH\right] - \omega^2\left[\frac{1}{2\rho_1}\left(\frac{1}{A_2{}^2} - \frac{1}{A_1{}^2}\right)+k\right] \quad . \qquad (8\text{-}40)$$

Note that in steady state, $\partial\omega/\partial t = 0$ and

$$\left[\frac{1}{2\rho_1}\left(\frac{1}{A_2{}^2} - \frac{1}{A_1{}^2}\right)+k\right] = \left[p_1-p_2+\rho gH\right]_0 \quad . \qquad (8\text{-}41)$$

Substituting Equation (8-41) in (8-40) and considering that the mass flux at the inlet of the channel is

$$G(z_0) = \frac{\omega}{A} \quad , \qquad\qquad (8\text{-}42)$$

the recirculation loop equation becomes

$$\frac{\partial G(z_0)}{\partial t} = \frac{1/A}{\left(\frac{L}{A}\right)}\left[p_1-p_2+\rho gH\right]_0 \left[1-\frac{G^2(z_0)}{G_0{}^2(z_0)}\right]$$

$$+ \frac{1/A}{\left(\frac{L}{A}\right)}\left[(p_1-p_2)-(p_1-p_2)_0\right] \quad . \qquad (8\text{-}43)$$

This equation relates the inlet mass flux to the pressure drop across the core. Consequently, it couples momentum equation with the mass and energy balances.

8.2 The Program TLAP

The above equations have been implemented in the computer code TLAP, which is written in FORTRAN-77. A listing of the program with a sample input is contained in Appendix D.

The code solves the equations in the time domain using an A-stable ordinary differential equation solver.[106] Up to 12 axial nodes can be used to solve the channel thermal-hydraulics equations. The one-dimensional void reactivity feedback is calculated from the void fraction spatial distribution and serves as coupling between the channel equations and the point kinetics representation of the neutronic equations. The fuel behavior is represented by a two-node expansion, corresponding to the pellet and cladding, respectively, with explicit representation of the pellet-to-clad gap. The recirculation loop is also modeled as a single-node integral momentum equation and serves as coupling between the channel pressure drop and the inlet mass flux.

Several empirical correlations are used in the code. All of them are based on the correlations used by LAPUR[48] for consistency.

The slip ratio is computed using the modified Bankoff empirical correlation determined by A. B. Jones.[17-22] In the bulk boiling region, which is the case in all our channel, the correlation becomes

$$s = \frac{1 - \alpha}{k_s - \alpha + (1-k_s)\alpha^r} \quad , \qquad (8\text{-}44)$$

where k_s and r are functions of the operating pressure.[48] Their nominal value at 1000 psi is $k_s = 0.8$ and $r = 3.97$.

With the slip ratio and the steam quality known, the void fraction, α, is given by the expression

$$\alpha = \frac{\rho_1 x}{s\rho_g + x(\rho_1 - s\rho_g)} \quad .$$ (8-45)

The two phase pressure drop is normally related to the single phase pressure drop through a multiplier coefficient. The Martinelli-Nelson correlation[100] is the one most commonly used. LAPUR uses a polynomial fit valid for steam qualities less than 0.7 (a typical BWR exit quality is 0.15). In this code we used only the first coefficient of the polynomial as an approximation. In this way,

$$\Phi^2 = 1 + 30x \quad .$$ (8-46)

This expression approximates LAPUR's correlation to within ±5% up to steam qualities of x = 0.8.

Jones[17-22] calculated a correction factor for the Martinelli-Nelson coefficient which depends on the flow rate and pressure. For the nominal operating pressure, the Jones correction factor can be approximated as

$$\Omega = 1.9 - 5 \times 10^{-4} G \quad ,$$ (8-47)

where G is the mass flux in $Kg/m^2 s$.

The single phase friction coefficient, f, can be considered constant in the turbulent region (a typical BWR Reynolds number is 70000). The value of f for a typical new fuel element is f = 0.019. A multiplier factor of 1.4 is applied to account for aging and curd deposition processes which increase the friction.

The density reactivity feedback is calculated as a weighted integral of the density reactivity of each individual node. The

density reactivity coefficient for each node is calculated as a function of the void fraction. A second order polynomial was fitted to the density reactivity coefficient calculated from a void dependent two-group cross-section set.[63] The resulting correlation is

$$\frac{d\rho}{d\alpha} = -0.108 - 0.207\alpha + 0.140\alpha^2 - 0.135\alpha^3 \quad , \quad (8-48)$$

where ρ is measured in absolute units (i.e., $\rho = \Delta K/K$.)

8.3 Application to a Typical BWR

The conditions of test 7N in the Vermont Yankee low-flow stability tests[58] were modeled using the code TLAP. The input parameters were extracted from the results of a detailed LAPUR calculation. Then, the density reactivity coefficient was adjusted by a factor of 0.97 so that the inception of the limit cycle oscillations would correspond approximately to the conditions of the test. Once the effective model parameters were determined, the operating power and flow were changed to study the dynamic behavior of the reactor over a wide operating range.

8.3.1 The Limit Cycle

A typical limit cycle produced by TLAP is shown in Figure 8.1. In figure 8.1a, the power time-trace, n(t), is presented as it develops a limit cycle from equilibrium. Once the limit cycle is reached (Figure 8.1b) the power oscillates between 30 and 120%. Therefore, these operating conditions (64% power and 32% flow) would not cause an
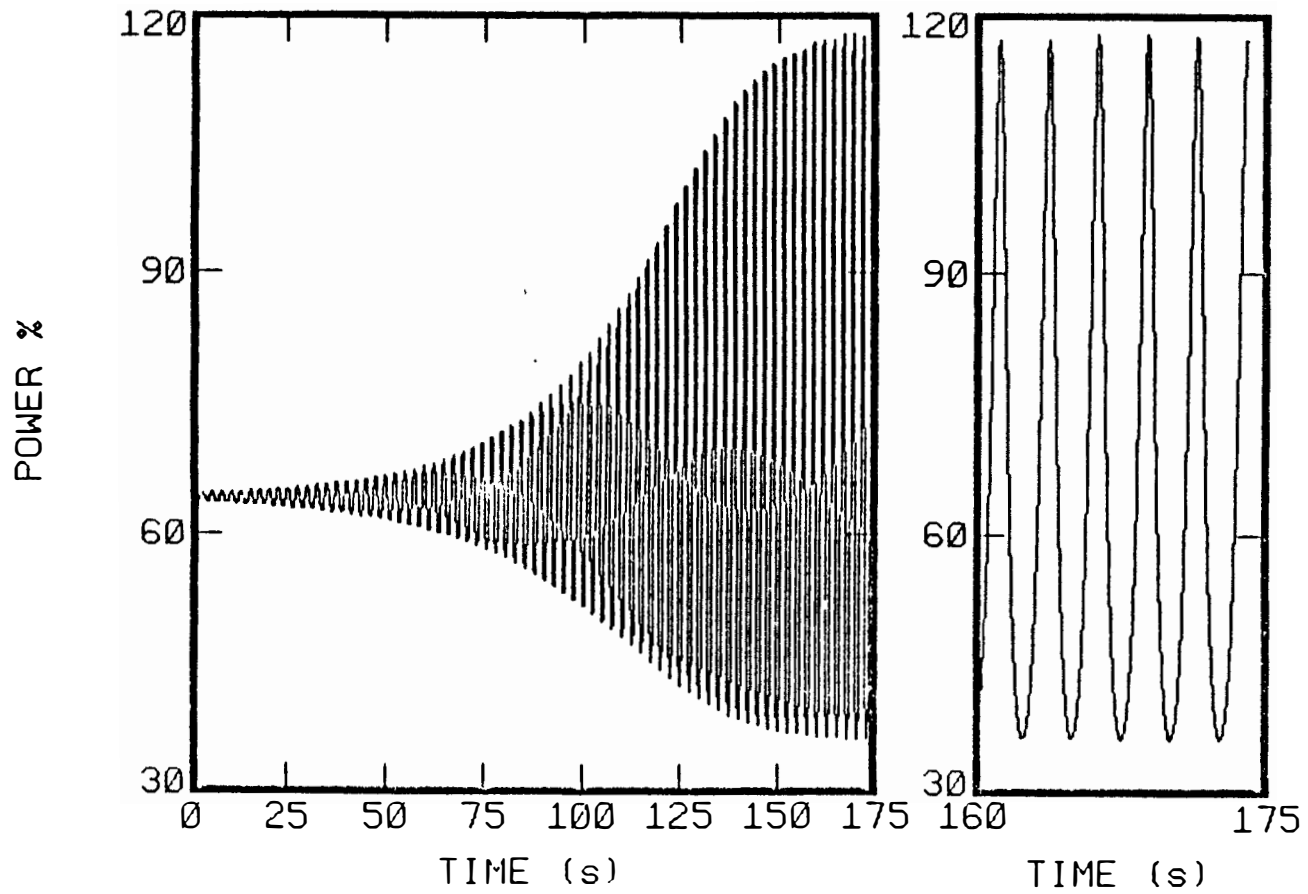
(a) Limit cycle development

(b) Detail of limit cycle oscillation

Figure 8.1   Development of a large amplitude limit cycle.

automatic scram on high power. These results show that large amplitude limit cycles are possible in BWRs. Figure 8.1b presents a detail of the oscillations. We can observe that they have the main characteristics found in the oscillations of the simpler model of Chapter 7; that is, the peaks are sharp and the valleys of the oscillation are smooth. This tends to indicate that, similar to the model in Chapter 7, the main nonlinearity causing the limit cycle phenomena is the parametric reactivity feedback in the point kinetics equation.

The oscillations around the equilibrium point of the main variables involved are presented in Figure 8.2. This figure corresponds to 64% power and 32% flow operating conditions once the limt cycle has been reached. We observe similarities between the results of this model and the ones in Chapter 7. The neutron time trace is formed by a series of sharp peaks followed by some valleys that are relatively flat. During the peaks, the fuel temperature rises and then slowly cools down by transferring energy to the coolant. At this moment of the cycle, the void fraction starts increasing until the convection cooling takes over and makes the void fraction oscillation negative. At this moment the reactivity feedback becomes positive and a new neutron pulse occurs. The oscillation in downcomer pressure seems to follow the void fraction with a phase lag. The mass flux follows the downcomer pressure. Since the upper plenum pressure is kept constant by the pressure regulator control system,
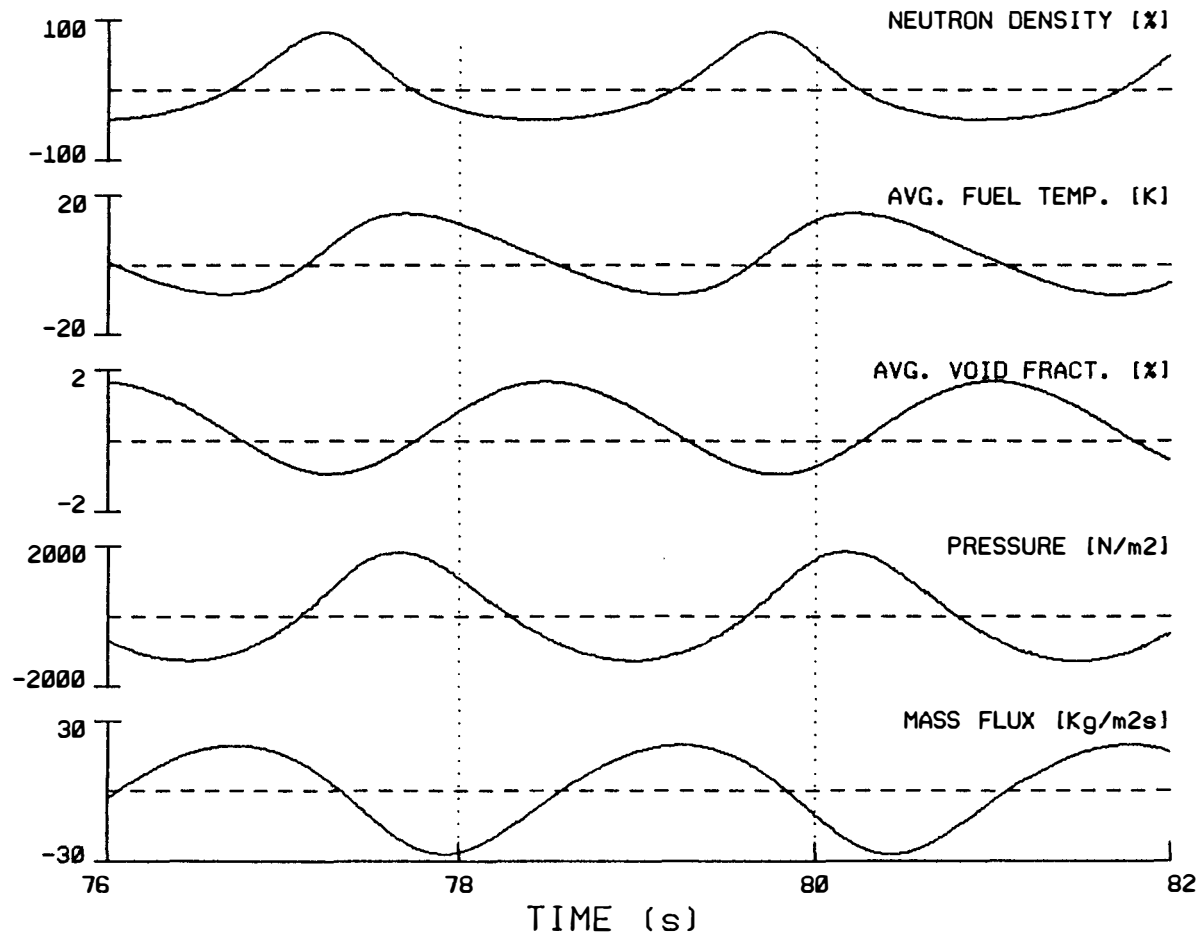
Figure 8.2 Time domain representation of the reactor limit cycle.

then if the lower plenum pressure is negative, the recirculation mass flux increases. Hence, we conclude that the average void fraction oscillation and the associated reactivity feedback is the driving source which determine the frequency of the pulses.

The phase space plots (Figure 8.3) show the limit cycle and the relation between variables in more detail. The plot of n versus dn/dt (Figure 8.3a) is very similar to the one in Chapter 7 as is the plot of n versus T (Figure 8.3b). The plot of n versus the average channel void fraction (Figure 8.3c) shows a high correlation of these two signals, which are in phase; however, contrary to the results of Chapter 7, the average α here is not the reactivity feedback, because the reactivity is weighted by the square of the power. This is the reason for the difference between this plot and the one in Chapter 7. The rest of the phase space plots in Figure 8.3 show the relationship between the different process signals. The most interesting of these relations are average void fraction versus pressure (Figure 8.3g) and mass flux versus pressure (Figure 8.3i). Here we see the nonlinear relationship between void fraction, mass flux, and pressure. If the relationship were linear, then the phase space plots would be perfect elipses.

8.3.2 Space Dependence of the Void Fraction Oscillations

The reactivity-type instability is also referred to in the literature as a density wave instability. Whenever there is a pulse in the power, a density perturbation is produced which travels upwards
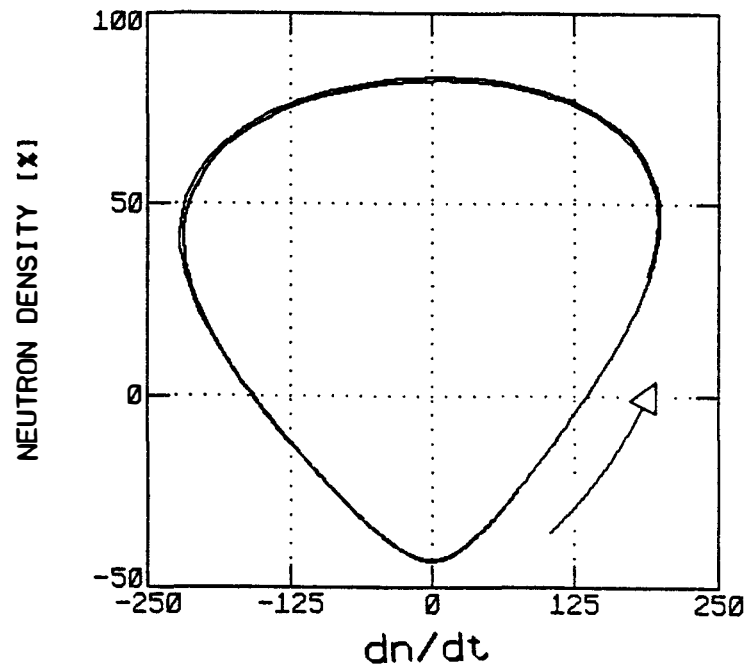
(a) Neutron density versus dn/dt
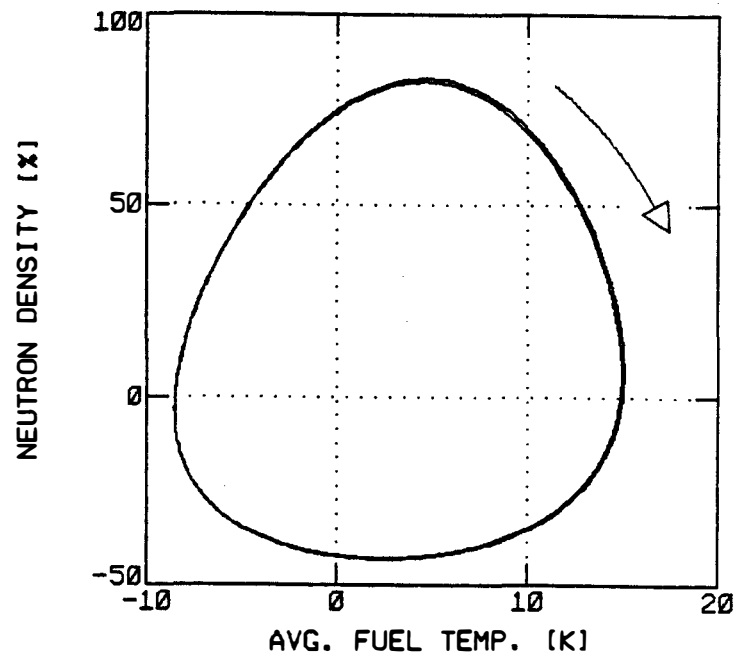
(b) Neutron density versus fuel temperature

Figure 8.3 Phase space representation of the reactor limit cycle.
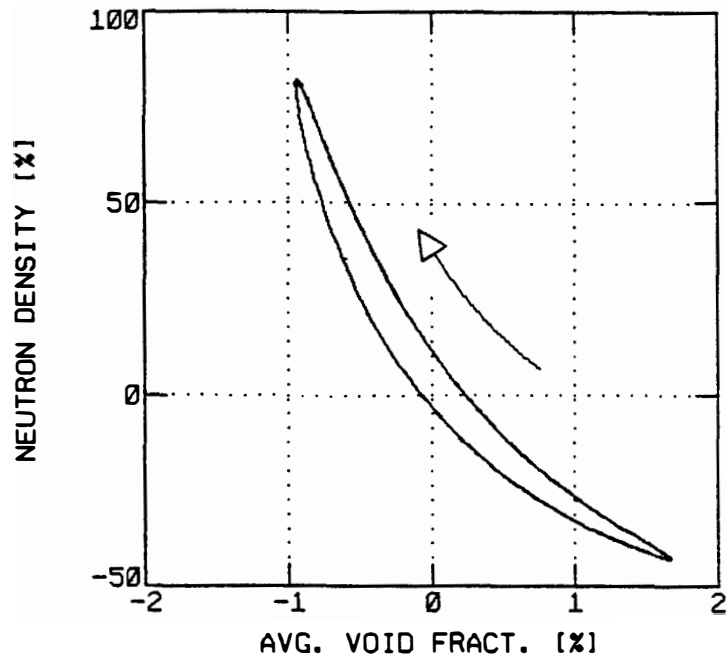
(c) Neutron density versus average void fraction
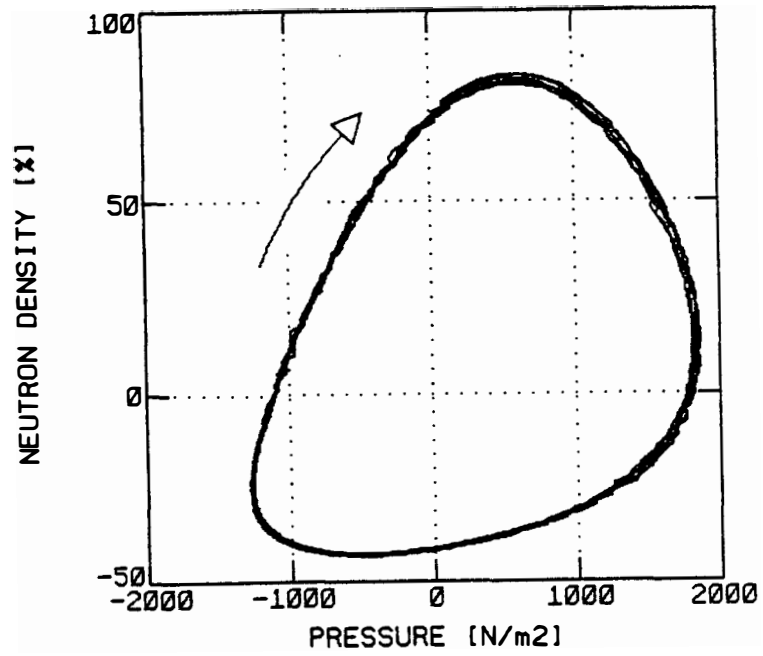
(d) Neutron density versus pressure drop

Figure 8.3 (continued).
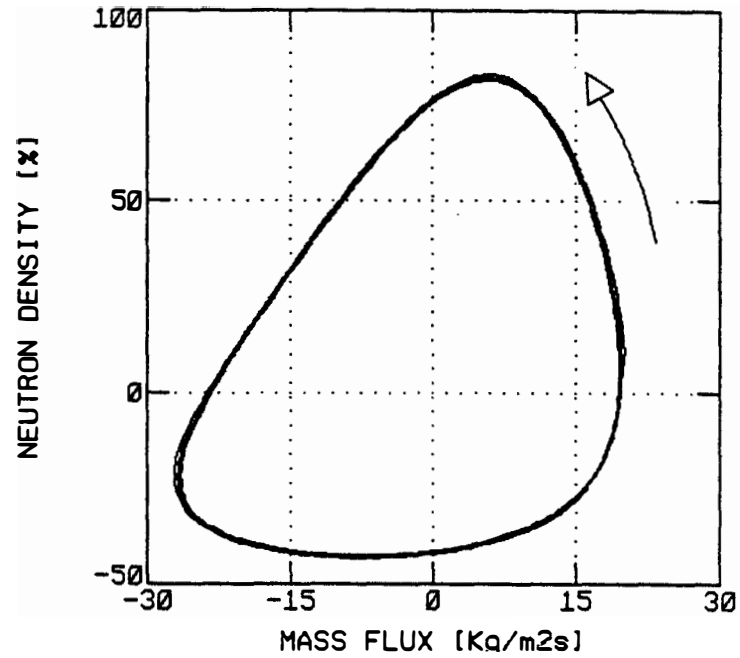
(e) Neutron density versus inlet mass flux
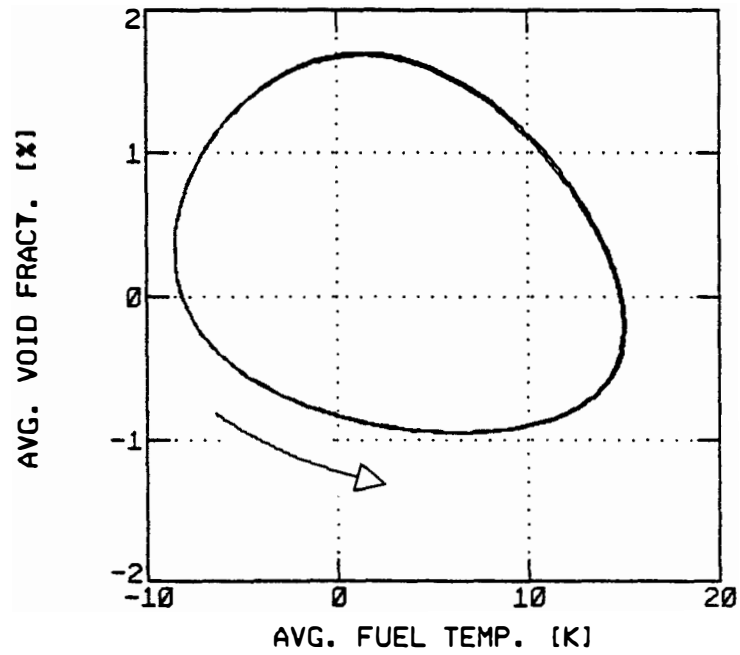
(f) Average void fraction versus fuel temperature

Figure 8.3 (continued).
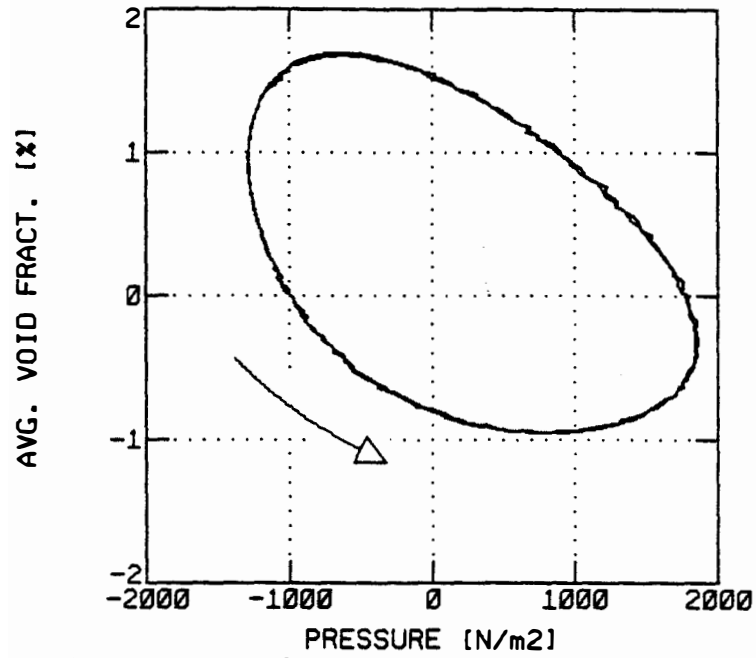
(g) Average void fraction versus pressure drop
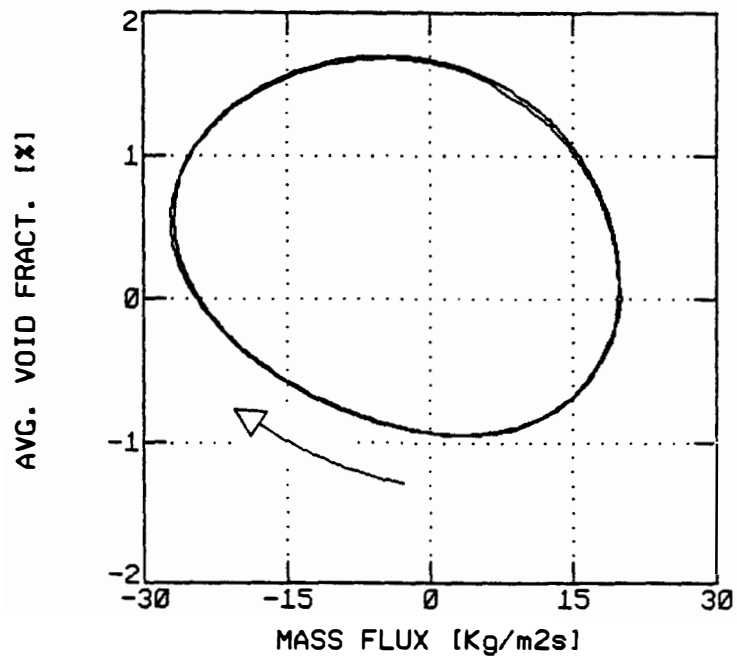
(h) Average void fraction versus inlet mass flux

Figure 8.3 (continued).
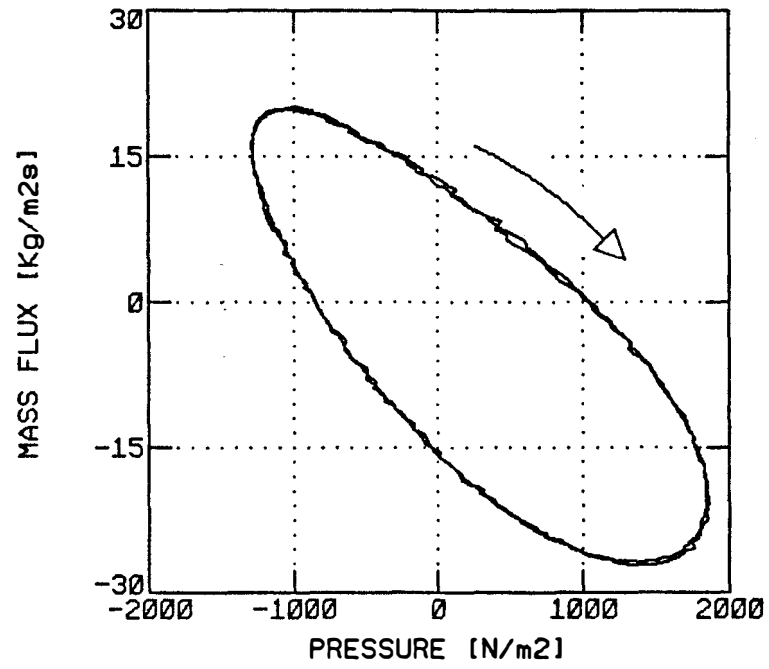
(i) Inlet mass flux versus pressure drop

Figure 8.3 (continued).

through the channel as a wave. We showed in Chapter 4 that the reactor instability is caused by the way in which this wave is weighted to obtain the global reactivity feedback.

A three-dimensional representation of the density wave as calculated by TLAP is presented in Figures 8.4a and b from two different perspectives. We observe that as a function of time, there is a buildup of voids in the lower part of the channels. The transport process of the density wave tilts the void fraction axial shape upwards until eventually the perturbation is eliminated and the process repeats itself. The evolution of the void fraction axial distribution can be seen more clearly in Figure 8.5, where the void shape is shown at various times during the oscillation.

8.3.3 Sensitivity to Operating Conditions

Customarily, the decay ratio is used to quantify stability; however, for any operating conditions in the nonlinear region the asymptotic decay ratio is by definition equal to 1.0 due to the appearance of limit cycles. An alternative parameter is therefore needed to describe the dynamic behavior of the reactor in this region. Our research indicates that the parameter best suited for this purpose is the amplitude of the oscillations.

Figure 8.6 shows the contours of constant decay ratio in the stable region (stable) and contours of constant oscillation amplitude in the nonlinear (unstable) region. This figure indicates several facts:
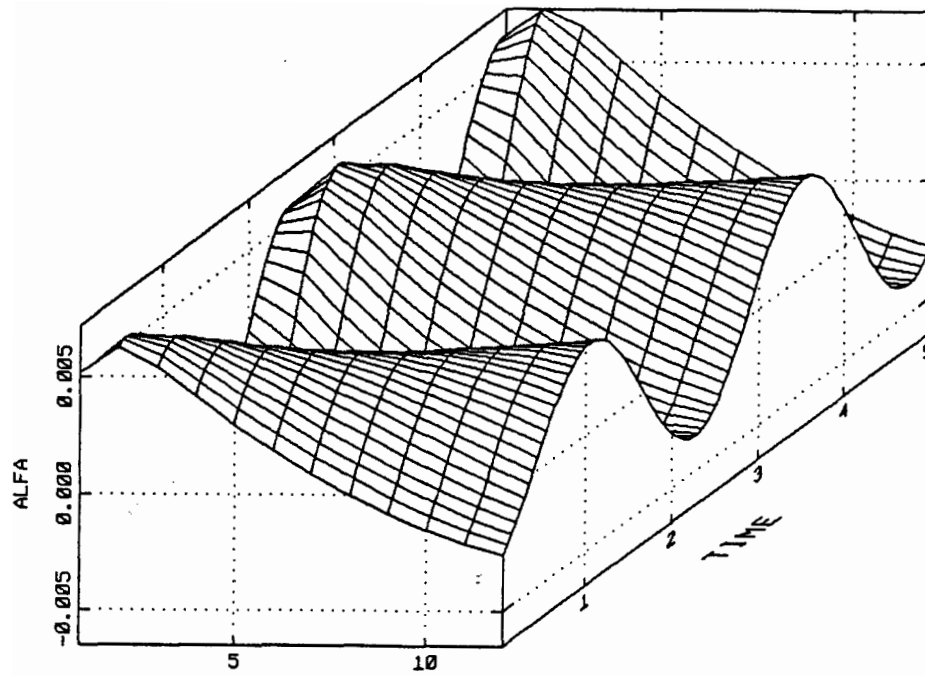
(a) Space dependence
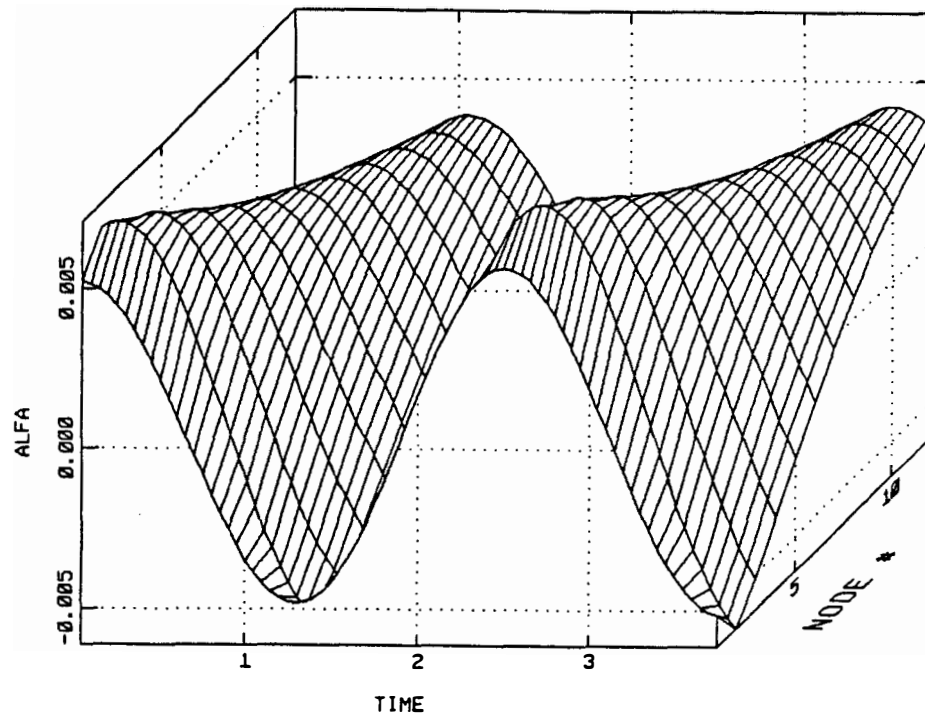
(b) Time dependence

Figure 8.4 Density wave during limit cycle oscillations.
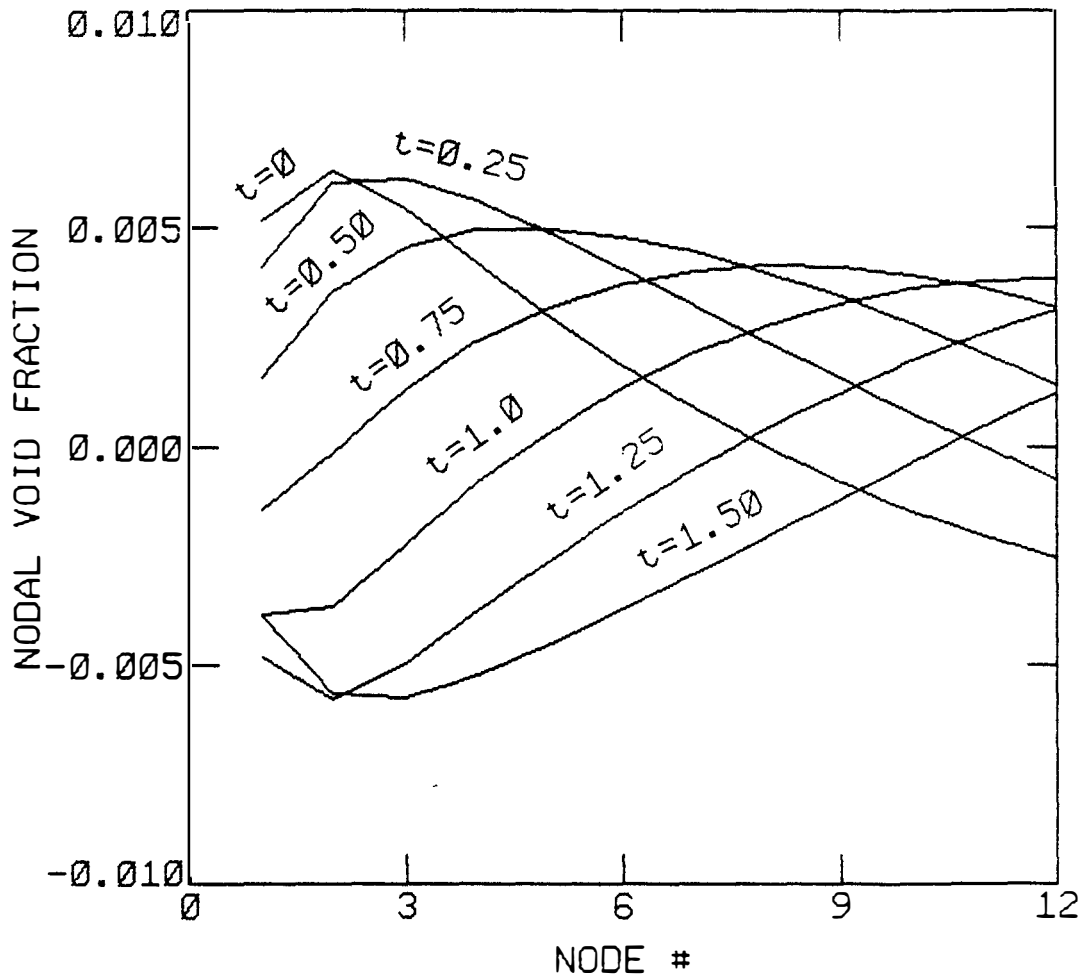
Figure 8.5 Variation in time of the space dependence of the density wave.
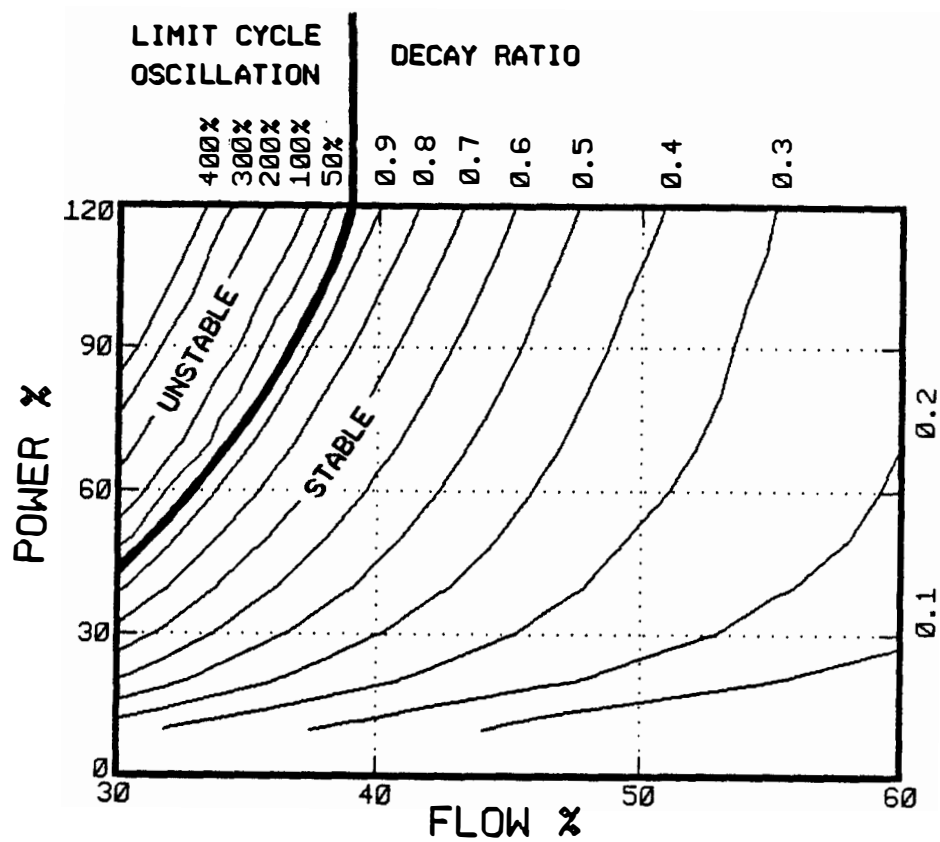
Figure 8.6 Sensitivity of limit cycle amplitude (unstable region)
            or decay ratio (stable region) to changes in
            operating conditions.

(a) The oscillation amplitude appears to behave in the nonlinear region in a similar manner than the decay ratio in the linear region.

(b) The decay ratio and oscillation amplitude are more sensitive to flow changes than to power changes.

(c) Large amplitude limit cycles are possible in nonlinear BWR operation.

(d) The limit cycle amplitude is very sensitive to changes in operating conditions.

The last point is better seen in Figure 8.7, which presents the oscillation amplitude as a function of power along the natural circulation line (32% flow). The limit cycle appears at 56% power and the oscillation reaches the 120% high power safety trip point at about 64% power.

8.3.4 Limit Cycle Stability

For all the conditions studied in section 8.3.3 (see figure 8.6) the calculated limit cycles were stable. When the power was increased further or the mass flux decreased, then the bifurcations and aperiodic behavior described in Chapter 7 were observed. Unfortunately, at the high power required for the bifurcations (for instance 120% power at 32% flow,) the large power oscillations caused the flow in the channel to be of saturated steam at some nodes during the high part of the oscillation. This effect produced in the present
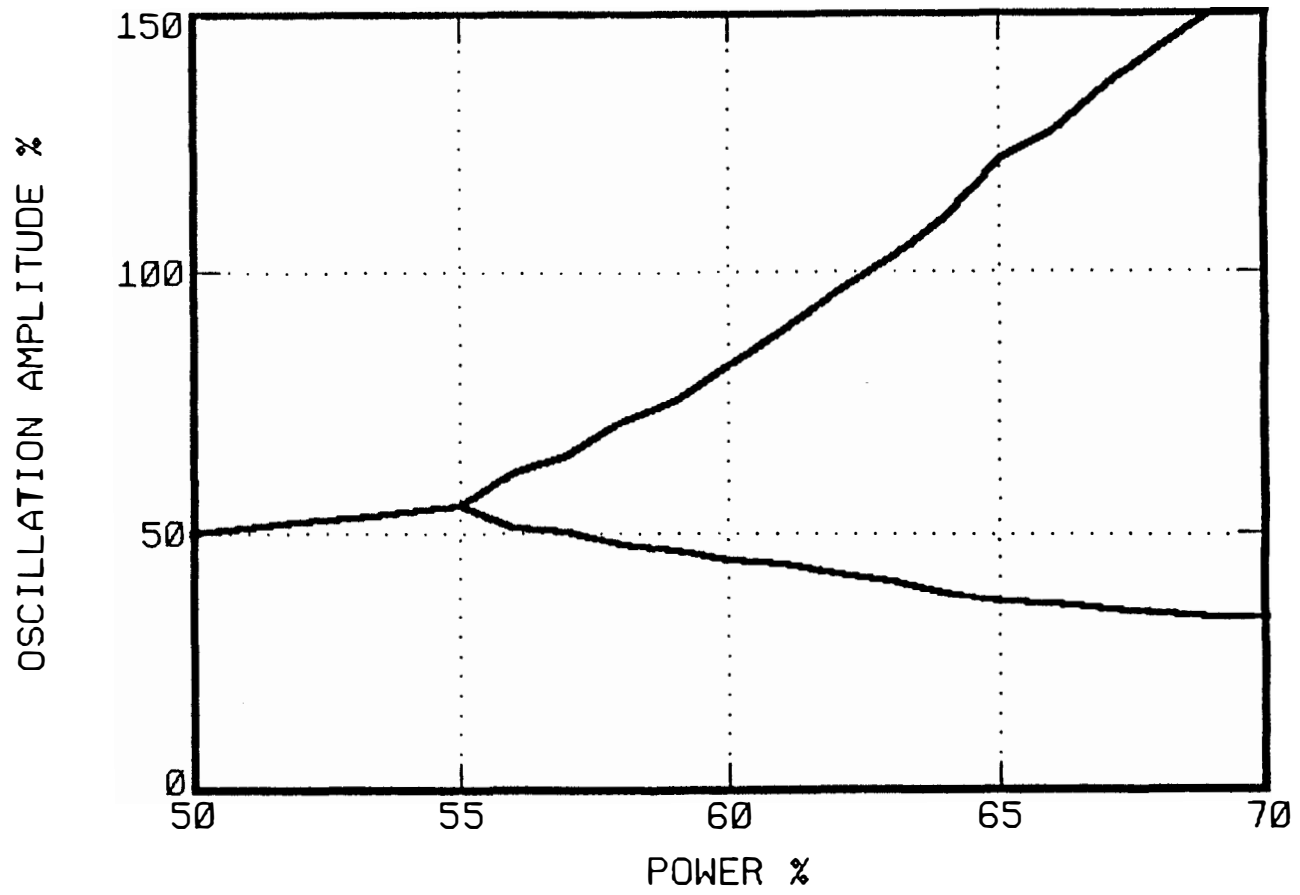
Figure 8.7   Sensitivity of limit cycle amplitude to power along the
             natural circulation line.

model that the steam quality be greater than 1.0, because the model in the present state can not handle supersaturated steam. For this reason, bifurcations and aperiodic behavior were not studied with the present model.

Nevertheless, the model results showed that for all probable modes of operation of a commercial BWR the limit cycles will be stable and the solutions will be periodic.

CHAPTER 9

CONCLUSIONS AND RECOMMENDATIONS

During the course of this research BWR dynamic behavior from
the linear and nonlinear points of view has been studied. Several
reduced order models have been developed as an aid for the iden-
tification of physical processes, which have been associated with
observable reactor features. As a result of this research an
understanding of the linear and nonlinear behavior of this type of
reactors has been obtained. In particular, the questions about what
to expect from nonlinearities in BWR operation and when to expect it
have been addressed. Finally, new methods have been developed for the
study of BWR stability in both the linear and nonlinear regimes. The
main accomplishements of this dissertation are highlighted in
Section 9.1. Recommendations for future work in this area are given
in Section 9.2

9.1 Accomplishments

The main accomplishments of this research can be summarized
as follows:

9.1.1 Features of the BWR Transfer Function

The pole-zero configuration of the reactivity-to-power
transfer function of BWRs has been studied. It has been determined
that an optimal model must contain three zeros and four poles to
properly represent the BWR dynamics. One of the zeros is real and

located at low frequency (about 0.03 Hz); the other two zeros are complex and have a frequency of approximately 0.5 Hz. Two of the poles are complex with a frequency in the vicinity of 0.5 Hz; the other two poles are real, one of them is located at a frequency of about 20 Hz and the other is close to the complex poles (about 0.3 Hz). Hence, an important conclusion of this work is that empirical fits to BWR reactivity-to-power transfer functions should contain at least three zeros and four poles. Note, however, that in the low frequency range (<10 Hz), a 3-zeros/3-poles model should suffice.

9.1.2 Association of Features with Physical Processes

The poles and zeros of the reactor transfer function have been associated with reactor dynamic processes:

(a) The low-frequency zero is directly related to the heat transfer process between fuel and coolant; it arises from a pole in this open loop feedback transfer function.

(b) The pair of complex zeros are related to the void reactivity feedback in the channel; they are caused by a combination of the cumulative effect of the axial void reactivity perturbations and the convection process with its associated density wave.

(c) The pair of complex poles are caused by the same mechanism as the complex zeros, as it can be concluded from the fact that their branch in the root locus diagram starts from this zeros.

(d) The high frequency pole arises from the β/Λ pole of the neutron field equations.

(e) Finally, the real pole at low frequency originates from the fuel dynamics, as its branch in the root locus starts at the low-frequency zero.

9.1.3 A Reduced Order Model for the Physical Processes

A reduced order linear model for BWR dynamics has been developed. This model has been used to study the sensitivity of transfer function features (i.e., poles and zeros) to changes in operating condition or variations in reactor parameters. This simple model accounts for the significant processes involved in the reactor dynamic behavior.

9.1.4 Linear BWR Stability Measurements

An automated technique has been developed and applied to determine the stability of commercial BWRs based on the analysis of their inherent power fluctuations (noise). This technique, thus, avoids the need of perturbative tests for this purpose.

This thechnique, which was the result of an improved understanding of the physical processes involved in the BWR dynamics, has the following unique features:

(a) It distinguishes between the asymptotic and apparent decay ratio; hence avoiding the pitfalls incurred by previous methods.

(b) It estimates the maximum decay ratio of the system, rather than the decay ratio of the pole with the smallest real part.

(c) It provides an estimation of the statistical error arising from the stochasticity of the noise data. Thus, allowing for a mean to check if enough data have been collected for the measurement.

(d) Furthermore, it supplies the user with a confidence level which defines the goodness of the estimate. This level is computed on the basis of a priori knowledge about the reactor dynamics and self-consistency checks.

9.1.5 <u>Importance of Nonlinearities in BWR Operation</u>

It has been shown that nonlinearities become important for BWR operation when the linear stability threshold is reached, which may occur at low flow and high power. This nonlinearities manifest themselves through the appearance of limit cycles.

9.1.6 <u>The Causative Mechanism Leading to the Appearance of the</u> <u>Limit Cycle</u>

It has been shown that the main nonlinearity causing the appearance of the limit cycle in BWRs is inherent to the neutronics equations, as it is caused by the parametric nature of the reactivity feedback (i.e., the ρn term in the point kinetics representation). An important consequence of this finding is that limit cycles will always appear even in cases when the reactor geometrical configuration has

been altered (for instance, as a result of a severe accident). Thus, the reactor response will always be bound.

9.1.7 The Limit Cycle in Phase Space

The study with the more complete linear model has shown the relationship between the process variables during limit cycle oscillations. It has been found that the void reactivity feedback defines the frequency of the limit cycle oscillation and that the rest of the process variables merely follow the reactivity feedback evolution in phase space.

9.1.8 Amplitude of the Limit Cycle Oscillation

It has been found that, although the oscillations are always bound, the amplitude of the limit cycle may reach values above the threshold for automatic scram. In addition, the sensitivity study performed in this research shows that the limit cycle amplitude is very sensitive to plant operating conditions. As a consequence, caution should be excercised in the operation of BWRs in the nonlinear regime to avoid unwanted scrams or excessive fuel temperature cycling.

9.1.9 Stability of the Limit Cycle

It has been found that the amplitude of the limit cycle might become unstable as the operating conditions are changed. This instability occurs in the amplitude of the limit cycle, which produces a doubling of the basic oscillation period. This research is the first published reference in nuclear reactors of this process, which is known in the literatue as a period-doubling pitchforck bifurcation.

9.1.10 Universality and Aperiodic Behavior in BWRs

As a parameter is changed, a cascade of period-doubling bifurcations occurs. The critical values of the parameters for which bifurcations occur have proven to converge geometrically to an accumulation point. This rate of convergence and the scaling of the bifurcations have been shown to satisfy Feigenbaum's universality theory. It is worthwhile to note that the same universal behavior has been found in such diverse systems as weather prediction algorithms, transition from laminar to turbulent flow, and many more physical processes.

As a consequence of the bifurcation process, aperiodic solutions of the deterministic reactor equations have been found for parameter values above the accumulation point. This result, besides its academic value, bears importance in the interpretation of the results of large numerical codes which might confuse the unsuspecting user with aperiodic solutions.

The results from the higher-order model, however, show that the limit cycle is stable, and the solution is periodic for reasonable ranges of power, flow, and pressure. That is, the period-doubling bifurcations and aperiodic region, although present, are in an extremely abnormal range of operating conditions in commercial BWRs. They can be found, though, in low-pressure reactors that have a larger void reactivity feedback.

9.1.11 Nonlinear Stochastic Phenomena

Nonlinear noise propagation in BWRs has been studied. A non-perturbative technique has been developed for detecting the onset of linear instability, and thus the transition to the nonlinear regime. This technique complements the linear stability measurement methodology developed using noise analysis, as it gives an independent and reliable evaluation for the limiting case in which the decay ratio is 1.0 (i.e., limit cycle conditions).

9.2 Recommendations for Further Research

During the course of this research we have come acros several interesting topics which we have not been able to pursue in more detail; these can be recommended as areas for future studies. Most of these topics are related to the nonlinear part of this research.

We showed in Chapter 7 that some BWR designs could sustain large power oscillations which resemble a series of sharp pulses. A neutronic machine could be designed for this purpose. This would be a self-pulsed reactor which could supply neutron pulses of large magnitude. These large pulses could be used for a range of purposes, from cross-section measurements to nuclear-pumped lasers. The pulse magnitudes could be maximized without fuel melting if the reactor operated at low pressure where the density reactivity coefficient is maximum.

Another area of further research is the study of local channel instabilities. As in the case of the reactivity instability, these

types of instabilities define a new area of reactor nonlinear operation, where local effects are of importance; for instance, a single channel might become unstable producing severe local damage, but the reactor protection system would not detect it because the average power is not affected in a significant amount. A study of these local nonlinearities could yield some techniques to identify and correct this abnormal operating condition.

LIST OF REFERENCES

LIST OF REFERENCES

1. "Power Reactors '77," <u>Nuclear Engineering International</u>-April 1977 Supplement, Vol 22 No.258 (1977).

2. Dietrich, J. R. and D. C. Layman, "Transient and Steady State Characteristics of a Boiling Water Reactor." The Borax Experiment AECD 3840 (1953).

3. Dietrich, J. R., "Experimental Determination of the Self-Regulation and Safety of Operating Water-Moderating Reactors." <u>Proceedings of the 1st U. N. International Conference of Peaceful Uses of Atomic Energy</u>, p. 481, Geneva, Switzerland (1955).

4. Dietrich, J. R., "Stability and Steam Voids in a Reactor." TID-7506, Part I (1956).

5. Skinner, R. E., and D. L. Hetrick, "The Transfer Function of a Water Boiler Reactor." NAA SR-1948 (1957).

6. Iriarte-Beauchamp, M., "Dynamic Behavior of Boiling Water reactors." Thesis, University of Michigan (1958).

7. Beckjord, E., "Dynamic Analysis of Natural Circulation BWRs." ANL-5799 (1958).

8. Beckjord, E. and S. Levy, "Hydraulic Instability in a Natural Circulation Loop with Net Steam Generation at 1000 psia." GEAP-3215 (1959).

9. Thie, J. A., "Dynamic Behavior of BWRs." ANL-5849 (1959).

10. Akcasu, A. Z., "Theoretical Feedback Analysis in BWRs." ANL-6221 (1960).

11. Akcasu, A. Z., "Mean Square Instability in Boiling Water Reactors." <u>Nuc. Sci. Eng.</u>, Vol 10, p. 337 (1961).

12. Christensen, H., "Power-to-Void Transfer Functions." ANL-6385 (1961).

13. Suda, N. and F. Rehbach, "Dynamic Analysis of the Operation of EBWR at 100 Mw." ANL-6478 (1961).

14. Garlid, K., H. R. Amudson, and H. S. Isbin, "A Theoretical Study of the Transient Operation and Stability of Two-Phase Natural Circulation Loops." ANL-6381 (1961).

15. Zivi, S. M. and R. W. Wright, "Flow void Effects on Power Void Transfer Functions and Hydrodynamic Oscillation in BWR." TID-7662 (1963).

16. Margolis, S. O. and S. Kaplan, "Transfer Functions for Boiling Reactor Stability Calculation." WAPD-T-1158 (1960).

17. Jones, A. B., "Hydrodynamic Stability of a Boiling Channel." KAPL-2170 (1961).

18. Jones, A. B. and D. G. Dight, "Hydrodynamic Stability of a Boiling Channel, Part 2." KAPL-2208 (1962).

19. Jones, A. B. and D. G. Dight, "Hydrodynamic Stability of a Boiling Channel, Part 3." KAPL-2290 (1963).

20. Jones, A. B., "Hydrodynamic Stability of a Boiling Channel, Part 4." KAPL-3070 (1964).

21. Jones, A. B. and W. M. Yarbrough, "Reactivity Stability of a Boiling Reactor." KAPL-3072 (1964).

22. Jones, A. B., "Reactivity Stability of a Boiling Reactor, Part 2." KAPL-3093 (1965).

23. Fleck, J. A., "The Dynamic Behavior of a Two-Phase Natural Circulation Loop." Journal of Nuclear Energy, Vol. 11-A, p.114 (1960).

24. Fleck, J. A., "The Influence of Pressure on Boiling Water Reactor Dynamic Behavior at Atmospheric Pressure." Nuc. Sci. Eng., Vol. 9, p.271 (1961).

25. Niemi, R. O. and J. C. Rawlingd, "Experimental Measurements of Design Parameter Effects on Void-Power Transfer Fucntion." GECR-4940 (1965).

26. Charmichael, L. A. and G. J. Scatena, "Stability and Dynamic Performance of the General Electric Boiling Water Reactor." APED-5652 (1969).

27. Linford, R. B., "Analytical Methods of Plant Transient Evaluation for the GE BWR." NEDO-10802 (1973).

28. Fisher, D. L. and J. E. Wood, "Stability and Dynamic Performance of the General Electric Boiling Water Reactor." NEDO-21506 (1977).

29. Lahey, R. T., Jr, The Thermal Hydraulics of a Boiling Water Nuclear Reactor. ANS Monograph (1977).

30. Podowski, M., R. P. Taleyakkhan, and R. T. Lahey, Jr., "Channel-to-Channel Instabilities in Parallel Channel Boiling Systems." Trans. Amer. Nucl. Soci., Vol 44, p. 383 (1983).

31. Park, G. C., M. Becker, R. T. Lahey, Jr., and M. Podowski, "Multichannel Stability Analysis of BWR Recirculating Loops." Trans. Amer. Nucl. Soci., Vol 44, p. 384 (1983).

32. Shen, C. N. and R. T. Lahey, Jr., "A State Variable Analysis of Boiling Water Reactor Stability and Response to Various External Perturbations." Trans. Amer. Nucl. Soci., Vol 44, p. 386 (1983).

33. Park, G. C., R. T. Lahey, Jr., M. Podowski, and M. Becker, "The Stability Analysis of BWR Recirculating Loop." Trans. Amer. Nucl. Soc., Vol 41 (1982).

34. Balaram, J., C. N. Shen, R. T. Lahey, Jr., and M. Becker, "An Analysis of Boiling Water Reactor Stability Margin." Rensselaer Polytechnic Institute. NUREG/CR-3291 (1983).

35. Murata, S. and O. Yokomizo, "Analysis of BWR Fuel Channel Instability Using a Three-Dimensional BWR Core Dynamics Computer Program." Trans. Amer. Nucl. Soci., Vol 45, p. 800 (1983).

36. Cobb, R. S. and R. S. Borland, "New BWR Thermal-Hydraulic Methodology in SIMULATE." Trans. Amer. Nucl. Soci., Vol 45, p. 801 (1983).

37. Yokomizo, O., T. Hayase, S. Murata, and H. Motoda, "Analysis of BWR Core Stability with Space-Dependent Neutron Feedback." Trans. Amer. Nucl. Soci., Vol 44, p. 560 (1983).

38. Holt, R. and J. Rasmussen, "RAMONA-II, A FORTRAN Code for Transient Analysis of Boiling Water Reactors." KR-147 (1972).

39. Rettig, W. H. et al., "RELAP4/MOD5 - A Computer Program for Transient Thermal-Hydraulic Analysis of Nuclear Reactors and Reload Systems User's Manual," ANCR-NUREG-1335, Idaho National Engineering Laboratory (1976).

40. Stewart, C. W. et al., "COBRA-IV: The Model and the Method." BNWL-2214, NRC-4 (1977).

41. Moore, K. V. et al., "RETRAN – A Program for One-Dimensional Transient Thermal-Hydraulic Analysis of Complex Fluid Flow Systems." EPRI NP-408 (1977).

42. Fennern, L. E. et al., "ODYSY: A Computer Program to Predict Stability of Boiling Water Reactors." Trans. Amer. Nucl. Soci., Vol 45, p. 727 (1983).

43. Tsunoyama, S., A. Tanabe, and S. A. Snadoz, "Vermont Yankee Stability Simulation with a Three-Dimensional Transient Model." Trans. Amer. Nucl. Soci., Vol 46, p. 752 (1984).

44. Pruitt, D. W., "COTRAN: A Two-Dimensional Reactor Kinetics Program with Reactivity Feedback for BWR Core Analysis." Exxon Nuclear Corporation, XN-NF-661 (1982).

45. Yokomizo, O., "Time-Domain Analysis of BWR Core Stability." J. of Nucl. Sci. and Tech., Vol 20, p. 63 (1983).

46. Lahey, R. T., Jr., and G. Yadigaroglu, "NUFREQ, A Computer Program to Investigate Thermo-Hydraulic Stability." General Electric Company. NEDO-13344 (1973).

47. Park, G. C., M. Podowski, M. Becker, and R. T. Lahey, Jr, "The Development of NUFREQ-N, An Analytical Model for the Stability Analysis of Nuclear Coupled Density-Wave Oscillations in Boiling Water reactors." Rensselaer Polytechnic Institute. NUREG/CR-3375 (1983).

48. Otaduy, P. J., "Modeling of the Dynamic Behavior of Large Boiling Water Reactor Cores." Ph. D. Dissertation, University of Florida, (1979).

49. Waddell, T. A., T. A. Keys, and G. W. Smith, "Development of a BWR Stability Analysis Methodology at TVA." Trans. Amer. Nucl. Soci., Vol 46, p. 754 (1984).

50. Hodde, J. A., "Control Rod Oscillation and Transient Pressure Tests, Big Rock Point BWR." GEAP-4448 (1966).

51. Howard, C. L., "Development Program on the Garigliano Nuclear Reactor." Quarterly Report No 18 GEAP-5473 (1967).

52. Niemi, R. O., "Dresden-3 Rod Oscillator Tests." NEDM-13381 (1973).

53. Larsen, N. H., "Core Design and Operating Data for Cycles 1 and 2 of Peach Bottom 2." EPRI NP-563 (1978).

54. Carmichael, L. A. and R. O. Niemi, "Transient and Stability Tests at Peach Bottom Atomic Power Station Unit 2 at End of Cycle 2." EPRI NP-564 (1978).

55. Holloway, G. L. and L. M. Shiraishi, "Core Design and Operating Data for Cycle 3 of Peach Bottom 2." EPRI NP-971 (1981).

56. Woffinden, F. B. and R. O. Niemi, "Low-Flow Stability Tests at Peach Bottom Atomic Power Station Unit 2 During Cycle 3." EPRI NP-972 (1981).

57. Waaranpera, Y. and S. Anderson, "BWR Stability Testing: Reaching the Limit Cycle Threshold at Natural Circulation." Trans. Am. Nucl. Soc.,Vol 39, p. 868 (1981).

58. Sandoz, S. A. and S. F. Chen, "Vermont Yankee Stability Tests During Cycle 8." Trans. Am. Nucl. Soc., Vol 45, p. 727 (1983).

59. March-Leuba, J., D. N. Fry, M. E. Buchannan, and C. O. McNew, "Evaluation of the Effect of a New 9×9 Fuel Design on Local Stability in Dresden 2." Trans. Amer. Nucl. Soci., Vol 46, p. 746 (1984).

60. Hornyick, K. and J. A. Naser, "RETRAN Analysis of the Turbine Trip Tests at Peach Bottom Atomic Power Station, Unit 2, at the End of Cycle 2." EPRI NP-1076-SR (1979).

61. Hornyick, K., "Special RETRAN Modeling Studies Performed in Conjuction with the Peach Bottom-2 Transient Test Analysis." EPRI NP-1842 (1981).

62. Congdon, S. P. and R. B. Linford, "Peach Bottom Turbine Trip Simulation with a One-Dimensional Transient Model." Trans. Amer. Nucl. Soc., Vol 30, p. 217 (1978).

63. March-Leuba, J. and P. J. Otaduy, "A Comparison of BWR Stability Measurements with Calculations Using the Code LAPUR-IV." NUREG/CR-2998, ORNL/TM-8546 (1983).

64. Keys, T. A., "Validation of BWR Core Methodology at TVA." Trans. Amer. Nucl. Soc., Vol 45, p. 736 (1984).

65. Uhrig, R. E., Random Noise Techniques in Nuclear Reactor Systems. New York: Ronald Press Co. (1970).

66. Bendat, J. S. and A. G. Piersol, Random Data: Analysis and Measurements Procedures. New York: Wiley-Interscience (1971).

67. Williams, M.M.R., Random Processes in Nuclear Reactors. Oxford, England: Pergamon Press (1974).

68. Thie, J. A., Power Reactor Noise. LaGrande Park, Ill. American Nuclear Society Monograph (1981).

69. Williams, M.M.R., Reactor Noise-SMORN-II. Progress in Nuclear Energy, Vol 1. Pergamon Press (1977).

70. Williams, M.M.R., Reactor Noise-SMORN-III. Progress in Nuclear Energy, Vol 9. Pergamon Press (1982).

71. Upadhyaya, B. R. and M. Kitamura, "Monitoring BWR Stability Using Time Series Analysis of Neutron Noise." Trans. Amer. Nucl. Soci., Vol 33, p. 342 (1979).

72. Kosaly, G., "Noise Investigations in Boiling Water and Pressurized Water Reactors." Prog. Nucl. Energy, Vol 5, p. 145 (1980).

73. Sweeney, F. J., "A Theoretical Model of Boiling Water Reactor Neutron Noise." Ph. D. Dissertation, The University of Tennessee, Knoxville, Tennessee (1980).

74. Tsunoyama et al., "Peach Bottom-2 Reactor Stability Estimation Using PRBS and Noise Data." Trans. Amer. Nucl. Soci., Vol 35, p. 590 (1980).

75. Upadhyaya, B. R. and M. Kitamura, "Stability Monitoring of Boiling Water Reactors by Time Series Analysis of Neutron Noise." Nucl. Sci. Eng., Vol 77, p. 480 (1981).

76. Wu, S. M. et al., "Stochastic Modeling of On-Line BWR Stability." EPRI NP-1753 (1981).

77. March-Leuba, J., F. J. Sweeney, W. T. King, J. A. Renier, R.T. Wood, and D. N. Fry, "In-Core Flow Velocity Profiles During the First Cycle of Hatch-1 Inferred from Neutron Noise." EPRI NP-2083 (1981).

78. Difilipo, F. C., "Neutron Wave Propagation in Heterogeneous Media and the Interpretation of Neutron Noise in Boiling Water Reactors." Nucl. Sci. Eng., Vol 80, p. 211 (1982).

79. Fry, D. N., J. March–Leuba, and F. J. Sweeney, "Neutron Noise in Light Water Reactors." NUREG/CR–3303, ORNL/TM–8774 (1984).

80. Ott, E., "Strange Attractors and Chaotic Motions of Dynamical Systems." Review of Modern Physics, Vol 53, NO 4, Part I, p. 655 (1981).

81. Eckmann, J.–P., "Roads to Turbulence in Dissipative Dynamical Systems." Review of Modern Physics, Vol 53, NO 4, Part I, p. 643 (1981).

82. Feigenbaum, M. J., "The Transition to Aperiodic Behavior in Turbulent Systems." Commun. Math. Phys., Vol 77, p. 65 (1980).

83. Feigenbaum, M. J., "Universal Behavior in Non–Linear Systems." Los Alamos Science. Los Alamos Scientific Laboratory, (1980).

84. Huberman, B. A. and A. B. Zisook, "Power Spectra of Strange Attractors." Physical Review Letters, Vol 46, No. 10, p. 626 (1981).

85. Jensen, R. V. and C. R. Oberman, "Calculation of the Statistical Properties of Strange Attractors." Physical Review Letters, Vol 46, No. 14, p. 1547 (1981).

86. Geisel, T. and J. Nierwetberg, "Universal Fine Structure of the Chaotic Region in Period–Doubling Systems." Physical Review Letters, Vol 47, No. 14, p. 975 (1981).

87. Linsay, P. S., "Period Doubling and Chaotic Behavior in a Driven Anharmonic Oscillator." Physical Review Letters, Vol 47, No. 19, p. 1349 (1981).

88. Geisel, T. and J. Nierwetberg, "Onset of Diffussion and Universal Scaling in Chaotic Systems." Physical Review Letters, Vol 48, No. 1, p. 7 (1982).

89. Testa, J., J. Perez, and C. Jeffries, "Evidence for Universal Chaotic Behavior of a Driven Nonlinear Oscillator." Physical Review Letters, Vol 48, No. 11, p. 714 (1982).

90. Ott, E. and J. A. Yorke, "Chaotic Attractors in Crisis." Physical Review Letters, Vol 48, No. 22, p. 1507 (1982).

91. Gonzalez, D. L. and O. Piro, "Chaos in a Nonlinear Driven Oscillator with Exact Solution." Physical Review Letters, Vol 50, No. 12, p. 870 (1982).

92. Grassberger, P. and I. Procaccia, "Characterization of Strange Attractors." Physical Review Letters, Vol 50, No. 5, p. 346 (1983).

93. Kitanno, M., T. Yabuzaki, and T. Ogawa, "Chaos and Period-Doubling Bifurcations in a Simple Acoustic System." Physical Review Letters, Vol 50, No. 10, p. 713 (1983).

94. March-Leuba, J. and R. B. Perez, "A Physical Model of Non-Linear Noise with Application to BWR Stability." Trans. Amer. Nucl. Soci., Vol 44, p. 523 (1983).

95. March-Leuba, J., D. G. Cacuci, and R. B. Perez, "Non-Linear Dynamics of Boiling Water Reactors." Trans. Amer. Nucl. Soci., Vol 45, p. 725 (1983).

96. March-Leuba, J., D. G. Cacuci, and R. B. Perez, "Universality and Aperiodic Behavior of Nuclear Reactors." Nuc. Sci. Eng., Vol 86, p. 401 (1984).

97. March-Leuba, J., R. B. Perez, and D. G. Cacuci, "Calculation of Limit Cycle Amplitudes in Commercial Boiling Water Reactors." Trans. Amer. Nucl. Soci., Vol 46, p. 749 (1984).

98. Joseph, D. D., Stability of Fluid Motions II. Springer Tracts in Natural Philosophy, Vol 27. Springer-Verlag, New York (1979).

99. Kuo, B. C., Automatic Control Systems. Prentice-Hall, Englewood Cliffs, New Jersey (1975).

100. Hsu, Y. and R. W. Graham, Transport Processes in Boiling and Two-Phase Systems. McGraw Hill Book Co., New York (1976).

101. Smith, O. J. H., Feedback Control Systems. McGraw Hill Book Co., New York (1958).

102. Baker, G. A., Essentials of Pade Approximants. Academic Press, New York (1975).

103. Smith, C. M., "A Description of the Hardware and Software of the Power Spectral Density Recognition (PSDREC) Continuous On-Line Reactor Surveillance System (California Distribution)," Vol 2. NUREG/CR-3439, ORNL/TM-8862 (1983).

104. Akaike, H., IEE Trans. Autom. Control, Vol 19, p. 716 (1974).

105. March-Leuba, J., D. N. Fry, M. E. Buchanan, and C. O. McNew, "Local Stability Tests in Dresden 2 Boiling Water Reactor." ORNL/TM-9054 (1984).

106. Hindmarsh, A. C., "LSODE and LSODI, Two New Initial Value Ordinary Differential Equation Solvers." ACM-Signum Newsletter, Vol 15, 4, p. 10 (1980).

107. Bell, G. I., Glasstone, S., Nuclear Reactor Theory. Van Nostrand Reinhold Co., New York (1970).

108. Box, G. E. P. and G. H. Jenkins, Time Series Analysis: Forecasting and Control. Holden-Day, San Francisco (1970).

109. Perez, R. B., NE 6120: Selected Topics in Reactor Theory Course Notes. The University of Tennessee (1982).

110. Otaduy, P. J. and J. March-Leuba, "Results of the Dresden Stability Calculations." Letter report to Mr. G. A. Schwenk, U.S. Nuclear Regulatory Commission, March 29 (1983).

APPENDICES

APPENDIX A

SOME PROPERTIES OF AUTOREGRESSIVE MODELS

APPENDIX A

SOME PROPERTIES OF AUTOREGRESSIVE MODELS

Autoregressive or AR models have been widely used previously. There are many publications describing this methodology in detail; see for example References 75 or 108. In this appendix we present some properties of univariate AR models which are relevant to the subject of stability estimation from noise analysis. To the knowledge of the author, most of these properties have not been published before.

A univariate AR model of order N is of the form

$$x(t) = \sum_{k=1}^{N} A_k x(t-k\Delta t) + v(t) \quad , \tag{A-1}$$

where

. $x(t)$ = Sampled signal

A_k = Model parameters

N = Model order

$v(t)$ = Residual sequence

Once the model is known, the residual sequence can be calculated from the expression

$$v(t) = x(t) - \sum_{k=1}^{N} A(k)x(t-k\Delta t) \quad . \tag{A-2}$$

Having M samples of the signal, $x(t)$, we obtain the model parameters by minimizing the variance of the residual. Thus, we minimize the function

$$J = \frac{1}{M} \sum_{k=1}^{N} \left[x(t) - \sum A(k)x(t-k\Delta t) \right]^2 \quad . \tag{A-3}$$

233

Taking derivatives with respect to the parameters and equating them to zero, we obtain the celebrated Yule-Walker equations[108]

$$\sum_{K=1}^{N} A_k C(k-1) = C(1) \quad ; \quad 1=1,2,\ldots,N \quad , \tag{A-4}$$

where the correlations, $C(1)$, can be approximated by the expression

$$C(1) = \frac{1}{M} \sum_{k=1}^{N} x(t)x(t-1\Delta t) \quad . \tag{A-5}$$

A.1 The Impulse Response

Let us turn our attention now to the role of the residual sequence. In the Fourier domain, the original sequence is given by

$$X(\omega) = G(\omega)V(\omega) \quad , \tag{A-6}$$

where $G(\omega)$ is the AR model transfer function. The cross-correlation, $XC(\tau)$, between $x(t)$ and $v(t)$ is, therefore

$$XC(\tau) = F^{-1}\left[V^*(\omega)X(\omega)\right] = F^{-1}\left[V^*(\omega)G(\omega)V(\omega)\right] \quad , \tag{A-7}$$

where F^{-1} stands for inverse Fourier transform. On the other side, from equation (A-2), we have

$$XC(1\Delta t) = C(1) - \sum_{k=1}^{N} A(k)C(1+k) \quad . \tag{A-8}$$

And therefore, considering equations (A-4) and (A-8), we see that the cross-correlation is zero for negative lags, 1, but nonzero for positive or zero lags. In terms of equation (A-7), we see that in order for XC to be zero for $1>0$, $V(\omega)$ must not have any poles; otherwise, $V^*(\omega)$ would introduce some poles for the negative-time integral path

in the inverse Fourier transform. Thus, we can conclude that $V(\omega)$ must be a white noise. As a consequence,

$$XC(\tau) = F^{-1}[G(\omega)] \quad , \tag{A-9}$$

but the right-hand side is the impulse response of $G(\omega)$. Thus, the impulse response can be calculated as

$$h(1) = C(1) - \sum_{k=1}^{N} A(k)C(1+k) \quad . \tag{A-10}$$

A.2 Pole-Zero Configuration

Note that from equation (A-10) the impulse response is not equal to zero at zero lag. Note also that, given the form of $h(t)$ (i. e., zero for negative times), Fourier and Laplace transform are equivalent in this case. Thus, we can apply the final value theorem, which states that

$$\lim_{t \to 0} h(t) = \lim_{s \to \infty} sG(s) \quad , \tag{A-11}$$

and conclude that for $h(t)$ to be finite and nonzero at $t = 0$, the model transfer function, $G(s)$, must have one zero less than poles. This is, the order of $G(s)$ must be 0-zeros/1-poles, 1-zeros/2-poles,...

A.3 AR-Consistent Autocorrelation

Systems in general do not have the pole-zero configuration necessary for the AR model to be an exact representation. For this reason, AR models of large order must be used in this cases, but the

fit is never perfect because the Yule-Walker equations are satisfied
only up to lag N and, thus

$$C(1) \neq \sum_{k=1}^{N} A(k)C(k-1) \qquad ; \ 1 > N \quad . \qquad \text{(A-12)}$$

However, once the model order is chosen, we can define an
AR-consistent autocorrelation, $C'(1)$, which satisfies this equation

$$C'(1) = C(1) \qquad ; \ 1 = 0,1,\ldots,N \qquad \text{(A-13)}$$

$$C'(1) = \sum_{k=1}^{N} A(k)C'(k-1) \qquad ; \ 1 > N \quad . \qquad \text{(A-14)}$$

This autocorrelation defines an imaginary system with the
same pole-zero configuration as the AR model.

APPENDIX B

DETAILED DESCRIPTION OF THE NOISE ANALYSIS ALGORITHM

TO MEASURE BWR STABILITY

APPENDIX B

DETAILED DESCRIPTION OF THE NOISE ANALYSIS ALGORITHM

TO MEASURE BWR STABILITY

This appendix describes a series of FORTRAN-77 subroutines
that are used to evaluate the asymptotic decay ratio and frequency
of oscillation from a noise recording of an average power range
monitor (APRM) signal from a boiling water reactor (BWR). The
subroutines were originally developed as part of a diagnostics
package to be included in the PSDREC[103] system, but they can be used
as part of a general stability evaluation code.

The technique used to evaluate the stability of the APRM
signal is autoregressive (AR) modeling (see Appendix A). Subroutine
ARMODL fits an optimal AR model order of the form

$$x(i) = \sum_{k=1}^{N} A_k x(i-k) + v(i) \quad , \qquad (B-1)$$

where

$x(i)$ = sampled signal

N = model order

A_k = model parameters

$v(i)$ = residual noise

This subroutine selects the model order so that the likeli-
hood function (i. e., the joint probability of all the measurements)
is a maximum. Akaike[104] showed that this is equivalent to mini-
mizing the following function

238

$$AIC = M \ln(\sigma^2/C(0)) + 2N \quad , \qquad\qquad (B-2)$$

where σ is the residual noise variance. This function is often
called the Akaike's information criterium.

Once the model parameters, A_k, are determined, the impulse
response, $h(i)$, can be calculated as an initial value problem

$$h(i) = \sum_{k=1}^{N} A_k h(i-k) \quad , \qquad\qquad (B-3)$$

with boundary conditions

$$h(0) = 1. \qquad\qquad\qquad (B-4)$$

$$h(-i) = 0. \qquad ; \ i=1,2,..,N,.. \qquad\qquad (B-5)$$

The frequency domain equivalent of the AR model can be
obtained by Laplace transforming equation (B-1)

$$x(s) = \frac{\sigma^2}{1 - \sum_{k=1}^{N} A_k D^k} \quad , \qquad\qquad (B-6)$$

where D is the backshift operator

$$D = e^{-s\Delta t} \qquad\qquad\qquad (B-7)$$

Note that in Equation (B-6) s is not restricted to the
imaginary axis, and the equation can be evaluated at any point of
the s-plane. In this way, a search for the poles of $x(s)$ can be
performed.

The stability of the signal is quatified in terms of a decay
ratio (DR) and a natural frequency of oscillation (NF); both of these
parameters can be evaluated from the model. This code evaluates the
DR in three different ways: First, it measures the apparent DR of the
autocorrelation function; then, it estimates the asymptotic DR of the

model's impulse response; finally, it attemps to find the most unstable complex pole from the frequency domain representation of the AR model. The three estimates are evaluated, and a most probable DR and NF are selected along with a confidence level.

B.1 Apparent Decay Ratio of the Autocorrelation Function

The autocorrelation function is estimated in subroutine CORREL as the inverse fast Fourier transform (FFT) of the power spectral density (PSD) of the APRM signal. This estimate coincides with the true correlation as the number of samples, M, tend to infinity, and in general, for finite M, it is a good estimate. The PSD supplied should be the best available estimate of the true PSD of the signal and, therefore, Hann windowing[66] is recommended.

The apparent DR is directly measured in the correlation and the frequency of oscillation is obtained as the inverse of the time at which the first maximum occurs. This decay ratio is a very robust measure of the reactor stability, in the sense that it is well defined and that as the reactor approaches the stability threshold this DR tends to 1.0 (which is not necesarily true for the apparent DR of other functions like impulse or step responses). All this is accomplished in subroutine DRCORR, which returns three error conditions:

(a) IERC=-1 : No maximum was found. This could be due to a low decay ratio (DR<0.1) or to a big DC component in the signal.

(b) IERC=-2 : The first minimum (or valley) in the correlation

has a positive value. This could be due to the presence of low-frequency oscillations.

(c) IERC=-3 : The first maximum (or peak) in the correlation does not have the highest value of all the peaks in the correlation (excluding zero-lag). This is also probably due to low frequency oscillations.

The importance of determining the existance of low-frequency oscillation can not be underestimated. It has been observed[105] that some BWRs when operated at reduced flows exhibit a low-frequency oscillation (about 0.07 Hz) of almost pure-sine-wave type (i.e. decay ratio close to 1.0). This has been attributed to a poor tuning of the pressure control system. When these oscillations are present, most of the methods to determine the reactor stability will yield the stability of the oscillations that are introduced by the control system and not by the reactor thermal hydraulics. Once their presence has been determined in these subroutine, corrective action can be taken.

B.2 Asymptotic Decay Ratio of the Impulse Response

The impulse response is calculated using Equation (A-6) in subroutine DRTIME. The DR is evaluated directly and a check for convergence to the asymptotic value is performed in subroutine CONVRG.

The DR is measured using the formula

$$DR = \left(\frac{x_3 - x_2}{x_1 - x_2}\right)^2 \tag{B-8}$$

where x_1, x_2, and x_3 are three consecutive extrema (maxima or minima) of the impulse response. This formula explicitly takes into account the possibility of low-frequency (or DC) interferences.

The main problem with this approach to evaluate the DR is that this subroutine returns the DR of the complex pair of poles with the minimum real part, and not the maximum DR. For instance, if the system had two poles, one at s = -0.044±0.4i Hz with DR = 0.5 and another at s = -0.026±0.07i Hz with DR=0.1, DRTIME will return DR=0.1 and NF=0.07 Hz. This kind of error has to be considered in subroutine CONFID.

Three error conditions are returned by DRTIME

(a) IERS=-1 : DR did not converge, but the impulse response was within roundoff bounds. This is probably due to either a very low DR or the interaction of two pairs of poles with the same real part but different frequencies. An estimate of DR and NF is returned from the time it took to reach the roundoff level.

(b) IERS=-2 : DR did not converge in 3000 steps, but the impulse response was not yet within roundoff bounds. This is also due to the interaction between two poles, but the DR is high, since the impulse response magnitude did not decrease enough.

(c) IERS=-3 : No peaks were found in the impulse response. This is probably due to a signal that does not oscillate (i.e. DR=0).

B.3 Frequency Domain Pole Search

Subroutine DRFREQ executes a search for poles of equation
(B-6), the frequency domain representation of the AR model, in the
frequency range were the characteristic BWR oscillation is expected
(i. e., 0.25 to 0.85 Hz). The search is executed for the zeros of the
denominator, which is a polynomial in D. For this purpose, Newton's
method in the complex domain is used: given a starting guess, D_0,
the next estimate of the positon of the pole is

$$D_1 = D_0 + \frac{P(D_0)}{\frac{dP}{dD}\Big|_{D=D_0}} \quad , \qquad (B-9)$$

where

$$P(D) = 1 - \sum_{k=1}^{N} A_k \, D^k \quad . \qquad (B-10)$$

A convergence check is performed after each step along with
step size control if necessary.

DRFREQ uses three starting guesses, D_0, that correspond to
the maximum value of the PSD, the maximum value of the derivative of
the phase of Equation (B-6) with respect to frequency and finally, the
maximum value of the derivative of the phase close to the maximum
value of the PSD.

Three error conditions are returned:

(a) <u>IERP=-1</u> : All three estimates are within the range 0.25 to
0.85 Hz, but they are not the same. The most unstable pole is
returned.

(b) <u>IERP=-2</u> : At least one of the estimates is outside the frequency range. The most unstable pole within the range is returned.

(c) <u>IERP=-3</u> : All estimates are outside the frequency range. No estimate is returned.

B.4 Best Estimate Decay Ratio Evaluation

After the three previous estimates of the DR have been obtained, subroutine CONFID evaluates the results, picks a best estimate DR and assigns a confidence level to it. A confidence level of +7 is highest and means that the estimate passed all the tests. If the confidence is less than +3, the estimate could still be good, but it should be treated with caution; finally, if the confidence is negative, the estimate is probably absolutely worthless.

On return, the parameter IER contains a positive value that has coded in it the reasons for the confidence level assigned to the DR estimate (see the program listing for details). Note that if IER is negative, it means that some error was found in the input parameters and no intent was made to estimate the stability.

B.5 Error Estimate

This set of subroutines has a provision for estimating the error associated with the estimates. For this purpose, the raw time data is divided into several records; a PSD is caculated for each

record and then for the total length of data. After each individual-record PSD is computed, a call to STABIL is made with the parameter IFLAGZ equal to zero. Finally, a call is made with the average PSD and IFLAGZ equal to the number of data records; subroutine ERREST then evaluates the error in the DR and NF estimates as the maximum dispersion between all the previous estimates. In this way both the statistical and the bias error are addressed, since the estimate with the average PSD will have a reduced bias error.

This error estimate can not be guaranteed to be conservative, but experience has shown that in general it is.

APPENDIX C

LISTING OF THE STABIL SUBROUTINES

```
C
C
C
        SUBROUTINE STABIL(ITYPE,C,PSD,NP,DT,NBK,IFLAGZ
     >          ,DR,FN,ICONF,DRERR,FNERR,IER)
C
C **   This subroutine evaluates the stability of a BWR
C **   from the PSD of an APRM noise signal.
C
C **            Jose March-Leuba    October, 1983
C
C -----------------------------------------------------------------
C
C **   CALLING PARAMETERS:
C
C **           ITYPE   - Type of input
C **                        0 - Correlation supplied  -- RECOMMENDED
C **                        1 - PSD supplied
C **           C       - Autocorrelation array (it must be dimensioned
C **                     regardless of ITYPE) dimension > NP if ITYPE=1
C **           PSD     - Power Spectral Density of APRM noise signal.
C **                     The absolute magnitude of the PSD is not
C **                     important. The FFT blocksize and sampling
C **                     times should be chosen so that:
C **                     deltafreq < 0.1  & Nyquistfreq > 1.0 Hz.
C **                     Hanning window is recommended.
C **           NP      - Number of points in either correlation or PSD
C **                     NP < NPMX  if ITYPE=1
C **           DT      - Original sampling time 0.05 < DT < 0.5 s
C **           NBK     - Number of blocks analyzed
C **                     i.e.  NBK = NP/(2*NP)
C **           IFLAGZ  - Flag with values:
C **                        0 - Normal call (in PSDREC, PSD=PSDZ)
C **                        n - PSD is an average of the PSD's used
C **                            in the last <n> calls.
C **                     If IFLAGZ > 1 then an approximate error estimate
C **                     is returned along with the best estimate of
C **                     the decay ratio and natural frequency.
C
C -----------------------------------------------------------------
C
C **   THE SUBROUTINE RETURNS:
C
C **           DR      - Best available estimate of the asymptotic
C **                     decay ratio.
C **           FN      - Best available estimate of the damped
C **                     frequency of oscillation.  It is not the
C **                     natural frequency. This is the imaginary
C **                     part of the most unstable pole.
C **           ICONF   - Confidence level of the estimate:
C **                        +7 - highest confidence
C **                        ICONF >3 is a good confidence level
C **                        IF ICONF<0 then the estimate is no good
C **                     If ICONF is negative, the estimate should be
C **                     used with caution.
C **           DRERR   - An estimate of the error in DR.
C **                     WARNING:     only returned if IFLAGZ > 1
C **                        This is not guarantied to be a conservative
```

```
C **                     estimate of the error
C **           FNERR   - An estimate of the error in FN.
C **                     Same restrictions than for DRERR apply.
C **           IER     - If < 0 then, error in input parameters.
C
C -----------------------------------------------------------------
C
C **   SUBROUTINES CALLED:
C
C **           INPCHK  - Checks input parameters
C **           CHKPSD  - Computes RMS and checks if PSD's are positive
C **           CHKCOR  - Checks that C(0) > C(i) i=1,..,NP and C(0) > 0
C **           CORREL  - Inverse FFT PSD.  Returns autocorrelation.
C **           FFT     - Fast Fourier Transform
C **           DRCORR  - Calculates the apparent D.R. of the autocorr.
C **           ARMODL  - Fits a univariate AutoRegressive (AR) model
C **           AR      - Univariate AR model
C **           DECOMP  - Triangular matrix decomposition
C **           SOLVE   - Solve sytem of equations after DECOMP
C **           AIC     - Function to calculate Akaike's inform. crit.
C **           DRTIME  - Calculates the asymptotic D.R. from the
C **                     impulse response of the AR model in the
C **                     time domain.
C **           CONVRG  - Calculates DR and checks for convergence
C **           DRFREQ  - Calculates the D.R. and N.F. from the
C **                     AR model in the frequency domain.
C **           ZERO    - Finds a zero of a polynomial
C **           POLY    - Evaluates a polynomial and its derivative
C **           CONFID  - It evaluates the previous results and
C **                     determines the confidence level.
C **           ERREST  - Estimates the error if IFLAGZ > 1.
C
C -----------------------------------------------------------------
C
C **   OVERLAY INFORMATION:
C
C **           STABIL
C **                   INPCHK
C **                           CHKPSD
C **                           CHKCOR
C **                   CORREL
C **                           FFT
C **                   DRCORR
C **                   ARMODL
C **                           AR
C **                                   DECOMP
C **                                           SOLVE
C **                                   SOLVE
C **                           AIC
C **                   DRTIME
C **                           CONVRG
C **                   DRFREQ
C **                           ZERO
C **                                   POLY
C **                   CONFID
C **                   ERREST
C
C -----------------------------------------------------------------
```

```
C
C **      INTERNAL VARIABLES:
C
C **          RMS        - PSD RMS.
C **          C          - Autocorrelation function.
C **          DRC        - Apparent DR of C.
C **          FNC        - Apparent FN of C.
C **          A          - AR parameters.
C **          DTAR       - Equivalent sampling time used for AR model
C **          DRS        - Asymptotic DR using impulse response.
C **          FNS        - Asymptotic FN using impulse response.
C **          DRP        - Asymptotic DR using pole search.
C **          FNP        - Asymptotic FN using pole search.
C **          DRSTCK     - Array of last DR's for error estimation
C **          FNSTCK     - Array of last FN's for error estimation
C
C -------------------------------------------------------------
C
C
C **      DEFINITIONS:
C
C **      Apparent Decay Ratio:    The ratio between the first and
C **                               second peaks in either correlation
C **                               step or impulse responses.
C **      Asymptotic Decay Ratio:  The limit as time goes to infinity
C **                               of the ratio between consecutive peaks.
C **                               The Asymp. DR truly represents the
C **                               stability of the system, the apparent
C **                               DR does not.
C **      Decay Ratio:             Equivalent to asymptotic DR.
C
C -------------------------------------------------------------
C
C **      SAMPLE MAIN PROGRAM
C
C         REAL *4 C(0:128),PSD(0:128)
C         COMPLEX TMP,S,P(2),Z(2)
C
C         ITYPE=1           ! PSD SUPPLIED
C         NP=128
C         DT=0.1
C         NBK=25
C         IFLAGZ=0
C
C
C         DW=1./(2.*FLOAT(NP)*DT)
C         NPL=2
C         P(1)=CMPLX(-0.0441,.4)   ! DR=0.5 , FN=0.4
C         P(2)=CONJG(P(1))
C         DO 100 I=0,128
C         W=FLOAT(I)*DW
C         S=CMPLX(0.,W)
C         TMP=CMPLX(1.,0.)
C         DO 110 IP=1,NPL
C 110     TMP=TMP/(S+P(IP))
C         PSD(I)=TMP*CONJG(TMP)
C 100     CONTINUE
C
C
```

```
C         CALL STABIL(ITYPE,C,PSD,NP,DT,NBK,IFLAGZ
C        >            ,DR,FN,ICONF,DRERR,FNERR,IER)
C
C         WRITE(6,3020) NPL,(P(I),I=1,NPL)
C 3020    FORMAT('1'///' NUMBER OF POLES = ',I6,<NPL>(/' ',2G20.5))
C         WRITE(6,3000) DR,FN,ICONF,IER
C 3000    FORMAT(////' DR    = ',G20.5/
C        >          ' NF    = ',G20.5/
C        >          ' ICONF = ',I10/
C        >          ' IER   = ',I10)
C         STOP
C         END
```

```
C
C   -------------------------------------------------------------------
C   **********************************************************************
C   -------------------------------------------------------------------
C
        PARAMETER NPMX=256        ! MAXIMUM NUMBER OF POINTS
        PARAMETER MXSTCK=10       ! MAX. NUMBER OF CALLS BEFORE ERROR CHECK
        PARAMETER NARMX=30        ! MAX. AR MODEL ORDER TO USE
        PARAMETER NARMN=5         ! MIN. AR MODEL ORDER TO USE >=2
C
C   -------------------------------------------------------------------
C
        REAL *4 C(0:NP),PSD(0:NP),DT,DR,FN,DRERR,FNERR
        INTEGER *2 NP,NBK,IFLAGZ,ICONF
C
C   -------------------------------------------------------------------
C
        COMMON /STAB/ A,NAR,DTAR,VAR2,DRSTCK,FNSTCK
     >            ,DRC,FNC,IERC,DRS,FNS,IERS,DRP,FNP,IERP
        REAL *4 DRSTCK(MXSTCK),FNSTCK(MXSTCK),F(2)
        REAL *8 A(1:NARMX)
C
C   -------------------------------------------------------------------
C   **********************************************************************
C   -------------------------------------------------------------------
C
C
        CALL INPCHK(ITYPE,C,PSD,NP,DT,NBK,IFLAGZ,IER)
        IF(IER.LT.0) RETURN           ! FATAL ERROR IN INPUT
C
        IF(ITYPE.EQ.1)  CALL CORREL(PSD,NP,DT,C)
C
        CALL DRCORR(C,NP,DT,DRC,FNC,IERC)
C
        NAR=NARMX
        CALL ARMODL(C,NP,DT,NBK,NAR,A,DTAR,VAR2,IERA)
C
        CALL DRTIME(A,NAR,DTAR,DRS,FNS,IERS)
C
        CALL DRFREQ(A,NAR,DTAR,DRP,FNP,IERP)
C
        IER=0            ! NOTE, IF <0 IT WOULD HAVE RETURNED
        CALL CONFID(A,NAR,IERA,DRC,FNC,IERC,DRS,FNS,IERS
     >            ,DRP,FNP,IERP,DR,FN,ICONF,IER)
C
        CALL ERREST(DR,FN,DRSTCK,FNSTCK,MXSTCK,IFLAGZ,DRERR,FNERR,IER)
C
        RETURN
        END
```

```
C
C
C
        SUBROUTINE INPCHK(ITYPE,C,PSD,NP,DT,NBK,IFLAGZ,IER)
C
C **    This subroutine checks the input parameters
C **    calling parameters are the same than STABIL
C
C   -------------------------------------------------------------------
C   **********************************************************************
C   -------------------------------------------------------------------
C
        PARAMETER NPMX=256        ! MAXIMUM NUMBER OF FREQUENCY POINTS
        PARAMETER NARMX=30        ! MAX. AR MODEL ORDER TO USE
C
C   -------------------------------------------------------------------
C
        REAL *4 C(0:NP),PSD(0:NP),DT,DFC
        INTEGER *2 NP,NBK,IFLAGZ,IER
C
C   -------------------------------------------------------------------
C   **********************************************************************
C   -------------------------------------------------------------------
C
        IER=0
        IF(DT.LE.0.)                IER=-1
        IF(NBK.LE.0)                IER=-1
        IF(NP.LE.0 .OR. NP.GT.NPMX) IER=-1
C
        IF(ITYPE.EQ.1) THEN                     ! PSD INPUT
            DF=1./(2.*FLOAT(NP)*DT)
            IF(DF.GT.0.1)           IER=-2
            IF(DF*FLOAT(NP).LT.0.9) IER=-2
            CALL CHKPSD(PSD,NP,DT,IER)
        ELSE                                    ! CORRELATION INPUT
            IF(NP.LT.NARMX)         IER=-2
            IF(DT.LT.0.05 .OR. DT.GT.0.5) IER=-2
            CALL CHKCOR(C,NP,DT,IER)
        ENDIF
C
        IF(IFLAGZ.GT.MXSTCK.OR.IFLAGZ.LT.0.OR.IFLAGZ.EQ.1) IER=-3
C
        RETURN
        END
```

249

```
C
C
C
        SUBROUTINE CHKPSD(PSD,NF,DT,IER)
C
C **     This subroutine calculates RMS and checks
C **            if RMS>0 and PSD(i)>=0 i:1,NF
C
C **        PSD     - Real PSD array
C **        NF      - Number of points in PSD
C **        DT      - Equivalent sampling time for PSD
C **        RMS     - RMS
C **        IER     - Error condition
C
C --------------------------------------------------------------------
C ******************************************************************
C --------------------------------------------------------------------
C
        REAL *4 PSD(O:NF),DT,RMS
        INTEGER *2 NF,IER
C
C --------------------------------------------------------------------
C ******************************************************************
C --------------------------------------------------------------------
C
        RMS=0
        IER=0
        DO 100 I=0,NF
        IF(PSD(I).LT.0.)        IER=-4  ' INDIVIDUAL PSD ALLOWED = 0
100     RMS=RMS+PSD(I)
        IF(RMS.LE.0.)           IER=-5  ' RMS CAN NOT BE = 0
        RETURN
        END
```

```
C
C
C
        SUBROUTINE CHKCOR(C,NC,DT,IER)
C
C **     This subroutine checks that C(0) > C(i) i:1,...,N
C **            and that C(0) > 0
C **     It is called only if ITYPE <> 1
C
C **        C       - Real autocorrelation array
C **        NC      - Number of points in C
C **        DT      - Equivalent sampling time for C
C **        IER     - Error condition
C
C --------------------------------------------------------------------
C ******************************************************************
C --------------------------------------------------------------------
C
        REAL *4 C(O:NC),DT
        INTEGER *2 NC,IER
C
C --------------------------------------------------------------------
C ******************************************************************
C --------------------------------------------------------------------
C
        IER=0
        DO 100 I=1,NC
100     IF(C(I).GT.C(0))        IER=-4
        IF(C(0).LE.0.)          IER=-5
        RETURN
        END
```

250

```
C
C
C
        SUBROUTINE CORREL(PSD,NF,DT,C)
C
C **     This subroutine inverse FFTs PSD.
C
C **         PSD     - Real PSD array
C **         C       - Real correaltion array
C
C **     SUBROUTINES CALLED:
C
C **         FFT     - Real Fast Fourier Transform
C
C -------------------------------------------------------------------
C ***************************************************************************
C -------------------------------------------------------------------
C
        PARAMETER NPMX=256      ! MAXIMUM NUMBER OF FREQ. POINTS
C
C -------------------------------------------------------------------
C
        REAL *4 PSD(0:NF),C(0:NF)
        INTEGER *2 NF
C
C -------------------------------------------------------------------
C ***************************************************************************
C -------------------------------------------------------------------
C
C
        COMPLEX TMP(NPMX*2)
C
C
        LN=0
        DF=1./(FLOAT(NF)*2.*DT)
        JUNK=NF*2
   10   JUNK=JUNK/2
        LN=LN+1                 ! LOGARITH BASE 2 OF BLOCKSIZE (FOR FFT)
        IF(JUNK.GT.1) GO TO 10
C
        DO 100 I=0,NF
        L=I+1
  100   TMP(L)=CMPLX(PSD(I)*DF/2.,0.)    ! 2 BECAUSE OF NEGATIVE FREQ.
        DO 110 L=2,NF
        K=2*NF-L+2
  110   TMP(K)=CONJG(TMP(L))
C
        CALL FFT(1,LN,TMP)
C
        DO 120 I=0,NF
  120   C(I)=REAL(TMP(I+1))
C
        RETURN
        END
```

```
C
C
C
        SUBROUTINE DRCORR(C,NF,DT,DRC,FNC,IERC)
C
C **     This subroutine estimates the aparent decay ratio of
C **     the autocorrelation.
C **     If the first peak in C is not the maximum peak value,
C **     an error condition is returned.  This means tnat a lower
C **     frequency peak might influence the results, or that a
C **     first peak could not be found.
C
C **         C       - Real correaltion array
C **         NF      - Number of points in C
C **         DT      - Sampling time in C
C **         DRC     - Apparent DR of C
C **         FNC     - Apparent Natural Freq. of C
C **         IERC    - Error condition if <0
C **                      -1 : Peak could not be found
C **                      -2 : Low frequencies influence C
C **                              (first minimum not negative)
C **                      -3 : There is a low freq. peak with
C **                              smaller real part.
C
C -------------------------------------------------------------------
C ***************************************************************************
C -------------------------------------------------------------------
C
        REAL *4 C(0:NF),DT,DRC,FNC
        INTEGER *2 NF,IERC
C
C -------------------------------------------------------------------
C ***************************************************************************
C -------------------------------------------------------------------
C
        IERC=0
        V1=-999.
        C1=-999.
        I1=1
        CMAX=-999.
        DRC=-999.
        FNC=-999.
C
        DO 100 I=1,NF
        D1=C(I)-C(I-1)
        D2=C(I)-C(I+1)
        IF(D1.LE.0. .OR. D2.LE.0.) GO TO 10      ! NOT A PEAK
        IF(C1.EQ.-999.) I1=I
        IF(C1.EQ.-999.) C1=C(I)          ! FIRST PEAK
        IF(C(I).GT.CMAX) CMAX=C(I)       ! HIGHEST PEAK
        GO TO 100
   10   IF(D1.GT.0. .OR. D2.GT.0.) GO TO 100     ! NOT A VALLEY (MIN.)
        IF(V1.EQ.-999.) V1=C(I)          ! FIRST VALLEY
  100   CONTINUE
C
        IF(C1.NE.-999 .AND. V1.NE.-999.) DRC=( (C1-V1)/(C(0)-V1) )**2
        IF(C1.NE.-999) FNC=1./(FLOAT(I1)*DT)
        IF(C1.EQ.-999) IERC=-1           ! PEAKS NOT FOUND
```

```
        IF(V1.EQ.-999)  IERC=-1        ! VALLEYS NOT FOUND
        IF(V1.GE.0.)    IERC=-2        ! FIRST VALLEY NOT <0
        IF(C1.NE.CMAX)  IERC=-3        ! FIRST PEAK NOT HIGHEST
        RETURN
        END
```

```
C
C
C
        SUBROUTINE ARMODL(C,NF,DT,NBK,NAR,A,DTAR,VAR2,IERA)
C
C **   This subroutine returns the "optimal" AR model of order NAR
C **   "optimal" is defined as the best AIC criterion
C **          x(t)=sum A(k)*x(t-k*DT)   + w(t)   k=1,NAR
C
C **   CALLING PARAMETERS:
C
C **          C       - Autocorrelation function (magnitude unimportant)
C **          DT      - Sampling time in correlation
C **          NAR     - Order of AR model
C **          A       - Array containing the AR parameters
C **          DTAR    - Sampling time used for AR model
C **          VAR2    - Residual error variance
C **          IERA    - Error condition
C
C **   SUBROUTINES CALLED:
C
C **          AR      - Calculate AR parameters
C
C
C -----------------------------------------------------------------
C *****************************************************************
C -----------------------------------------------------------------
C
        PARAMETER NARMN=5        ! MINIMUM ORDER TO FIT
        PARAMETER NARMX=30       ! MAXIMUM ORDER TO FIT
        PARAMETER NARINC=5       ! ORDER INCREMENTS
C
C -----------------------------------------------------------------
C
        INTEGER *2 NF,NAR
        REAL *4 C(0:NF),DT,DTAR,CAR(0:NARMX)
        REAL *8 A(1:NAR)
C
C -----------------------------------------------------------------
C *****************************************************************
C -----------------------------------------------------------------
C
        IERA=0
        POINTS=2.*FLOAT(NF)*FLOAT(NBK)
        DTAR=DT
        NC=NF
        NCAR=NC
        ISKIP=1
C
  10    AICMX=-1.E38
C
        DO 200 I=0,NARMX
        CAR(I)=C(I*ISKIP)
  200   CONTINUE
C
        DO 100 IAR=NARMN,NARMX,NARINC
        CALL AR(CAR,NARMX,DT,IAR,A,VAR2)
        AIC=POINTS*ALOG(ABS(VAR2/C(0))) + 2.*FLOAT(NAR)
```

```
        AIC=ABS(AIC)      ' NOTE, VAR2/C(0)<1, SO AIC<0
        IF(AIC.GT.AICMX) NAR=IAR
        IF(AIC.GT.AICMX) AICMX=AIC
100     CONTINUE
        LAST=IAR-NARINC
C
        IF(NAR.NE.LAST) GO TO 30
        ISKIP=ISKIP+1
        DTAR=DT*ISKIP       ' IF MODEL NOT LARGE ENOUGH DOUBLE SAMPLING TIME
        NCAR=NC/(ISKIP)
        IERA=IERA+1      ! NUMBER OF SAMPLIG TIME DOUBLINGS
        IF(DTAR.GT.0.51) GO TO 20     ! MAXIMUM FREQUENCY TOO SMALL
        IF(NCAR.LT.NARMX) GO TO 20    ! NOT ENOUGH LOW FREQUENCY
        GO TO 10
C
20      ISKIP=ISKIP-1
        NCAR=NC*ISKIP
        DTAR=DT*ISKIP
        IERA=-1           ! MODEL ORDER USED WAS NOT LARGE ENOUGH
C
30      CALL AR(CAR,NARMX,DT,NAR,A,VAR2)
C
        RETURN
        END
```

```
C
C
C
        SUBROUTINE AR(C,NF,DT,NAR,A,VAR2)
C
C **   This subroutine returns an AR model of order NAR
C **          x(t)=sum A(k)*x(t-k*DT)   + w(t)    k=1,NAR
C
C **   CALLING PARAMETERS:
C
C **          C        - Autocorrelation function (magnitude unimportant)
C **          DT       - Sampling time in correlation
C **          NAR      - Order of AR model
C **          A        - Array containing the AR parameters
C
C **   SUBROUTINES CALLED:
C
C **          DECOMP   - Triangular matrix decomposition
C **          SOLVE    - Solve a system of equations
C -----------------------------------------------------------------
C ***************************************************************************
C -----------------------------------------------------------------
C
        PARAMETER NARMX=30
C
C -----------------------------------------------------------------
C
        REAL *4 C(0:NF),DT,VAR2
        REAL *8 A(1:NAR),WM(NARMX,NARMX),COND
        INTEGER *4 NDIM,NORDER,IPVT(NARMX)
        INTEGER *2 NF,NAR
C
C -----------------------------------------------------------------
C ***************************************************************************
C -----------------------------------------------------------------
C
        NDIM=NARMX
        NORDER=NAR
C
        DO 100 I=1,NAR
        DO 100 J=I,NAR
        WM(I,J)=C(IABS(I-J))
100     WM(J,I)=WM(I,J)
C
C
        CALL DECOMP(NDIM,NORDER,WM,COND,IPVT,A)
C
        DO 200 I=1,NAR
200     A(I)=C(I)
C
        CALL SOLVE(NDIM,NORDER,WM,A,IPVT)          ' AR PAR. CALC.
C
        VAR2=C(0)
        DO 300 I=1,NAR
300     VAR2=VAR2-A(I)*C(I)       ' RESIDUAL VARIANCE
C
        RETURN
        END
```

```
C
C
C
        SUBROUTINE DRTIME(A,NAR,DT,DRS,FNS,IERS)
C
C **     This subroutine calculates the DR and Nat. Freq.
C **     from the impulse response of the AR model.
C
C **     CALLING PARAMETERS:
C
C **             A         - AR model parameters
C **             NAR       - AR model order
C **             DT        - Sampling time in correlation
C **             DRS       - Impulse response DR
C **             FNS       - Impulse response N. F.
C **             IERS      - Error condition (used in determining
C **                         the DR confidence in CONFID)
C **                         IF <0 ERROR.
C
C -------------------------------------------------------------------
C *****************************************************************
C -------------------------------------------------------/----------
C
        PARAMETER NARMX=30      ! SAME AS NAR IN STABIL
        PARAMETER MXSTP=3000    ! MAXIMUM NUMMER OF TIME STEPS FOR CONVERGENCE
C
C -------------------------------------------------------------------
C
        REAL *8 A(1:NAR),X(0:NARMX)
        REAL *4 DT,DRS,FNS
        INTEGER *2 NAR,IERS
        LOGICAL *1 FIRST,END
C
C -------------------------------------------------------------------
C *****************************************************************
C -------------------------------------------------------------------
C
        FIRST=.TRUE.
        NPEAKS=0
        DRS=0.
        FNS=0.
        T=0.
        DO 100 I=0,NARMX
100     X(I)=0.
        V=1.                    ! IMPULSE MAGNITUDE = 1.
C
  10    DO 110 I=NAR,1,-1
110     X(I)=X(I-1)
        X(0)=V
        V=0.                    ! FORCING FUNCTION
        DO 120 I=1,NAR
120     X(0)=X(0)+A(I)*X(I)     ! IMPULSE RESPONSE
        T=T+DT                  ! TIME
        IF(T.GT. FLOAT(MXSTP)*DT) GO TO 20
C
        D1=X(1)-X(2)
        D2=X(1)-X(0)
        IF(D1*D2.LE. 0.) GO TO 10       ! NOT AN EXTREMA
```

```
C
C
        A2=(X(2)-2.*X(1)+X(0))/2.        ! PARABOLA FIT TO EXTREMA
        A1=(X(1)-X(0))-A2
        TEXT=(-A1/(2.*A2))*DT
        IF(TEXT.GT.2.*DT .OR. TEXT.LT.0.) TEXT=DT
        EXT=X(0)+A1*TEXT+A2*TEXT**2
        TEXT=(T-2.*DT)+TEXT
        NPEAKS=NPEAKS+1
C
        CALL CONVRG(EXT,TEXT,DRS,FNS,FIRST,END,IERS)
C
        IF(.NOT.END) GO TO 10           ! NOT CONVERGED
C
        RETURN
C
  20    DRS=-999.
        FNS=-999.
        IERS=-2         ! NOT CONVERGED IN MXSTP TIME STEPS (PEAKS FOUND)
        IF(NPEAKS.EQ.0) IERS=-3         ! NO PEAKS FOUND
        RETURN
        END
```

```
C
C
C
        SUBROUTINE CONVRG(X,T,DR,FN,FIRST,END,IERS)
C
C **     This subroutine calculates the DR and Nat. Freq.
C **     and checks for convergence (i.e. if it is asymptotic)
C **     The D.R. estimation uses a formula that allows for
C **     a bias in the response (i.e. if it was a STEP response
C **     this subroutine would also work).
C **     The calling subroutine has to do a search anf find
C **     the extrema (max. or min), CONVRG is called then
C **     with the value of the extrema (see DRTIME).
C **
C **     CALLING PARAMETERS:
C
C **         X       - Impulse response's current extrema
C **         T       - Current time
C **         DR      - Impulse response DR
C **         FN      - Impulse response N. F.
C **         FIRST   - =.TRUE. if it is first call
C **         END     - when =.TRUE., DR and FN are converged
C **         IERS    - Error condition (used in determining
C **                   the DR confidence in CONFID)
C **                   IF <O ERROR.
C
C ----------------------------------------------------------------
C ****************************************************************
C ----------------------------------------------------------------
C
        PARAMETER CONV=1.E-1    ! CONVERGENCE CRITERIUM FOR DR
        PARAMETER NOLDMX=10     ! NUMBER OF DR'S WITHIN CONV FOR CONVRG.
        PARAMETER ROUND=1.E-15  ! ROUNDOFF CRITERIUM
C
C ----------------------------------------------------------------
C
        REAL *4 X,T,DR,FN,DROLD(O:NOLDMX),FNOLD(O:NOLDMX)
        INTEGER *2 IERS
        LOGICAL *1 FIRST,END
C
C ----------------------------------------------------------------
C ****************************************************************
C ----------------------------------------------------------------
C
C
        IERS=O
        END=.FALSE.
        IF(.NOT.FIRST) GO TO 10
        FIRST=.FALSE.
        EX1=-999.
        EX2=-999.
        TEX1=0.
        TEX2=0.
        NOLD=0
        NPEAKS=0
        EXFRST=X
C
C
```

```
10      EX3=X
        TEX3=T
        NPEAKS=NPEAKS+1
        IF(EX1.EQ.-999.) GO TO 16
C
        D1=EX3-EX2
        D2=EX1-EX3
        IF(D2.NE.O.)  GM=D1/D2
        IF(D2.EQ.O.)  GM=1.E32
        IF(GM.NE.-1.) DR=(GM/(1.+GM))**2
        FN=1./(TEX3-TEX1)
        DROLD(O)=DR
        FNOLD(O)=FN
        IF(NOLD.LT.NOLDMX) GO TO 15
C
C
        DRERR=0.
        DRAV=0.
        FNERR=0.
        FNAV=0.
        DO 110 I=0,NOLD
        DRAV=DRAV+DROLD(I)/FLOAT(NOLD+1)
        FNAV=FNAV+FNOLD(I)/FLOAT(NOLD+1)
        DRERR=AMAX1(DRERR,ABS(DROLD(O)-DROLD(I)))
        FNERR=AMAX1(FNERR,ABS(FNOLD(O)-FNOLD(I)))
110     CONTINUE
        IF(DRERR.LT.DRAV*CONV.AND.FNERR.LT.FNAV*CONV) GO TO 20   ! CONVERGED
C
15      DO 120 I=NOLDMX,1,-1
        DROLD(I)=DROLD(I-1)
120     FNOLD(I)=FNOLD(I-1)
        IF(NOLD.LT.NOLDMX)        NOLD=NOLD+1
C
        IF(ABS(D2).LT.ROUND .AND. ABS(D1).LT.ROUND) GO TO 30      ! ROUNDOFF
C
16      EX1=EX2
        EX2=EX3
        TEX1=TEX2
        TEX2=TEX3
        END=.FALSE.
        RETURN
C
20      END=.TRUE.
        IERS=O
        DR=DRAV             ! IF CONVERGED RETURN AVERAGE DR
        FN=FNAV
        RETURN
C
30      END=.TRUE.
        IERS=-1           ! DR NOT CONVERGED AND RESPONSE WITHIN ROUNDOFF
        NPEAKS=NPEAKS/2           ! NOTE HALF ARE VALLEYS
C **    THE FOLLOWING ARE APPROXIMATED VALUES
        FN=FLOAT(NPEAKS)/T               ! TAKE AVERAGE PERIOD
        YR=ABS(X/EXFRST)
        DR=YR**(1./FLOAT(NPEAKS))        ! DECAYED FROM 1 TO YY IN NPEAKS
        RETURN
        END
```

```
C
C
C
        SUBROUTINE DRFREQ(A,NAR,DT,DR,FN,IERP)
C
C **   This subroutine estimates a DR and a N.F. from the
C **   AR model in the frequency domain.
C **   It first gets a rough estimate by looking at the maximum
C **   value of the derivative of the phase with respect to
C **   frequency.  Then, it improves the estimate by doing a pole
C **   search uising Newton's method in the compex domain.
C
C **   CALLING PARAMETERS:
C
C **        A       - AR parameters
C **        NAR     - AR model order
C **        DT      - Sampling time for AR model
C **        DR      - Decay ratio estimate
C **        FN      - Natural frequency estimate
C **        IERP    - Error code (error if IERP<0)
C
C **   SUBROUTINES CALLED:
C **        ZERO    - Estimates a zero of a polynomial
C
C --------------------------------------------------------------------
C *****************************************************************
C --------------------------------------------------------------------
C
        PARAMETER WMIN=0.2      ! MINIMUM FREQ FOR SEARCH
        PARAMETER WMAX=0.9      ! MAXIMUM FREQ FOR SEARCH
        PARAMETER NW=B          ! NUMBER OF INCREMENTS FOR SEARCH >=3
C
C --------------------------------------------------------------------
C
        COMPLEX EW,EWN,TF,EST,Z
        REAL *8 A(1:NAR)
        REAL *4 DT,DR,FN,DRE(3),FNE(3)
        INTEGER *2 NAR,IERP
C --------------------------------------------------------------------
C *****************************************************************
C --------------------------------------------------------------------
C
        IERP=0
        PI=3.14159265
        DW=(WMAX-WMIN)/FLOAT(NW-1)
        WO=WMIN
        IPHASE=1
C
   1    PSDMX=0.
        DPHMX=0.
        DO 100 IW=1,NW
        W=WO+FLOAT(IW-1)*DW
        EW=CEXP(CMPLX(0.,-2.*PI*W*DT))   ! BACKSHIFT OPERATOR
        TF=CMPLX(1.,0.)
        EWN=CMPLX(1.,0.)
        DO 110 I=1,NAR
        EWN=EWN*EW
```

```
 110    TF=TF-A(I)*EWN
        PSD=1./(TF*CONJG(TF))
        IF(REAL(TF).NE.0.)       PH=-ATAN(AIMAG(TF)/REAL(TF))
        IF(REAL(TF).LT.0.) PH=PH-PI
        IF(IW.EQ.1) GO TO 20
        IF(PSD.LT.PSDMX) GO TO 10
        PSDMX=PSD
        WPSDMX=W
  10    DPH=PHOLD-PH
        SIGN=DPH/ABS(DPH)
  11    IF(ABS(DPH).GT.2.*PI) DPH=SIGN*(ABS(DPH)-2.*PI)
        IF(ABS(DPH).GT.2.*PI) GO TO 11
        IF(ABS(DPH).LT.ABS(DPHMX)) GO TO 20
        DPHMX=DPH
        WDPHMX=W
  20    PHOLD=PH
        PSDOLD=PSD
 100    CONTINUE
        IF(PSDMX.EQ.0.) WPSDMX=WDPHMX   ! IF NO PEAK FOUND SAME AS PHASE
        GO TO (71,72,73) IPHASE
C
C **   DEFINE MAXIMA MORE ACCURATELY   (FINER MESH)
C
  71    DW=DW/FLOAT(NW)
        WP=WPSDMX-FLOAT(NW)*DW/2.       ! PSD MAX FIRST ITERATION
        WD=WDPHMX-FLOAT(NW)*DW/2.       ! DPH MAX FIRST ITERATION
        WO=WD
        IPHASE=2
        GO TO 1
  72    DPHMAX=DPHMX/DW                 ! d(PHASE)/d(FREQ)  RAD/HZ
        ARR=-1./DPHMAX  ! REAL PART OF POLE (HZ)
        ARI=WDPHMX              ! IMAG PART OF POLE (HZ)
        IF(WD.EQ.WP) GO TO 73   ! SAME RANGE THAN PSDMAX
        WO=WP                   ! DIFERENT RANGES
        IPHASE=3
        GO TO 1
  73    DPHMAX=DPHMX/DW         ! d(PHASE)/d(FREQ)  RAD/HZ
        ARRP=-1./DPHMAX ! REAL PART OF POLE (HZ)
        ARIP=WDPHMX             ! IMAG PART OF POLE (HZ)
C
C **   IMPROVE ESTIMATES
C
C FROM dP/dw
        EST=CEXP(-2.*PI*DT*CMPLX(ARR,ARI))      ! NOTE EST=exp(-s*dt)
        CALL ZERO(NAR,A,EST,Z,IER)
        IF(IER.EQ.0) Z=-CLOG(Z)/(2.*PI*DT)      ! ZERO IN HZ (s-domain)
        IF(IER.EQ.0) DRE(1)=EXP(2.*PI*REAL(Z)/ABS(AIMAG(Z)))     ! D.R.
        IF(IER.EQ.0) FNE(1)=ABS(AIMAG(Z))       ! N.F. FROM MAX dP/dw
        IF(IER.LT.0) DRE(1)=-999.
        IF(IER.LT.0) FNE(1)=-999.
C
C FROM MAX PSD
        EST=CEXP(-2.*PI*DT*CMPLX(0.,WPSDMX))    ! NOTE EST=exp(-s*dt)
        CALL ZERO(NAR,A,EST,Z,IER)
        IF(IER.EQ.0) Z=-CLOG(Z)/(2.*PI*DT)      ! ZERO IN HZ (s-domain)
        IF(IER.EQ.0) DRE(2)=EXP(2.*PI*REAL(Z)/ABS(AIMAG(Z)))     ! D.R.
        IF(IER.EQ.0) FNE(2)=ABS(AIMAG(Z))       ! N.F.
        IF(IER.LT.0) DRE(2)=-999.
```

256

```
          IF(IER.LT.0) FNE(2)=-999.
C
C FROM estimate of dP/dw   @ MAX PSD
          EST=CEXP(-2.*PI*DT*CMPLX(ARRP,ARIP))     ' NOTE EST=exp(-s*dt)
          CALL ZERO(NAR,A,EST,Z,IER)
          IF(IER.EQ.0) Z=-CLOG(Z)/(2.*PI*DT)       ' ZERO IN HZ (s-domain)
          IF(IER.EQ.0) DRE(3)=EXP(2.*PI*REAL(Z)/ABS(AIMAG(Z)))    ' D.R.
          IF(IER.EQ.0) FNE(3)=ABS(AIMAG(Z))               ! N.F. FROM MAX dP/dw
          IF(IER.LT.0) DRE(3)=-999.
          IF(IER.LT.0) FNE(3)=-999.
C
C
C **    EVALUATION OF RESULTS
C
          DR=-999.
          FN=-999.
          Y1=0.
          Y2=0.
          Y3=0.
          Y4=0.
          DO 200 I=1,3
          IF(FNE(I).GT.WMAX .OR. FNE(I).LT.WMIN) GO TO 210       ' OUTSIDE RANGE
          IF(DRE(I).GT.DR) FN=FNE(I)
          IF(DRE(I).GT.DR) DR=DRE(I)
          Y1=Y1+DR
          Y2=Y2+DR**2
          Y3=Y3+FN
          Y4=Y4+FN**2
          GO TO 200
   210    IERP=-2           ' AT LEAST ONE POLE OUT OF FREQ. RANGE
   200    CONTINUE
C
          IF(DR.EQ.-999.) GO TO 40
          IF(IERP.EQ.-2) RETURN    ! AT LEAST ONE POLE OUT OF FREQ. RANGE
C
          DRAV=Y1/3.
          DRVAR=Y2/3.-DRAV**2
          FNAV=Y3/3.
          FNVAR=Y4/3.-FNAV**2
          IF(DRVAR.GT.DRAV*(0.05**2)) GO TO 50     ! 5% STANDARD DEVIATION
          IF(FNVAR.GT.FNAV*(0.05**2)) GO TO 50     ! 5% STANDARD DEVIATION
          IERP=0
          RETURN               ! EVERYTHING ALL RIGHT
C
   40     IERP=-3              ! ALL POLES OUT OF RANGE
          RETURN               ! NO DR ESTIMATE   (DR=-999.)
   50     IERP=-1              ! ALL POLES IN RANGE, BUT NOT THE SAME
          RETURN               ! DR ESTIMATE GOOD BUT NOT RELIABLE
          END
```

```
C
C
C
C         SUBROUTINE ZERO(N,A,X,Z,IER)
C
C **    This subroutine find a zero of a polynomial of the form
C **         1.-sum( A(i)*z**i ) = 0
C **    Normally, it will find a zero close to the initial guess X.
C
C **    CALLING PARAMETERS:
C **            N          - Order of poly
C **            A          - Poly coeff.
C **            X          - First estimate in l-domain [exp(-2piwdt)]
C **            Z          - One of the zeros of the poly in l-domain
C **            IER        - Error parameter  (error if IER<0)
C
C **    SUBROUTINES CALLED:
C **            POLY       - Evaluates a polynomial and its derivative
C
C ------------------------------------------------------------------------
C ************************************************************************
C ------------------------------------------------------------------------
C
          PARAMETER CONV=1.E-5      ' CONVERGENCE CRITERIUM
          PARAMETER MXITER=50       ' MAX NUMBER OF ITERATIONS
C
C ------------------------------------------------------------------------
C
          COMPLEX X,Z,P,DP,P1,DP1,DX
          REAL *8 A(1:N)
          INTEGER N
C
C ------------------------------------------------------------------------
C ************************************************************************
C ------------------------------------------------------------------------
C
          IER=0
          EPS=AMAX1(CABS(X)*CONV,1.E-5)
          K=0
          CALL POLY(N,A,X,P,DP)
   10     J=0
   30     DX=(P/DP)/(2**J)
          Z=X-DX
          CALL POLY(N,A,Z,P1,DP1)
          IF(CABS(DX).LT.EPS) RETURN
          IF(CABS(P1).LE.CABS(P)) GO TO 20
          J=J+1
          IF(J.LT.10) GO TO 30     ' STEP CONTROL
   20     P=P1
          DP=DP1
          X=Z
          K=K+1
          IF(K.LE.MXITER) GO TO 10
          Z=CMPLX(1.E32,1.E32)
          IER=-1
          RETURN
          END
```

```
C
C
        SUBROUTINE POLY(N,A,X,P,DP)
C
C **    This sub. evaluates the polynomial A @ X and its derivative
C **    See sub. ZERO for details
C
C ------------------------------------------------------------------------
C ************************************************************************
C ------------------------------------------------------------------------
C
        COMPLEX P,DP,X,XK
        REAL *8 A(1:N)
C
C ------------------------------------------------------------------------
C ************************************************************************
C ------------------------------------------------------------------------
C
        P=CMPLX(1.0,0.)
        DP=CMPLX(-SNGL(A(1)),0.)
        XK=CMPLX(1.0,0.)
        DO 100 K=1,N-1
        XK=XK*X
        P=P-A(K)*XK
100     DP=DP-FLOAT(K+1)*A(K+1)*XK
        P=P-A(N)*XK*X
        RETURN
        END
```

```
C
C
C
        SUBROUTINE CONFID(A,NAR,IERA,DRC,FNC,IERC,DRS,FNS,IERS
     >          ,DRP,FNP,IERP,DR,FN,ICONF,IER)
C
C **    This subroutine evaluates the DR's and FN's calculated
C **    in DRCORR, DRTIME and DRFREQ  and the error codes generated.
C **    It estimates the best DR and NF.
C
C **    CALLING PARAMETERS:
C **        A       - AR parameters
C **        NAR     - AR order
C **        IERA    - Error parameter returned by ARMODL
C **        DRC     - Apparent DR of autocorrelation
C **        FNC     - Natural frequency of autocorrelation
C **        IERC    - Error parameter returned by DRCORR
C **        DRS     - Asymptotic DR of impulse response
C **        FNS     - Natural frequency of impulse resp.
C **        IERS    - Error parameter returned by DRTIME
C **        DRP     - Asymptotic DR evaluated in DRFREQ
C **        FNP     - Natural frequency
C **        IERP    - Error parameter returned by DRFREQ
C **        DR      - Best estimate decay ratio (returned by CONFID)
C **        FN      - Best estimate nat. freq.  (returned by CONFID)
C **        ICONF   - Confidence level of estimate
C **                        +7 highest confidence
C **                        if ICONF < 0 estimate is no good
C **        IER     - Error code
C **                  If IER=0 no error
C **                  If IER>0 some error occurred (not necesarily fatal)
C **                  Note that if IER<0 there was a fatal error
C **                        during input checking and never got to
C **                        this subroutine.
C
C ------------------------------------------------------------------------
C ************************************************************************
C ------------------------------------------------------------------------
C
        PARAMETER WMIN=0.25     ! MINIMUM FREQUENCY FOR BWR RESONANCE
        PARAMETER WMAX=0.8      ! MAXIMUM FREQUENCY
C
C ------------------------------------------------------------------------
C
        REAL *8 A(1:NAR)
        REAL *4 DRC,FNC,DRS,FNS,DRP,FNP,DR,FN
        INTEGER *2 IERA,IERC,IERS,IERP
C
C ------------------------------------------------------------------------
C ************************************************************************
C ------------------------------------------------------------------------
C
        IER=0
C
        IF(IERA.EQ.-1)  IER=1           ! AR ORDER NOT LARGE ENOUGH
C
        IF(IERC.EQ.-1)  IER=IER+2       ! NO PEAK IN CORRELATION
        IF(IERC.EQ.-2)  IER=IER+4       ! FIRST VALLEY IN CORR. >0
```

```
          IF(IERC.EQ.-3)  IER=IER+8        ! LOWER FREQUENCY IS IMPORTANT
C
          IF(IERS.EQ.-1)  IER=IER+16       ! IMPULSE RESP. NEVER CONVERGED
          IF(IERS.EQ.-2)  IER=IER+32       ! MX TIME REACHED WITHOUT CONV IN IMP
          IF(IERS.EQ.-3)  IER=IER+64       ! NO PEAKS IN IMPULSE RESPONSE
C
          IF(IERP.EQ.-1)  IER=IER+128      ! ALL PEAKS IN RANGE, NOT SAME
          IF(IERP.EQ.-2)  IER=IER+256      ! AT LEAST 1 POLE OUT OF FREQ RANGE
          IF(IERP.EQ.-3)  IER=IER+512      ! ALL POLES OUT OF FREQ RANGE
C
C ** BEST ESTIMATE FOR DR AND NF
C
          DR=-999.
          FN=-999.
C
          IF(FNS.GT.WMAX .OR. FNS.LT.WMIN .OR. FNS.GT.WMAX
     >           .OR. FNS.LT.WMIN) GO TO 10
C
          DR=DRS
          FN=FNS
          IF(DRP.GT.DRS) DR=DRP
          IF(DRP.GT.DRS) FN=FNP
          IF(IERS.NE.-1 .OR. DR.EQ.DRP) GO TO 20
C
          IF(DRS.LT.DRP*1.25) GO TO 20
          DR=DRS
          FN=FNS
          GO TO 20
C
C
10        IF(FNS.GT.WMAX .OR. FNS.LT.WMIN) DR=DRP
          IF(FNS.GT.WMAX .OR. FNS.LT.WMIN) FN=FNP
          IF(FNP.GT.WMAX .OR. FNP.LT.WMIN) DR=DRS
          IF(FNP.GT.WMAX .OR. FNP.LT.WMIN) FN=FNS
C
          IF(DR.EQ.-999.) ICONF=-7
          IF(DR.EQ.-999.) RETURN
C
C
20        ICONF=5
C
          IF(IERA.EQ.-1)  ICONF=ICONF-3
          IF(IERA.GT.0)   ICONF=ICONF-1
C
          IF(IERC.EQ.-1.AND.DR.GT.0.3)    ICONF=ICONF-3
          IF(IERC.EQ.-2)  ICONF=ICONF-1
          IF(IERC.EQ.-3)  ICONF=ICONF-1
C
          IF(IERS.EQ.-1.AND.DR.EQ.DRS)    ICONF=ICONF-3
          IF(IERS.EQ.-2)  ICONF=ICONF-1
          IF(IERS.EQ.-3.AND.DR.GT.0.3)    ICONF=ICONF-3
C
          IF(IERP.EQ.-1)  ICONF=ICONF-1
          IF(IERP.EQ.-2)  ICONF=ICONF-1
          IF(IERP.EQ.-3)  ICONF=ICONF-1
C
C
          IF((ABS(DRS-DRP).LT.0.05*DR).AND.(ABS(FNS-FNP).LT.0.05*FN))
```

```
     >          ICONF=ICONF+1
          IF((ABS(DRC-DR).LT.0.05*DR).AND.(ABS(FNC-FN).LT.0.05*FN))
     >          ICONF=ICONF+1
          RETURN
          END
```

```
C
C
C
      SUBROUTINE ERREST(DR,FN,DRSTCK,FNSTCK,MXSTCK
     >          ,IFLAGZ,DRERR,FNERR,IER)
C
C **   This subroutine calculates an error estimate for DR and FN.
C **   It is intended to be used with the PSDREC system.
C **   Sub. STAB is called N times with PSDZ's as imput.  Then, the
C **   average of the PSDZ's is done and STAB called again with
C **   this average.  The error estimate is the maximum of the
C **   dispersion of all this estimates.  This error accounts for the
C **   standard deviation of the DR's evaluated from PSDZ's, and also
C **   takes into account the bias, since the average PSD estimate
C **   is more accurate than with PSDZ's.
C **   Nevertheless, this is just an estimate and is not guarantied
C **   to be conservative.
C
C **   CALLING PARAMETERS:
C **        DR      - Current decay ratio
C **        FN      - Current natural freq.
C **        DRSTCK  - Array to store old DR
C **        FNSTCK  - Array to store old FN
C **        MXSTCK  - Maximum length of the previous arrays
C **        IFLAGZ  - If =0, then this call is with PSDZ's and
C **                  only updating of the arrays is done
C **                  If =N >0, then an estimate is obtained.
C **                  N is the number of PSDZ's that have already
C **                  been evaluated
C **        DRERR   - Estimate of error in DR
C **        FNERR   - Estimate of error in FN
C **        IER     - Error condition
C
C -------------------------------------------------------------------
C *****************************************************************
C -------------------------------------------------------------------
C
      REAL *4 DR,FN,DRSTCK(1:MXSTCK),FNSTCK(1:MXSTCK)
     >        ,DRERR,FNERR
      INTEGER *2 MXSTCK,IFLAGZ,IER
C
C -------------------------------------------------------------------
C *****************************************************************
C -------------------------------------------------------------------
C
      DRERR=-999.
      FNERR=-999.
      IF(IFLAGZ.LT.2) RETURN
C
      IF(IFLAGZ.GT.0) GO TO 10
C
      DO 100 I=MXSTCK,2,-1
      DRSTCK(I)=DRSTCK(I-1)
 100  FNSTCK(I)=FNSTCK(I-1)
      DRSTCK(1)=DR
      FNSTCK(1)=FN
      RETURN
```

```
C
 10   DRMX=DR
      DRMN=DR
      FNMX=FN
      FNMN=FN
      IF(IFLAGZ.GT.MXSTCK) IFLAGZ=MXSTCK
      DO 200 I=1,IFLAGZ
      IF(DRSTCK(I).GT.DRMX) DRMX=DRSTCK(I)
      IF(DRSTCK(I).LT.DRMN) DRMN=DRSTCK(I)
      IF(FNSTCK(I).GT.FNMX) FNMX=FNSTCK(I)
 200  IF(FNSTCK(I).LT.FNMN) FNMN=FNSTCK(I)
C
      DRERR=DRMX-DRMN
      FNERR=FNMX-FNMN
C
      RETURN
      END
```

```
C
C
C
        SUBROUTINE FFT(IDIR,LN,F)
C
C       FAST FOURIER TRANSFORM ALGORITHM
C
C       IDIR=0 FORWARD TRANSFORM
C           =1 INVERSE TRANSFORM
C       LN  =THE BLOCKSIZE IS 2**LN
C       F   =COMPLEX ARRAY CONTAINING ARRAY TO BE TRANSFORMED ON INPUT
C            AND CONTAINS TRANSFORM ON OUTPUT
C
C       FOR 1 CHANNEL FORWARD TRANSFORM, THE TIME DATA GOES INTO THE
C          REAL PART OF F.  ON OUTPUT, THE TRANSFORM IS PROPERLY ORDERED.
C       FOR 2 CHANNEL FORWARD TRANSFORM, THE TIME DATA FOR CHANNEL 1
C          GOES INTO THE REAL PART OF F AND CHANNEL 2 INTO THE
C          IMAGINARY PART OF F.  ON OUTPUT, THE TRANSFORMS ARE JUMBLED.
C          USE FUNCTION TWOSP IN THIS LIBRARY TO UNJUMBLE THE TRANSFORM.
C
C       THIS ROUTINE HAS NOT BEEN USED FOR 2 CHANNEL INVERSE TRANSFORM.
C          THEREFORE, ITS APPLICABILITY IS NOT KNOWN.
C
        COMPLEX F,U,W,T,CMPLX,CONJG
        DIMENSION F(1)
        PI=3.141592654
        N=2**LN
        IF(IDIR.EQ.0)GO TO 7
        DO 8 I=1,N
8       F(I)=CONJG(F(I))
7       NV2=N/2
        NM1=N-1
        J=1
        DO 3 I=1,NM1
        IF(I.GE.J)GO TO 1
        T=F(J)
        F(J)=F(I)
        F(I)=T
1       K=NV2
2       IF(K.GE.J)GO TO 3
        J=J-K
        K=K/2
        GO TO 2
3       J=J+K
        DO 5 L=1,LN
        LE=2**L
        LE1=LE/2
        U=(1.0,0.0)
        W=CMPLX(COS(PI/LE1),-SIN(PI/LE1))
        DO 5 J=1,LE1
        DO 4 I=J,N,LE
        IP=I+LE1
        T=F(IP)*U
        F(IP)=F(I)-T
4       F(I)=F(I)+T
5       U=U*W
        IF(IDIR.EQ.1)RETURN
        DO 6 I=1,N
```

```
6       F(I)=F(I)/FLOAT(N)
        RETURN
        END
```

```
C
C
C
        SUBROUTINE DECOMP(NDIM,N,A,COND,IPVT,WORK)
C
        IMPLICIT REAL *8(A-H,O-Z)
        IMPLICIT INTEGER *4(I-N)
        INTEGER *4 NDIM,N
        REAL *8 A(NDIM,N),COND,WORK(N)
        INTEGER *4 IPVT(N)
C
C       DECOMPOSES A REAL MATRIX BY GAUSSIAN ELIMINATION
C       AND ESTIMATES THE CONDITION OF THE MATRIX
C
C       USE SOLVE TO COMPUTE SOLUTIONS TO LINEAR SYSTEMS.
C
C       INPUT..
C
C       NDIM=DECLARED ROW DIMENSION OF THE ARRAY CONTAINING A.
C
C       N=ORDER OF THE MATRIX
C
C       A=MATRIX TO BE TRIANGULARIZED
C
C       OUTPUT..
C
C       A CONTAINS AN UPPER TRIANGULAR MATRIX U AND A PERMUTED
C       VERSION OF A LOWER TRIANGULAR MATRIX I-L SO THAT
C       (PERMUTATION MATRIX)*A=L*U
C
C       COND= AN ESTIMATE OF THE CONDITION OF A.
C       FOR THE LINEAR SYSTEM A*X=B, CHANGES IN A AND B
C       MAY CAUSE CHANGES COND TIMES AS LARGE IN X.
C       IF COND+1.0 .EQ. COND, A IS SINGULAR TO WORKING
C       PRECISION. COND IS SET TO 1.0E+32 IF EXACT
C       SINGULARITY IS DETECTED.
C       IPVT=THE PIVOT VECTOR
C       IPVT(K)=THE INDEX OF THE K-TH PIVOT ROW
C       IPVT(N)=(-1)**(NUMBER OF INTERCHANGES)
C
C       WORK SPACE..  THE VECTOR WORK MUST BE DECLARED AND INCLUDED
C       IN THE CALL.  ITS INPUT CONTENTS ARE IGNORED.
C       ITS OUTPUT CONTENTS ARE USUALLY UNINPORTANT.
C
C       THE DETERMINANT OF A CAN BE OBTAINED ON OUTPUT BY
C       DET(A)=IPVT(N)*A(1,1)*A(2,2)*...*A(N,N).
C
        REAL *8 EK,T,ANORM,YNORM,ZNORM
        INTEGER *4 NM1,I,J,K,KP1,KB,KM1,M
C
        IPVT(N)=1
        IF(N.EQ.1) GO TO 80
        NM1=N-1
C
C       COMPUTE 1-NORM OF A
C
        ANORM=0.0
        DO 10 J=1,N
```

```
            T=0.0
            DO 5 I=1,N
            T=T+DABS(A(I,J))
    5       CONTINUE
            IF(T.GT.ANORM) ANORM=T
    10  CONTINUE
C
C       GAUSSIAN ELIMINATION WITH PARTIAL PIVOTING
C
        DO 35 K=1,NM1
        KP1=K+1
C
C       FIND PIVOT
C
        M=K
        DO 15 I=KP1,N
        IF(DABS(A(I,K)).GT.DABS(A(M,K))) M=I
    15  CONTINUE
        IPVT(K)=M
        IF(M.NE.K) IPVT(N)=-IPVT(N)
        T=A(M,K)
        A(M,K)=A(K,K)
        A(K,K)=T
C
C       SKIP STEP IF PIVOT IS ZERO
C
        IF(T.EQ.0.0) GO TO 35
C
C       COMPUTE MULTIPLIERS
C
        DO 20 I=KP1,N
        A(I,K)=-A(I,K)/T
    20  CONTINUE
C
C       INTERCHANGE AND ELIMINATE BY COLUMNS
C
        DO 30 J=KP1,N
        T=A(M,J)
        A(M,J)=A(K,J)
        A(K,J)=T
        IF(T.EQ.0.0) GO TO 30
        DO 25 I=KP1,N
        A(I,J)=A(I,J)+A(I,K)*T
    25  CONTINUE
    30  CONTINUE
    35  CONTINUE
C
C       COND=(1-NORM OF A)*(AN ESTIMATE OF 1-NORM OF A-INVERSE)
C       ESTIMATE OBTAINED BY ONE STEP OF INVERSE ITERATION FOR THE
C       SMALL SINGULAR VECTOR.  THIS INVOLVES SOLVING TWO SYSTEMS
C       OF EQUATIONS, (A-TRANSPOSE)*Y=E AND A*Z=Y WHERE E
C       IS AVECTOR OF +1 OR -1 CHOSEN TO CAUSE GROWTH IN Y.
C       ESTIMATE=(1-NORM OF Z)/(1-NORM OF Y)
C
C       SOLVE (A-TRANSPOSE)*Y=E
C
        DO 50 K=1,N
        T=0.0
```

```
         IF(K.EQ.1) GO TO 45
         KM1=K-1
         DO 40 I=1,KM1
         T=T+A(I,K)*WORK(I)
40       CONTINUE
45       EK=1.0
         IF(T.LT.0.0) EK=-1.0
         IF(A(K,K).EQ.0.0) GO TO 90
         WORK(K)=-(EK+T)/A(K,K)
50       CONTINUE
         DO 60 KB=1,NM1
         K=N-KB
         T=0.0
         KP1=K+1
         DO 55 I=KP1,N
         T=T+A(I,K)*WORK(K)
55       CONTINUE
         WORK(K)=T
         M=IPVT(K)
         IF(M.EQ.K) GO TO 60
         T=WORK(M)
         WORK(M)=WORK(K)
         WORK(K)=T
60       CONTINUE
C
         YNORM=0.0
         DO 65 I=1,N
         YNORM=YNORM+DABS(WORK(I))
65       CONTINUE
C
C        SOLVE A*Z=Y
C
         CALL SOLVE(NDIM,N,A,WORK,IPVT)
C
         ZNORM=0.0
         DO 70 I=1,N
         ZNORM=ZNORM+DABS(WORK(I))
70       CONTINUE
C
C        ESTIMATE CONDITION
C
         COND=ANORM*ZNORM/YNORM
         IF(COND.LT.1.0) COND=1.0
         RETURN
C
C        1-BY 1
C
80       COND=1.0
         IF(A(1,1).NE.0.0) RETURN
C
C        EXACT SINGULARITY
C
90       COND=1.0E+32
         RETURN
         END
```

```
C
C
C
         SUBROUTINE SOLVE(NDIM,N,A,B,IPVT)
C
         IMPLICIT REAL *8 (A-N,O-Z)
         IMPLICIT INTEGER *4 (I-N)
         INTEGER *4 NDIM,N,IPVT(N)
         REAL *8 A(NDIM,N),B(N)
C
C        SOLUTION OF LINEAR SYSTEM, A*X=B
C        DO NOT USE IF DECOMP HAS DETECTED SINGULARITY
C
C        INPUT...
C
C        NDIM=DECLARED ROW DIMENSION OF ARRAY CONTAINING A
C
C        N=ORDER OF MATRIX
C
C        A=TRIANGULARIZED MATRIX OBTAINED FROM DECOMP
C
C        B=RIGHT HAND SIDE VECTOR
C
C        IPVT=PIVOT VECTOR OBTAINED FROM DECOMP
C
C        OUTPUT...
C
C        B=SOLUTION VECTOR, X.
C
         INTEGER *4 KB,KM1,NM1,KP1,I,K,M
         REAL *8 T
C
C        FORWARD ELIMINATION
C
         IF(N.EQ.1) GO TO 50
         NM1=N-1
         DO 20 K=1,NM1
         KP1=K+1
         M=IPVT(K)
         T=B(M)
         B(M)=B(K)
         B(K)=T
         DO 10 I=KP1,N
         B(I)=B(I)+A(I,K)*T
10       CONTINUE
20       CONTINUE
C
C        BACK SUBSTITUTION
C
         DO 40 KB=1,NM1
         KM1=N-KB
         K=KM1+1
         B(K)=B(K)/A(K,K)
         T=-B(K)
         DO 30 I=1,KM1
         B(I)=B(I)+A(I,K)*T         50      B(1)=B(1)/A(1,1)
30       CONTINUE                           RETURN
40       CONTINUE                           END
```

APPENDIX D

LISTING OF THE CODE TLAP

```
C
C **          IT SOLVES THE NON LINEAR EQUATIONS REPRESENTING A BWR
C **          WITH NON-LINEAR FEEDBACK
C **          AND RECIRCULATION LOOP      (ONLY ONE CHANNEL)
C
C **          IT INCLUDES 2-NODE FUEL AND POINT KINETICS
C -----------------------------------------------------------------
C
        PARAMETER NEQMX=20
        PARAMETER MXND=12
        IMPLICIT REAL*8 (A-H,O-Z)
        COMMON /MODEL/
     >         NND,A1,A2,ETA,DZ,XKF,PSHP
     >        ,XLA,DPFO,DPO,FLOW
     >        ,H,RHOL,RHOG,GC,XKC
     >        ,XKS,RCOEF,HL,HG,AFLOW
     >        ,WHITE,IMPUL,STEP,HSIN
        REAL *8 A1,A2,H,RHOL,RHOG,GC,XKF,XKC
     >        ,XLA,DPFO,DPO,FLOW
     >        ,XKS,RCOEF,ETA,HL,HG,PSHP(MXND),DZ
        LOGICAL *1 WHITE,IMPUL,STEP,HSIN
C -----------------------------------------------------------------
        COMMON /BOUND/ YZ,XZ,FHZ,ECZ,FR,QDOT
        REAL *8 YZ(O:MXND,3),XZ(O:MXND),FHZ(O:MXND),ECZ(O:MXND)
     >        ,FR(1:MXND),QDOT(1:MXND)
        COMMON /STEADY/ YZO,XZO,ECZO
        REAL *8 YZO(O:MXND,3),XZO(O:MXND),ECZO(O:MXND)
C -----------------------------------------------------------------
        EXTERNAL F
        REAL *8 ATOL(NEQMX),Y(NEQMX)
        INTEGER ICH(NEQMX+1)
        COMMON /PERTUR/ R,WPERT,PHPERT
        LOGICAL *1 FILE(28)
C -----------------------------------------------------------------
        COMMON /NEUT/ XL,GT,BETA,F3,DOP,REACO
        REAL *8 REACO(MXND)
        COMMON /HTR/ U,A3,A4,A5,POW,PSUM2,DRCM
        COMMON /IN2/ POWPC,FLOWPC
C -----------------------------------------------------------------
        COMMON /OUTP/ PRINT,PRALL,DSK
        LOGICAL *1 PRINT,PRALL,DSK
        COMMON /UNIT4/ DEV
        LOGICAL *1 DEV(28)
C -----------------------------------------------------------------
C
        CALL INPUT(NEQ,F,JAC,DT,T,Y,ATOL,RTOL,MF,FILE,ICH,NCH)
C
  10    CALL INPF(FILE)
C
        CALL STEADY(DT,Y,FILE)
C
        CALL DTGEN(NEQ,F,JAC,DT,T,Y,ATOL,RTOL,MF,FILE,ICH,NCH)
C
        GO TO 10
        END
```

```
C
C
C
        SUBROUTINE INPUT(NEQ,F,JAC,DT,T,Y,ATOL,RTOL,MF,FILE,ICH,NCH)
        PARAMETER NEQMX=20
        PARAMETER MXND=12
        IMPLICIT REAL*8 (A-H,O-Z)
        COMMON /MODEL/
     >         NND,A1,A2,ETA,DZ,XKF,PSHP
     >        ,XLA,DPFO,DPO,FLOW
     >        ,H,RHOL,RHOG,GC,XKC
     >        ,XKS,RCOEF,HL,HG,AFLOW
     >        ,WHITE,IMPUL,STEP,HSIN
        REAL *8 A1,A2,H,RHOL,RHOG,GC,XKF,XKC
     >        ,XLA,DPFO,DPO,FLOW
     >        ,XKS,RCOEF,ETA,HL,HG,PSHP(MXND),DZ
        LOGICAL *1 WHITE,IMPUL,STEP,HSIN
C -----------------------------------------------------------------
        COMMON /BOUND/ YZ,XZ,FHZ,ECZ,FR,QDOT
        REAL *8 YZ(O:MXND,3),XZ(O:MXND),FHZ(O:MXND),ECZ(O:MXND)
     >        ,FR(1:MXND),QDOT(1:MXND)
        COMMON /STEADY/ YZO,XZO,ECZO
        REAL *8 YZO(O:MXND,3),XZO(O:MXND),ECZO(O:MXND)
C -----------------------------------------------------------------
        EXTERNAL F
        REAL *8 ATOL(NEQMX),Y(NEQMX)
        INTEGER ICH(NEQMX+1)
        COMMON /PERTUR/ R,WPERT,PHPERT
        LOGICAL *1 FILE(28)
C -----------------------------------------------------------------
        COMMON /NEUT/ XL,GT,BETA,F3,DOP,REACO
        REAL *8 REACO(MXND)
        COMMON /HTR/ U,A3,A4,A5,POW,PSUM2,DRCM
        COMMON /IN2/ POWPC,FLOWPC
C -----------------------------------------------------------------
        COMMON /OUTP/ PRINT,PRALL,DSK
        LOGICAL *1 PRINT,PRALL,DSK
        COMMON /UNIT4/ DEV
        LOGICAL *1 DEV(28)
        LOGICAL *1 IANS,FILE6(28)
C -----------------------------------------------------------------
C
C
  1     WRITE(5,1000)
 1000   FORMAT(////' ** PROGRAN TLAP **'//
     >  ' ENTER NUMBER OF NODES [DEF=12]  :',$)
        READ(4,1010) NND
 1010   FORMAT(8I10)
        IF(NND.LE.0) NND=12
        IF(NND.GT.MXND) GO TO 1
        NEQ=NND+1+2+1+1 ' ALFA, INLET FLOW, 2 TEMP, NEUTR, DELAYED
        WRITE(5,1020)
 1020   FORMAT(' DELTAT [DEF=0.02]      :',$)
        READ(4,1030) DT
 1030   FORMAT(4F20.0)
        IF(DT.LE.0.) DT=0.02
        WRITE(5,1040)
 1040   FORMAT(' TOTAL TIME [DEF=200]   :',$)
```

```
        READ(4,1030) T                                               IF(IANS.EQ.'Y') DSK=.TRUE.
        IF(T.LE.0.) T=200.                                           WRITE(5,2060)
        WRITE(5,1070)                                      2060      FORMAT(' DEVICE FOR STEADY AND DR OUTPUT [DEF=CL:]:',$)
1070    FORMAT(' ENTER  1 - FOR WHITE NOISE PERT.'/                  READ(4,2030) NCHR,(DEV(J),J=1,NCHR)
   >    ,'        2 - FOR IMPULSE RESP.'/                             IF(NCHR.EQ.0) DEV(1)='C'
   >    ,'        3 - FOR DAMPED SINE RESPONSE. [DEF]'/               IF(NCHR.EQ.0) DEV(2)='L'
   >    ,'        4 - FOR STEP RESP.              :',$)              IF(NCHR.EQ.0) NCHR=2
        READ(4,1010) IOPT                                            IF(DEV(NCHR).EQ.':') NCHR=NCHR-1
        IF(IOPT.LE.0.OR.IOPT.GT.4) IOPT=3                            DEV(NCHR+1)=':'
        WHITE=.FALSE.                                                DEV(NCHR+2)=0
        IMPUL=.FALSE.                                                CLOSE(UNIT=6)
        STEP=.FALSE.                                                 CALL ASSIGN(6,DEV,NCHR+1)
        IF(IOPT.EQ.1) WHITE=.TRUE.                         C
        IF(IOPT.EQ.2) IMPUL=.TRUE.                                   WRITE(5,2040)
        IF(IOPT.EQ.3) HSIN=.TRUE.                          2040      FORMAT(' DEVICE FOR POWER AND FLOW INPUT [DEF=TI:] :',$)
        IF(IOPT.GE.4) STEP=.TRUE.                                    READ(4,2030) NCHR,(DEV(J),J=1,NCHR)
C                                                                    IF(NCHR.EQ.0) DEV(1)='T'
        WRITE(5,1075)                                                IF(NCHR.EQ.0) DEV(2)='I'
1075    FORMAT(' PERTURBATION MAGNITUDE (DOLLARS) [DEF=0.1] :',$)    IF(NCHR.EQ.0) NCHR=2
        READ(4,1030) R                                               IF(DEV(NCHR).EQ.':') NCHR=NCHR-1
        IF(R.LE.0.) R=0.1                                            DEV(NCHR+1)=':'
        WRITE(5,1080)                                                DEV(NCHR+2)=0
1080    FORMAT(' ENTER ATOL,RTOL [DEF=1.E-8,1.E-4]:',$)             CLOSE(UNIT=4)
        READ(4,1030) ATOL(1),RTOL                                    CALL ASSIGN(4,DEV,NCHR+1)
        IF(ATOL(1).EQ.0.) ATOL(1)=1.E-8                    C
        IF(RTOL.EQ.0.) RTOL=1.E-4                                    MF=23              ! DIAGONAL JAC.  (NOT SUPLIED)
        WRITE(5,1090)                                                RETURN
1090    FORMAT(' ENTER DRC MULTIPLIER [DEF=1]:',$)                   END
        READ(4,1030) DRCM
        IF(DRCM.EQ.0.) DRCM=1.
        WRITE(5,2000)
2000    FORMAT(' POWER SHAPE [BOT,...,TOP]')
        F1=FLOAT(NND)+2.5
        F2=FLOAT(NND)+2.
        DO 110 I=1,NND
110     PSHP(I)=SQRT(ABS(SIN(3.14*((F1-FLOAT(I))/F2)**2)))
        DO 100 I=1,NND
        WRITE(5,2010) I,PSHP(I)
2010    FORMAT(' PSHP(',I2,') = [DEF=',G12.2,'] :',$)
        READ(4,1030) PPP
        IF(PPP.NE.0.) PSHP(I)=PPP
100     CONTINUE
C
        PRINT=.FALSE.
        PRALL=.FALSE.
        DSK=.FALSE.
        WRITE(5,2020)
2020    FORMAT(' PRINT PROGRESS? [DEF=NO] :',$)
        READ(4,2030) I,IANS
2030    FORMAT(Q,80A1)
        IF(IANS.EQ.'Y') PRINT=.TRUE.
        IF(PRINT) WRITE(5,2025)
2025    FORMAT(' PRINT ALL VARIABLES? [DEF= ONLY POWER] :',$)
        IF(PRINT)      READ(4,2030) I,IANS
        IF(IANS.EQ.'Y') PRALL=.TRUE.
        WRITE(5,2050)
2050    FORMAT(' SAVE ALL VARIABLES IN DISK? [DEF=NO] :',$)
        READ(4,2030) I,IANS
```

266

```
C
C
C
        SUBROUTINE INPF(FILE)
        PARAMETER NEQMX=20
        PARAMETER MXND=12
        IMPLICIT REAL*8 (A-H,O-Z)
        COMMON /MODEL/
        >          NND,A1,A2,ETA,DZ,XKF,PSHP
        >          ,XLA,DPFO,DPO,FLOW
        >          ,H,RHOL,RHOG,GC,XKC
        >          ,XKS,RCOEF,HL,HG,AFLOW
        >          ,WHITE,IMPUL,STEP,HSIN
        REAL *8 A1,A2,H,RHOL,RHOG,GC,XKF,XKC
        >          ,XLA,DPFO,DPO,FLOW
        >          ,XKS,RCOEF,ETA,HL,HG,PSHP(MXND),DZ
        LOGICAL *1 WHITE,IMPUL,STEP,HSIN
C -----------------------------------------------------------------
        COMMON /BOUND/ YZ,XZ,FHZ,ECZ,FR,QDOT
        REAL *8 YZ(0:MXND,3),XZ(0:MXND),FHZ(0:MXND),ECZ(0:MXND)
        >          ,FR(1:MXND),QDOT(1:MXND)
        COMMON /STEADY/ YZO,XZO,ECZO
        REAL *8 YZO(0:MXND,3),XZO(0:MXND),ECZO(0:MXND)
C -----------------------------------------------------------------
        COMMON /NEUT/ XL,GT,BETA,F3,DOP,REACO
        REAL *8 REACO(MXND)
        COMMON /PERTUR/ R,WPERT,PHPERT
        COMMON /HTR/ U,A3,A4,A5,POW,PSUM2,DRCM
        COMMON /IN2/ POWPC,FLOWPC
        COMMON /UNIT4/ DEV
        LOGICAL *1 DEV(28)
C
C -----------------------------------------------------------------
C
        LOGICAL *1 FILE(28)
C
C -----------------------------------------------------------------
C
        IF(DEV(1).EQ.'T')        WRITE(5,1000)    ! TERMINAL INPUT
 1000   FORMAT(' POWER, FLOW [%,DEF=100%]        :',$)
        READ(4,1010) POWPC,FLOWPC
 1010   FORMAT(6F20.0)
        IF(POWPC.EQ.0.) POWPC=100.
        IF(FLOWPC.EQ.0.) FLOWPC=100.
C
        IP=IFIX(SNGL(POWPC))
        IF=IFIX(SNGL(FLOWPC))
        IF(IP.EQ.0.AND.IF.EQ.0) STOP
        ENCODE(8,1030,FILE(1)) IP,IF
 1030   FORMAT('P',I3,'F',I3)
C **           FILE FOR STEADY STATE INFORMATION  **
        FILE(9)='.'
        FILE(10)='S'
        FILE(11)='T'
        FILE(12)='D'
        FILE(13)=0
        CLOSE(UNIT=6)
        OPEN(UNIT=6,NAME=FILE,TYPE='NEW',ACCESS='SEQUENTIAL'
```

```
        >           ,FORM='FORMATTED',SHARED)
        CLOSE(UNIT=1)
        FILE(9)=0                    ! .DAT
C
        RETURN
        END
```

267

```
C                                                                      FLOW=FLOWPC*FLO100/100.        ! Kg/sm2
C                                                                      DZ=H/FLOAT(NND)
C                                                                      SUM=0.
       SUBROUTINE STEADY(DT,Y)                                         DO 100 I=1,NND
C ---------------------------------------------------------     100    SUM=SUM+PSHP(I)
       PARAMETER NEQMX=20                                              SUM=SUM/FLOAT(NND)
       PARAMETER MXND=12                                               PSUM2=0.
       IMPLICIT REAL*8 (A-H,O-Z)                                       DO 110 I=1,NND
       REAL *8 Y(1:NEQMX)                                              PSHP(I)=POW*PSHP(I)/SUM        ! J/ms
       COMMON /MODEL/                                                  PSUM2=PSUM2+PSHP(I)**2        ! FOR DRC WEIGHTING
     >          NND,A1,A2,ETA,DZ,XKF,PSHP                       110    CONTINUE
     >         ,XLA,DPFO,DPO,FLOW                              C
     >         ,H,RHOL,RHOG,GC,XKC                             C
     >         ,XKS,RCOEF,HL,HG,AFLOW                          C
     >         ,WHITE,IMPUL,STEP,HSIN                                  YZ(0,1)=0.0     ! ALFA 0
       REAL *8 A1,A2,H,RHOL,RHOG,GC,XKF,XKC                            YZ(NND,2)=0.0   ! P(H)  UPPER PLENUM PRESS. CONSTANT
     >         ,XLA,DPFO,DPO,FLOW                                      YZ(0,3)=FLOW    ! G0
     >         ,XKS,RCOEF,ETA,HL,HG,PSHP(MXND),DZ                      XZ(0)=0.0       ! X0
       LOGICAL *1 WHITE,IMPUL,STEP,HSIN                                A1=1./(DZ*(RHOL-RHOG))
       DATA    H/3.65/        ! CORE HEIGTH                            A2=(HG*RHOG-HL*RHOL)
     >         ,RHOL/0.74E3/  ! SATURATED WATER DENSITY  Kg/m3         ETA=(RHOL-RHOG)/RHOL
     >         ,RHOG/0.035E3/ ! SATURATED STEAM DENSITY  Kg/m3         DZ=H/FLOAT(NND)
     >         ,GC/9.81/      ! g  (m/s2)                              FMOODY=0.019*1.4          ! FOR Re=70000   TURBULENT REGION APROX CONST.
     >         ,XKC/30.0/     ! Kc for Martinelli Nelson (see Pedros th  DE=0.0134             ! m
     >         ,XKS/0.8/      ! Ks                                     XJONES=1.9-5.E-4*FLOW   ! JONES CORRECTION FACTOR FOR MARTINELLI-NELSON
     >         ,RCOEF/3.97/   ! r    "     "                           XKF=XJONES*FMOODY/(DE*2.*RHOL)  ! SINGLE PHASE FRICTION COEFF.
     >         ,HL/1.25E6/    ! Sat. water enthalpy   (J/m3)    C
     >         ,HG/2.77E6/    ! Sat. steam enthalpy   (J/m3)    C
     >         ,AFLOW/3.98/   ! Flow area (m2)
     >         ,XLA/4.0/      ! L/A ratio for R.L.    (m-1)            DO 120 I=1,NND
C ---------------------------------------------------------             YZ(I,3)=FLOW            !G(ZI)
       COMMON /BOUND/ YZ,XZ,FHZ,ECZ,FR,QDOT                            XZ(I)=XZ(I-1)+DZ*PSHP(I)/(AFLOW*FLOW*(HG-HL))
       REAL *8 YZ(0:MXND,3),XZ(0:MXND),FHZ(0:MXND),ECZ(0:MXND)  120    YZ(I,1)=XZ(I)           ! FIRST GUESS
     >         ,FR(1:MXND),QDOT(1:MXND)                         C
       COMMON /STEADY/ YZO,XZO,ECZO                             C **    ITERATION FOR ALFA CONVERGENCE
       REAL *8 YZO(0:MXND,3),XZO(0:MXND),ECZO(0:MXND)           C
C ---------------------------------------------------------             K=0
       DATA POW100/0.45E9/        ! J/ms thermal U.Y.                   CONU=FLOAT(NND)*1.E-10
       DATA FLO100/1.5E3/         ! Kg/sm2      U.Y.            10     SUM=0.0
       COMMON /NEUT/ XL,GT,BETA,F3,DOP,REACO                           DO 200 IN=1,NND
       REAL *8 REACO(MXND)                                             ALF=YZ(IN,1)
       DATA    XL/0.08/           ! DEALYED N. LAMBDA                  IF(ALF.LT.0.) ALF=0.            ! ALF <0 NOT ALLOWED
     >         ,GT/4.E-5/         ! GENERATION TIME (s)                SLIP=( 1.-ALF )/( XKS-ALF+( 1.-XKS )*(ALF**SNGL(RCOEF)) )
     >         ,BETA/0.0056/      ! DELAYED N. BETA                    YZ(IN,1)=XZ(IN)/(SLIP*(1.-ETA)+XZ(IN)*(1.-SLIP*(1.-ETA)))
     >         ,DOP/-1.4E-5/      ! DOPPLER REACTIV. COEFF (Kelvin-1)  200    SUM=SUM+ABS(ALF-YZ(IN,1))
     >         ,A3/0.2325/        ! FUEL TIME CONSTANT (FROM LAPUR FIT) s-1   K=K+1
     >         ,A4/0.048/         ! A4=A3*(RHOCPFUEL)/(RHOCPCLAD)      IF(K.GT.100) STOP ' STEADY -- TOO MANY ALFA ITERATIONS'
     >         ,A5/7.54/          ! CLADING TIME CONSTANT (LAPUR)  s-1 IF(SUM.GT.CONU) GO TO 10
     >         ,U/2.96E8/         ! SEE BWR 5  (2-NODE HEAT TR. COEFF)  C
       COMMON /HTR/ U,A3,A4,A5,POW,PSUM2,DRCM                   C
C ---------------------------------------------------------             DO 210 IN=1,NND
       COMMON /IN2/ POWPC,FLOWPC                                       ALF=(YZ(IN,1)+YZ(IN-1,1))/2.    ! NODAL ALFA
C ---------------------------------------------------------             REACO(IN)=-10.8*ALF-20.7/2.*ALF**2+14.03/3.*ALF**3
C                                                              >        -13.54/4.*ALF**4         ! R=INT( DR/DALF * DLAF)
C                                                              210    REACO(IN)=0.97*DRCM*REACO(IN)/100.    ! NODAL REACTIVITY (m-1)
       POW=POWPC*POW100/100.            ! J/ms  thermal         C            0.97 ADJUSTED SO UY 7N IS THE START OF LIMIT CYCLE OPERATION
       RCPUF=6.62*3.125E6              ! J/K  (BWR5)            C
       F3=POW*H/RCPUF                   ! FOR USE IN FUEL EQ.    C
                                                                       DO 300 IZ=0,NND
```

268

```
C
C **     COMPTUTE XZ COMPATIBLE WITH THE CONVERGED ALFA
C
        ALF=YZ(IZ,1)
        G=YZ(IZ,3)
        SLIP=( 1.-ALF )/( XKS-ALF+( 1.-XKS )*ALF**RCOEF )
        XZ(IZ)=ALF*SLIP*(1.-ETA) /( 1.-ALF*( 1.-SLIP*( 1.-ETA ) ) )
C
        FHZ(IZ)=G*( HL+( HG-HL )*XZ(IZ) )          ' FLOW ENTHALPY
        ECZ(IZ)=0.                                 ' KINETIC ENERGY
        IF(ALF.NE.1.)
     >  ECZ(IZ)=ECZ(IZ)+.5*(G**2)*( (1.-XZ(IZ))**2/( RHOL*(1.-ALF )))
        IF(ALF.NE.0.)
     >  ECZ(IZ)=ECZ(IZ)+.5*(G**2)*((XZ(IZ)**2)/(RHOG*ALF) )
C
        ECZO(IZ)=ECZ(IZ)
        XZO(IZ)=XZ(IZ)
  300   CONTINUE
C
C
        DO 400 IN=NND,1,-1
        XI=(XZ(IN)+XZ(IN-1))/2.                     ' NODE AVERG. QUALITY
        FR(IN)=XKF*(1.+31.*XI)*(FLOW**2)            ' FRICTION
        IF(IN.EQ.1) FR(IN)=FR(IN)+XKC*(YZ(0,3)**2)/(2.*DZ*RHOL) ' ENTR. ORIF
C
        YZ(IN-1,2)=YZ(IN,2) + (ECZ(IN)-ECZ(IN-1))
     >  +DZ*(RHOL-(RHOL-RHOG)*Y(IN))*GC +DZ*FR(IN) ! PRESSURE Nw/m2
  400   CONTINUE
C
C
        DO 500 J=1,3
        DO 500 I=0,NND
        YZO(I,J)=YZ(I,J)
        YZ(I,J)=0.0
  500   CONTINUE
        DO 510 I=1,NEQMX
        Y(I)=0.
  510   CONTINUE
C
C **     THE RECIRC. LOOP CONSTANTS ARE CALCULATED SO THAT THE NATURAL
C **     CIRCULATION TIME CONSTANT BE APPROX. 0.3 SEC
C **     THE FRICTION IS ASSUMED PROPORTIONAL TO THE FLOW **2
C
        RLTNC=0.3            ! R.L. TIME CONSTANT AT NAT. CIRCULATION (APPROX)
        DPFNC=FLOW*AFLOW*XLA/(2.*RLTNC) ! FRICTION DP AT NAT. CIRC.
        DPFO=DPFNC*(FLOWPC/32.)**2      ' 32xF = NAT. CIRC.
        DPO=-YZO(NND,2)                 ' DPO=PO-PN
C
C
        LUN=6
  111   WRITE(LUN,2000) POW,POWPC,FLOW,FLOWPC
 2000   FORMAT(1H1//' POWER = ',G20.5,' J/s  -- ',F8.3,' %'/
     >             ' FLOW  = ',G20.5,' Kg/s -- ',F8.3,' %'///
     >  //' NODAL VALUES '/' '
     >  ,13X
     >  ,' ALFA         DRC      REACTIVITY   PRESSURE   POWER SHAPE'
     >  ,/13X
     >  ,'                                   Nw/m2        J/ms')
```

```
        DO 700 I=1,NND
        ALF=(YZO(I,1)+YZO(I-1,1))/2.
        DRC=-10.8-20.7*ALF+14.03*ALF**2-13.54*ALF**3 ' DRC FIT LAPUR XSEC
        DRC=0.97*DRCM*DRC/100.      ' LAPUR UNITS = %K/K
        PP=(YZO(I,2)+YZO(I-1,2))/2.
        WRITE(LUN,2050) I,ALF,DRC,REACO(I),PP,PSHP(I)
  700   CONTINUE
        WRITE(LUN,2020)
 2020   FORMAT(////' BOUNDARY VALUES '/' ',13X
     >  ,' QUALITY          ALFA            PRESSURE        FLOW'/
     >  ,13X
     >  ,'                                  Nw/m2           Kg/sm2')
        DO 710 I=0,NND
        WRITE(LUN,2030) I,XZO(I),(YZO(I,J),J=1,3)
 2030   FORMAT(' ',I3,3X,4G16.3)
  710   CONTINUE
        WRITE(LUN,2040)
 2040   FORMAT(////' BOUNDARY VALUES '/' '
     >  ,13X
     >  ,' ECZ            FHZ            FR        KINETIC DP      HEAD DP'
     >  ,/13X
     >  ,' Nw/m2          J/sm2         Nw/m2       Nw/m2         Nw/m2')
        DO 711 I=0,NND
        IF(I.NE.0)      P1=( ECZ(I)-ECZ(I-1) )
        IF(I.NE.0)      P2=( RHOL-(RHOL-RHOG)*Y(I) )*GC*DZ
        IF(I.EQ.0)      P30=XKC*(YZO(0,3)**2)/(2.*RHOL)
        IF(I.EQ.0)      P3=DZ*FR(I)
        IF(I.EQ.1)      P3=P3-P30 ! ONLY FIRST NODE CONTRIBUTION (NO INLET)
        IF(I.EQ.0)      WRITE(LUN,2050) I,ECZ(I),FHZ(I),P30
        IF(I.NE.0)      WRITE(LUN,2050) I,ECZ(I),FHZ(I),P3,P1,P2
 2050   FORMAT(' ',I3,3X,5G14.3)
  711   CONTINUE
C
        WRITE(LUN,2060)
 2060   FORMAT(////' BOUNDARY VALUES '/' ',13X
     >  ,'V(LIQ)        V(STEAM)        SLIP'
     >  ,/13X
     >  ,' m/s           m/s')
        DO 712 I=0,NND
        ALF=YZO(I,1)
        G=YZO(I,3)
        SLIP=( 1.-ALF )/( XKS-ALF+( 1.-XKS )*ALF**RCOEF )
        UL=G*(1.-XZ(I))/(RHOL*(1.-ALF))
        US=SLIP*UL
        IF(I.NE.0)      WRITE(LUN,2050) I,UL,US,SLIP
        IF(I.EQ.0)      WRITE(LUN,2050) I,UL
  712   CONTINUE
C
        RLG=FLOW/(2.*DPFO)
        RLT=RLG*AFLOW*XLA
        WRITE(LUN,2070) RLG,RLT
 2070   FORMAT(//' If linearized,'/
     >  ,' Recirc. Loop Gain         : ',G14.3,' (Kg/sm2)/(Nw/m2)'/
     >  ,' Recirc. Loop Time Constant : ',G14.3,' s')
C
C
        WRITE(LUN,2080)
     >            NND,A1,A2,ETA,DZ,XKF
```

```
        >               ,(PSHP(I),I=1,MXND)
        >               ,XLA,DPFO,DPO,FLOW,H
        >               ,RHOL,RHOG,GC,XKC,XKS
        >               ,RCOEF,HL,HG,AFLOW
  2080   FORMAT(///' COMMON /MODEL/ '//I10,8(/5G15.4))
C
        RETURN
        END
```

```
C
C
C  ........................................................
C  _____
C  ........................................................
C
C **    THIS IS THE NONLINEAR MODEL OF A SINGLE CHANNEL BWR
C **    THE RECIRC. LOOP IS INCLUDED.
C **                                    24-AUG-83
C **    DP/DT IS ASSUMED VERY SMALL (NO ACUSTIC PHENOMENA)
C
        SUBROUTINE F(NEQ,T,Y,YDOT)
C ------------------------------------------------------------
        PARAMETER NEQMX=20
        PARAMETER MXND=12
        IMPLICIT REAL*8 (A-H,O-Z)
        REAL *8 Y(NEQMX),YDOT(NEQMX)
        COMMON /MODEL/
        >         NND,A1,A2,ETA,DZ,XKF,PSHP
        >         ,XLA,DPFO,DPO,FLOW
        >         ,H,RHOL,RHOG,GC,XKC
        >         ,XKS,RCOEF,HL,HG,AFLOW
        >         ,WHITE,IMPUL,STEP,HSIN
        REAL *8 A1,A2,H,RHOL,RHOG,GC,XKF,XKC
        >         ,XLA,DPFO,DPO,FLOW
        >         ,XKS,RCOEF,ETA,HL,HG,PSHP(MXND),DZ
        LOGICAL *1 WHITE,IMPUL,STEP,HSIN
C ------------------------------------------------------------
        COMMON /BOUND/ YZ,XZ,FHZ,ECZ,FR,QDOT
        REAL *8 YZ(0:MXND,3),XZ(0:MXND),FHZ(0:MXND),ECZ(0:MXND)
        >         ,FR(1:MXND),QDOT(1:MXND)
        COMMON /STEADY/ YZO,XZO,ECZO
        REAL *8 YZO(0:MXND,3),XZO(0:MXND),ECZO(0:MXND)
C ------------------------------------------------------------
        COMMON /PERTUR/ R,WPERT,PHPERT
        COMMON /PP/ PERT,TO,DTO,TOLD
        DATA DGDT/0./
C ------------------------------------------------------------
        COMMON /NEUT/ XL,GT,BETA,F3,DOP,REACO
        REAL *8 REACO(MXND)
        COMMON /HTR/ U,A3,A4,A5,POW,PSUM2,DRCM
C ------------------------------------------------------------
C
C **    YZ(I,J)        I=NODE BOUNDARY 0,...,NND
C **                   J=SIGNAL       1-ALFA, 2-P, 3-G
C
        COMMON /NOISE/ P
        REAL *8 P(0:20)
C
        IF(STEP) PERT=R
        IF(STEP) GO TO 10
        PERT=0.
        IF(TO.EQ.0..AND.IMPUL) PERT=2.*R*EXP(-3.*T/DTO)
        IF(IMPUL) GO TO 10
        IF(HSIN.AND. T.LT.1)
        > PERT=R*EXP(1.5-3*T)*SIN(3.141592654*SNGL(T))      ' 0.5 Hz
        IF(HSIN) GO TO 10
        IF(T.EQ.TOLD) GO TO 10
```

```
        TOLD=T                                                        >    + (YZ(NND,2)-YZ(0,2)) ) / (AFLOW*XLA)  ! RECIRC. LOOP
C **        ** BAND LIMITED NOISE INTERPOLATION **          C ------------------------------------------------------------
        DTD=DTO/10.             ! INTERMEDIATE DT                     DO 200 IZ=NND,1,-1
        TINT=SNGL((T-TO)/DTD)                                         YZ(IZ-1,2)=YZ(IZ,2) + (ECZ(IZ)-ECZ(IZ-1)) + DGDT*DZ
        II=IFIX(SNGL(TINT))                                      >    +DZ*(-(RHOL-RHOG)*Y(IZ))*GC +DZ*FR(IZ) ! PRESSURE NW/m2
        IF(II.GT.20) II=20      ! IF OUTSIDE RANGE EXTRAPOLATE  200   CONTINUE
        XINT=TINT-FLOAT(II)                                     C
        PERT=R*(P(II)*(1.-XINT)+P(II+1)*XINT)                   C
C                                                                     YDOT(NND+1)=(-DPFO*(2.*YZO(0,3)*YZ(0,3)+YZ(0,3)**2)/YZO(0,3)**2
C                                                                >    + (YZ(NND,2)-YZ(0,2)) ) / (AFLOW*XLA)  ! RECIRC. LOOP FLOW
C                                                               C
C **    NOTE, ALL THE VARIABLES ARE NORMALIZED TO THEIR         C
C **    STEADY STATE VALUE                                            UFB=0.
C                                                                     DO 250 I=1,NND
   10   XZ(0)=0.                 ! X(Z=0)                        C
        YZ(0,1)=0.               ! ALFA(Z=0)                    C **        REACT IS THE NODAL VOID REACTIVITY
        YZ(0,3)=Y(NND+1)         ! G(Z=0)                       C **        I.E. INT(DRC * DALFA)
        YZ(NND,2)=0.             ! P(H)                         C **        WITH DRC=DRHO/DALFA= -10.8-20.7A+14.03A2-13.54A3 'LAPUR FIT
C                                                               C
C                                                                     ALF=(YZO(I,1)+YZO(I-1,1))/2. + Y(I)
        GO2=2.*YZO(0,3)*YZ(0,3)+YZ(0,3)**2                      C
        DO 100 IN=1,NND                                         C **    NOTE: ALFA IS ALLOWED HERE TO BE >1 OR <0 .
        YZ(IN,1)=2.*Y(IN)-YZ(IN-1,1)    ! ALFA(Z)              C **    FOR REACTIVITY FEEDBACK POURPOSES, IF THERE IS SUPERHEATING,
        ALF=YZO(IN,1)+YZ(IN,1)                                  C **    THE STEAM DENSITY DECREASES.  THIS IS ONLY AN APPROXIMATION.
C                                                               C
C **            CORRECTION FOR SUBCOOLING OR SUPERHEATING             REACT=-10.8*ALF-20.7/2.*ALF**2+14.03/3.*ALF**3
C                                                               >        -13.54/4.*ALF**4 ! DRC FIT LAPUR XSEC
        IF(ALF.LT.1. .AND. ALF.GT.0.)                                REACT=0.97*DRCM*REACT/100.      ! LAPUR UNITS = %K/K
    >   SLIP=( 1.-ALF )/( XKS-ALF+( 1.-XKS )*ALF**RCOEF )            REACT=REACT-REACO(I)           ! CHANGE IN REACTIVITY
        IF(ALF.LT.1. AND. ALF.GT.0.)                            250  UFB=UFB+REACT*PSHP(I)**2       ! VOID FEEDBACK
    >   XZ(IN)=ALF*SLIP*(1.-ETA) /( 1.-ALF*( 1.-SLIP*( 1.-ETA ) ) )  UFB=UFB/PSUM2 !NOTE: UFB=INT(DRC*ALF(Z)*PSHP2*DZ)/INT(PSHP2*DZ)
        IF(ALF.GE.1.) XZ(IN)=1.                                      RHO=DOP*Y(NND+2) + UFB + PERT*BETA
        IF(ALF.LE.0.) XZ(IN)=0.                                      YDOT(NND+2)=F3*Y(NND+4)-A3*(Y(NND+2)-Y(NND+3))   ! FUEL
        IF(ALF.GE.1.) ALF=1.                                         YDOT(NND+3)=A4*(Y(NND+2)-Y(NND+3))-A5*Y(NND+3)   ! CLADDING
        IF(ALF.LE.0.) ALF=0.                                         YDOT(NND+4)=(RHO-BETA)*Y(NND+4)/GT+XL*Y(NND+5) + RHO/GT ! N
C                                                                     YDOT(NND+5)=BETA*Y(NND+4)/GT-XL*Y(NND+5)         ! DELAYED N
        XI=(XZ(IN)+XZ(IN-1))/2. ! NODE AVERG. QUALITY           C
        DXI=XI-(XZO(IN)+XZO(IN-1))/2.                           D     WRITE(5,5010) A3,U
        DG=(YZ(IN,3)+YZ(IN-1,3))/2.    ! NODE AVERAGE (G-GO)    D5010 FORMAT(' ',T20,2G20.5)
        G2=2.*FLOW*DG+DG**2           ! G**2-GO**2              D     WRITE(5,5000) T,(Y(I),YDOT(I),I=1,NEQ)
        FR(IN)=XKF*(G2*(1.+31.*XI)+31.*DXI*FLOW**2)             D5000 FORMAT(' ',G14.3,<NEQ>(/2G20.5))
        IF(IN.EQ.1) FR(IN)=FR(IN)+XKC*(GO2)/(2.*DZ*RHOL) ! ENTR. ORIF. C
C                                                               C
        G=YZO(IN,3)+YZ(IN,3)                                         DO 300 IN=1,NND
        ECZ(IN)=0.                             ! KINETIC ENERGY      QDOT(IN)=U*Y(NND+3)*PSHP(IN)/POW
        IF(ALF.LT.1.)                                               DENI=(RHOG*HG-RHOL*HL)
    >   ECZ(IN)=ECZ(IN)+.5*(G**2)*( (1.-XZ(IN))**2/( RHOL*(1.-ALF ))) >   +( HL*(1.-XZ(IN))+HG*XZ(IN) )*(RHOL-RHOG)
        IF(ALF.GT.0.)                                               GX1=YZO(IN-1,3)*(XZ(IN)-XZO(IN))+YZ(IN-1,3)*XZ(IN)
    >   ECZ(IN)=ECZ(IN)+.5*(G**2)*((XZ(IN)**2)/(RHOG*ALF) )         GX2=YZO(IN-1,3)*(XZ(IN-1)-XZO(IN-1))+YZ(IN-1,3)*XZ(IN-1)
        ECZ(IN)=ECZ(IN)-ECZO(IN)      ! NORMALIZED                  YDOT(IN)=( QDOT(IN)/AFLOW
  100   CONTINUE                                                C   >     -YZ(IN-1,3)*(HG-HL)*(XZ(IN)-XZ(IN-1))/DZ )/DENI ! DALFA/DT
        ECZO(0)=.5*((YZO(0,3)+YZ(0,3))**2)/RHOL - ECZO(0)       >      -(HG-HL)*(GX1-GX2)/DZ )/DENI    ! DALFA/DT
C                                                                     YZ(IN,3)=YZ(IN-1,3) + DZ*(RHOL-RHOG)*YDOT(IN)
C                                                               300   CONTINUE
C **    NOTE    DGDT=YDOT(NND+1)               ! APPROX 2ND ALF DER.=0  RETURN
C                                                                     END
C ------------------------------------------------------------
        DGDT=(-DPFO*(2.*YZO(0,3)*YZ(0,3)+YZ(0,3)**2)/YZO(0,3)**2
```

271

```
C
C
C
          SUBROUTINE DTGEN(NEQ,F,JAC,DT,TIME,Y,ATOL,RTOL
        >             ,MF,FILE,ICH,NCH)
C
C **   THIS VERSION OF DTGEN STORES THE AVERAGE CHANNEL VOID FRACTION,
C **        THE WHITE NOISE, THE INPUT FLOW AND PRESSURE,
C **        ALONG WITH THE POWER
C
C          ** TO BE USED ONLY WITH PROGRAM NNLFB.FTN **
C
C **   THIS SUBROUTINE SOLVES A SYSTEM OF EQUATIONS DEFINED
C **   IN SUBROUTINE F (WITH JACOBIAN JAC IF MF>20).
C
C **   PARAMETERS:
C **       NEQ    - NUMBER OF EQUATIONS  <10
C **       F      - NAME OF THE FUNCTION SUBROUTINE (DECLARED EXTERNAL)
C **       JAC    - NAME OF THE JACOBIAN SUB.          "     "
C **       DT     - DESIRED TIME INCREMENT FOR OUTPUT
C **                   LSODE COMPUTES ITS OWN TIME INCREMENT
C **                        TO LIMIT ERRORS
C **       TIME   - TOTAL TIME
C **       Y      - INITIAL VALUE
C **       ATOL   - ABSOLUTE TOLERANCE
C **       RTOL   - RELATIVE TOLERANCE
C **                   LSODE MAKES THE ERROR APPROX.
C **                        ERROR < RTOL*Y + ATOL
C **       MF     - 10  FOR NON STIFF PROBLEM
C **                - 21  FOR STIFF PROBLEM WITH FULL SUPLIED JACOBIAN
C **                - 23  FOR STIFF PROBLEMS.  LSODE COMPUTES A DIAG. JAC
C **       FILE   - FILE NAME TO STORE DATA IN MULSMN FORM
C **       ICH    - SIGNAL NUMBERS TO STORE IN DISK
C **       NCH    - NUMBER OF SIGNALS TO STORE
C
C
C ------------------------------------------------------------------
          PARAMETER MXND=12
          PARAMETER NEQMX=20
          IMPLICIT REAL*8 (A-H,O-Z)
          COMMON /MODEL/
        >           NND,A1,A2,ETA,DZ,XKF,PSHP
        >          ,XLA,DPFO,DPO,FLOW
        >          ,H,RHOL,RHOG,GC,XKC
        >          ,XKS,RCOEF,HL,HG,AFLOW
        >          ,WHITE,IMPUL,STEP,HSIN
          REAL *8 A1,A2,H,RHOL,RHOG,GC,XKF,XKC
        >          ,XLA,DPFO,DPO,FLOW
        >          ,XKS,RCOEF,ETA,HL,HG,PSHP(MXND),DZ
          LOGICAL *1 WHITE,IMPUL,STEP,HSIN
C ------------------------------------------------------------------
          COMMON /BOUND/ YZ,XZ,FHZ,ECZ,FR,QDOT
          REAL *8 YZ(0:MXND,3),XZ(0:MXND),FHZ(0:MXND),ECZ(0:MXND)
        >          ,FR(1:MXND),QDOT(1:MXND)
          COMMON /STEADY/ YZO,XZO,ECZO
          REAL *8 YZO(0:MXND,3),XZO(0:MXND),ECZO(0:MXND)
C ------------------------------------------------------------------
          REAL *8 Y(NEQ),ATOL(1),RWORK(256),RTOL,T,TOUT,Y1,Y2,YOLD(NEQMX)
```

```
C    >            ,YDOT(NEQMX)
          REAL *4 YDAT(256),YMAX(NEQMX),DT4,X1
          INTEGER *2 IWORK(30),IDAT(256),ICH(1)
          LOGICAL *1 FILE(28),LNCH,LICH,LB,AD,LO,IANS
          EXTERNAL F
          COMMON /PP/ PERT,TO,DTO,TOLD
          REAL *8 TO,DTO,TOLD
          COMMON /SEEDS/ I1,I2
          DATA I1/0/,I2/0/
          LOGICAL *1 ESC
          DATA ESC/27/,RWORK/256*0./
C -------------------------------------------------------------------
          COMMON /OUTP/ PRINT,PRALL,DSK
          LOGICAL *1 PRINT,PRALL,DSK
C -------------------------------------------------------------------
          COMMON /CNV/ FIRST,CONV,Y1,Y2,Y3,YEX1,YEX2,DR1,DR2,FN1,FN2
        >            ,IREC,KOUNT
          LOGICAL *1 END,FIRST,CONV
C -------------------------------------------------------------------
C
          DATA ITOL/1/,ITASK/1/,ISTATE/1/,IOPT/0/,LRW/256/,LIW/30/
C
          DT4=DT
          DTO=DT
          IF(NEQ.GT.NEQMX) STOP    ' TOO MANY EQUATIONS'
          IF(ATOL(1).EQ.0.) ATOL(1)=1.E-5
C
C
          ALFMX=0.
          KALFO=0
          KALF1=0
          FIRST=.TRUE.
          END=.FALSE.
          NCH=6
          IBKSZ=256
          NPB=256
          IF(NCH.NE.0)      NPB=IBKSZ/NCH        ! NUMBER OF DATA POINTS
          IF(NCH.NE.0)      IBKSZ=NCH*NPB        ! <IBKSZ
          NBK=IFIX(SNGL(TIME/(DT*FLOAT(NPB)))) + 0.9999
          IF(MF.LT.21.OR.MF.GT.25) MF=23         ! DEFAULT NON STIFF
C
C
          IF(DSK)
        > OPEN(UNIT=1,NAME=FILE,TYPE='NEW',ACCESS='DIRECT'
        >         ,RECORDSIZE=IBKSZ)
C
C **    START CALCULATIONS
C
          CALL CLREF(40)
          TOUT=0.
          DO 100 IB=1,NBK
          TOUT=FLOAT(IB-1)*FLOAT(NPB)*DT
          K=1
          DO 110 I=1,NPB
          TO=TOUT
          TOUT=TOUT+DT
C
C
```

```
      IF(WHITE) CALL BLGWN(I1,I2)      ' BAND LIMITED NOISE FOR USE IN FUNC  C
C                                                                            C    110   CONTINUE
      LKOUNT=0                                                               C
   91 CALL LSODE(F,NEQ,Y,T,TOUT,ITOL,RTOL,ATOL,ITASK,ISTATE,                 C **   WRITE RESULTS
     >           IOPT,RWORK,LRW,IWORK,LIW,JAC,MF)                            C
      IF(ISTATE.EQ.-1) GO TO 90                                                    IF(END) GO TO 777
      IF(ISTATE.NE.2) WRITE(5,1000) ISTATE                                         IF(DSK) WRITE(1'IB) (YDAT(I),I=1,IBKSZ)
 1000 FORMAT(' ERROR IN LSODE --  ISTATE =',I7)                                    NREC=IB
      IF(ISTATE.NE.2) STOP                                                         CALL READEF(40,IDS)               !SET EF 40 TO STOP
C                                                                                  IF(IDS.NE.0) GO TO 777
C                                                                            C
      IF(.NOT.WHITE .AND. .NOT.END) CALL CONVRG(Y(NND+4),T,FILE,END)         C
C                                                                            C    100   CONTINUE
C                                                                            C
      IF(PRALL) WRITE(5,1020) TOUT,(Y(J),J=1,NEQ)                            C
 1020 FORMAT(' ',1PG11.3,':',(T14,6G11.3/'+'))                               C    777   CLOSE (UNIT=1)
      PWR=Y(NND+4)*100      ! POWER IN %                                           IF(.NOT.END .AND. .NOT.WHITE) THEN
      IF(PRINT.AND. .NOT.PRALL) WRITE(5,1021) TOUT,PWR         '                     END=.TRUE.    ! DID NOT CONVERGE, BUT RUN OUT OF TIME
 1021 FORMAT(' ',1PG11.3,':',(T14,G15.5,'%'))                                       CALL CONVRG(Y(NND+4),T,FILE,END)
D     IF(PRALL)      WRITE(6,3020)                                                  ENDIF
D    >      ((YZ(L,J),J=1,3),XZ(L),FHZ(L),ECZ(L),L=0,NND)                           IF(KALFO.NE.0 .OR. KALF1.NE.0) WRITE(6,7020) KALFO,KALF1,ALFMX
D3020 FORMAT(' YZ,XZ,FHZ,ECZ'/4(3G14.3,'/',3G14.3/))                         7020  FORMAT(//60('*')/60('-')/' ALFA WAS < 0 ',I5, 'TIMES'/
D     IF(PRALL)      WRITE(6,3030) (FR(L),L=1,NND)                                 >           ' ALFA WAS > 1 ',I5, 'TIMES'/
D3030 FORMAT(' FR',4G20.5)                                                         >           ' MAXIMUM ALFA = ',G15.4/
C                                                                                  >           60('-')/60('*'))
      YDAT(K)=PERT              ! WHITE NOISE PERT                                 IF(.NOT.DSK) RETURN
      K=K+1                                                                 C
      SUM=0.                                                                 C **   CREATE REDUCED ID FILE
      DO 115 J=1,NND                                                         C
      SUM=SUM+Y(J)                                                                 NFL=0
      ALF=YZO(J,1)+YZ(J,1)                                                         NFLMX=10
      IF(ALF.GT.ALFMX) ALFMX=ALF                                              70   NFL=NFL+1
      Y(J)=ALF              ! TEMPORARY STORAGE                                    IF(FILE(NFL).EQ.':') NFLMX=NFLMX+NFL    ! FILE NOT IN SY:
      IF(ALF.GE.1. .AND. KALF1.EQ.0) WRITE(6,7000)                                IF(NFL.LT.NFLMX.AND.
      IF(ALF.GE.1.) KALF1=KALF1+1                                                 >      (FILE(NFL).NE.0.AND.FILE(NFL).NE.'.')) GO TO 70
 7000 FORMAT(//60('*')/60('-')/                                                    FILE(NFL)='.'
     >    ' ALFA IS GREATER THAN 1.'/' STANDARD CORRECTION TAKEN'                  FILE(NFL+1)='I'
     >    ' -- NO MORE MESSAGES WILL BE PRINTED'/60('-')/60('*'))                  FILE(NFL+2)='D'
      IF(ALF.LE.0. .AND. KALFO.EQ.0) WRITE(6,7010)                                 FILE(NFL+3)=0
      IF(ALF.LE.0.) KALFO=KALFO+1                                                  OPEN (UNIT=1,NAME=FILE,ACCESS='DIRECT',TYPE='NEW',
 7010 FORMAT(//60('*')/60('-')/                                              1     MAXREC=1,INITIALSIZE=2,RECORDSIZE=165)
     >    ' ALFA IS GREATER THAN 1.'/' STANDARD CORRECTION TAKEN'            C     DO 720 I=1,NCH
     >    ' -- NO MORE MESSAGES WILL BE PRINTED'/60('-')/60('*'))            C 720   YMAX(I)=1./YMAX(I)
  115 CONTINUE                                                                     DO 720 I=1,NCH
C                                                                            720   YMAX(I)=1.
      IF(PRALL) WRITE(5,7015) (Y(J),J=1,NND)                                       LNCH=NCH
 7015 FORMAT(' ALFA  :: ',2(T14,6G11.3/'+'))                                       LO=0
C                                                                                  LB=' '
      YDAT(K)=SUM/FLOAT(NND)  ' AVERAGE VOID                                       AD='N'
      K=K+1                                                                        X1=1.
      YDAT(K)=Y(NND+4)        ' POWER                                              D1='PERTURB.'
      K=K+1                                                                        D2='AV. ALFA'
      YDAT(K)=Y(NND+1)        ' FLOW (Kg/sm2)                                      D3='POWER   '
      K=K+1                                                                        D4='FLOW    '
      YDAT(K)=YZ(0,2)         ' PRESSURE (Nw/m2)                                   D5='PRESSURE'
      K=K+1                                                                        D6='F. TEMP.'
      YDAT(K)=Y(NND+2)        ' FUEL TEMPERATURE (K)                               D0='        '
      K=K+1                                                                        WRITE(1'1)IBKSZ,NREC,LNCH,LO,DT4,X1,LO,LO,LO,AD,
```

273

```
C     > (YMAX(I),I=1,NCH),(X1,I=1,(48-NCH)),(LB,I=1,446)
      > (YMAX(I),I=1,NCH),(X1,I=1,(48-NCH))
      > ,D1,DO,DO,D2,DO,DO,D3,DO,DO,D4,DO,DO,D5,DO,DO,D6,DO,DO
      > ,(LB,I=1,302)
C
      CLOSE (UNIT=1)
      RETURN
C
  90  WRITE(5,3000)
3000  FORMAT(' LSODE MADE MORE THAN 500 ITERATIONS'/
      >        ' CALCULATIONS RESUME')
      LKOUNT=LKOUNT+1
      IF(LKOUNT.GT.10) GO TO 777
      ISTATE=2
      GO TO 91
      END
```

```
C
C
C
      SUBROUTINE BLGWN(I1,I2)
C
C **   FUNCTION BLGWN          BAND LIMITED GAWSIAN WHITE NOISE
C
C **   WITH THIS PARAMETERS THE FILTER IS SET AT 80% OF NYQUIST FREQ.
C **   WHEN SAMPLING ONE EVERY TEN POINTS
C **   THIS NOISE HAS A STANDARD DEVIATION OF 1.0
C
      IMPLICIT REAL*8 (A-H,O-Z)
      REAL *8 G,H(8)
      DATA G/0.0000081941793343/,
     > H/6.7106791240246675,-19.0375452120500596,
     > 33.6094741679254439,-35.7270320917172163,
     > 24.3960891362192910,-10.4474756767113902,
     >  2.5649604195750299, -0.2763572614363015/
     > NPOLES/8/,STDNRM/0.20335/
C
C **   ARRAY Y CONTAINS THE BLGWN IN POSITION 11.
C **   IT ALSO CONTAINS 10 POINTS BEFORE AND 10 POINTS AFTER
C **   IN INCREMENTS OF DT/10.
C
      REAL *8 Y(21)
      COMMON /NOISE/ Y,FIRST
      DATA Y/21*0./
C
      LOGICAL *1 FIRST
      DATA FIRST/.TRUE./
C
C
      K=0
  10  DO 200 IB=1,10            ' CALCULATE NEW 10 POINTS (DT/10.)
      X=GWN(I1,I2)
      DO 100 I=1,20
 100  Y(I)=Y(I+1)
C
      Y(21)=G*X/STDNRM
      DO 110 I=1,NPOLES
 110  Y(21)=Y(21)+H(I)*Y(21-I)
 200  CONTINUE
      WN=SNGL(Y(11))   ! OUTPUT NEW BLGWN (REAL DT) [ NOT RETURNED ]
C
      IF(.NOT.FIRST) RETURN
      K=K+1
      IF(K.LT.25) GO TO 10               ' INITIALIZE FILTER
      FIRST=.FALSE.
      RETURN
      END
```

274

```
C                                                                           IF(ABS(D2).GT.DROUND .OR. ABS(D1).GT.DROUND) THEN
C                                                                               GM=D1/D2
C                                                                                 IF(GM.NE.-1.) DR=GM/(1.+GM)
        SUBROUTINE CONVRG(YY,TT,FILE,END)                                         IF(GM.EQ.-1.) DR=-1.
C                                                                               FN=1./(TEX3-TEX1)
C **    THIS SUBROUTINE FINDS THE MAXIMA AND MINIMA                          ELSE
C **    OF A STEP RESPONSE GENERATED BY TLAF                                     KOUNT=KOUNT+1
C **    AND CALCULATES D.R. AND WHEN IS THE RESPONSE CONVERGED               ENDIF
C                                                                        ENDIF
        REAL *8 DRSUM,DRS2,FNSUM,FNS2,SMAX,SMAX2,SMIN,SMIN2              YEX1=YEX2
        REAL *8 YY,TT                                                   TEX1=TEX2
        LOGICAL *1 FILE(28)                                            YEX2=YEX3
        LOGICAL *1 END,FIRST,CONV                                       TEX2=TEX3
        COMMON /CNV/ FIRST,CONV,Y1,Y2,Y3,YEX1,YEX2,DR1,DR2,FN1,FN2   C
     >         ,IREC,KOUNT
        COMMON /OUTP/ PRINT,PRALL,DSK                               IF(CONV .AND. YEX1.GT.YEX2) THEN
        LOGICAL *1 PRINT,PRALL,DSK                                      YMX=YEX1
C                                                                       YMN=YEX2
        Y=SNGL(YY)                                                      SMAX=SMAX+YMX
        T=SNGL(TT)                                                      SMAX2=SMAX2+YMX**2
C                                                                       SMIN=SMIN+YMN
        IF(END) GO TO 10                                               SMIN2=SMIN2+YMN**2
        IF(.NOT.FIRST) GO TO 1                                          NN=NN+1
        FIRST=.FALSE.                                               ENDIF
        FILE(9)='.'                                             C
        FILE(10)='E'                                               IF(DR.NE.-999.) THEN
        FILE(11)='X'                                                  IF(.NOT.CONV) THEN
        FILE(12)='T'                                                    DRERR=ABS(DR1-DR)+ABS(DR2-DR)
        FILE(13)=0                                                      IF(DRERR.LT.DR*.01) THEN
        CLOSE(UNIT=3)                                                     CONV=.TRUE.
        OPEN(UNIT=3,NAME=FILE,TYPE='NEW',ACCESS='DIRECT'                 DRSUM=DR1+DR2+DR
     >         ,RECORDSIZE=2)                                             DRS2=DR1**2+DR2**2+DR**2
        IREC=1                                                           FNSUM=FN1+FN2+FN
C                                                                       FNS2=FN1**2+FN2**2+FN**2
        Y1=0                                                             NSUM=3
        Y2=0                                                             DRAV=DRSUM/FLOAT(NSUM)
        Y3=0                                                             DRSD=DSQRT(DABS(DRS2/FLOAT(NSUM)-DRAV**2))
        YEX1=-999.                                                        IF( ABS(DRAV-1.) .GT. 0.002 ) CONV=.FALSE. ' NOT LIMIT CYCLE
        YEX2=-999.                                                       YMX=AMAX1(YEX1,YEX2)
        KOUNT=0                                                          YMN=AMIN1(YEX1,YEX2)
        DR1=-999.                                                        SMAX=YMX
        DR2=-999.                                                        SMAX2=YMX**2
        CONV=.FALSE.                                                     SMIN=YMN
        END=.FALSE.                                                      SMIN2=YMN**2
C                                                                       NN=1
C                                                                       ELSE
    1   IF(KOUNT.GT.10) GO TO 10          ' CONVERGED                     DR1=DR2
        Y3=Y                                                             DR2=DR
        T3=T                                                             FN1=FN2
        IF((Y2.GT.Y1.AND.Y2.GT.Y3).OR.(Y2.LT.Y1.AND.Y2.LT.Y3) ) THEN ' EXTREMA    FN2=FN
            YEX3=Y2                                                      ENDIF
            TEX3=T2                                                    ELSE
            DR=-999.     ' FORMAT CHECK                                  DRAV=DRSUM/FLOAT(NSUM)
             IF(YEX1.NE.-999.) THEN                                      DRSD=DSQRT(DABS(DRS2/FLOAT(NSUM)-DRAV**2))
                D1=YEX3-YEX2                                             DRERR=ABS(DR-DRAV)
                D2=YEX1-YEX3                                             IF (DRERR .LT. 5.*DRSD) THEN
                DROUND=ABS(Y2*5.E-2)              ' ROUNDOFF CUTOFF         DRSUM=DRSUM+DR
                IF(DROUND.LT.2.E-3) DROUND=2.E-3                         DRS2=DRS2+DR**2
                                                                        FNSUM=FNSUM+FN
```

275

```
                    FNS2=FNS2+FN**2
                    NSUM=NSUM+1
                  ELSE
                    KOUNT=KOUNT+1
                  ENDIF
                ENDIF
              ENDIF
C
          IF(PRINT) WRITE(5,9000) DR,FN
 9000     FORMAT(' ',35('-'),2G15.4)
          WRITE(3'IREC) T2,Y2
          IREC=IREC+1
          IF(CONV .AND. ABS(DRAV-1.).LT.3.*DRSD) KOUNT=KOUNT+1 ! LIMIT CYCLE
C
        ENDIF
C
        Y1=Y2
        T1=T2
        Y2=Y3
        T2=T3
C
C
        RETURN
C
C **    CONVERGED
C
   10   END=.TRUE.
        IF(CONV) THEN
          DRAV=DRSUM/FLOAT(NSUM)
          DRSD=DSQRT(DABS(DRS2/FLOAT(NSUM)-DRAV**2))
          FNAV=FNSUM/FLOAT(NSUM)
          FNSD=DSQRT(DABS(FNS2/FLOAT(NSUM)-FNAV**2))
C
          WRITE(6,5000) DRAV,DRSD,FNAV,FNSD
 5000     FORMAT(////' AVERAGE DECAY RATIO = ',G18.5,' +/- ',G18.5
     >              /' AVERAGE FREQUENCY   = ',G18.5,' +/- ',G18.5)
          WRITE(6,5005) NSUM
 5005     FORMAT(' ',I6,' OSCILLATIONS USED FOR AVERAGE'/)
        ELSE
          WRITE(6,5010)
 5010     FORMAT(////' DECAY RATIO DID NOT CONVERGED ')
        ENDIF
C
C
        IF(ABS(DRAV-1.) .LT. DRSD*5.) THEN
          AVMX=SMAX/FLOAT(NN)
          SDMX=DSQRT(DABS(SMAX2/FLOAT(NN)-AVMX**2))
          AVMN=SMIN/FLOAT(NN)
          SDMN=DSQRT(DABS(SMIN2/FLOAT(NN)-AVMN**2))
          WRITE(6,5020) AVMX,SDMX,AVMN,SDMN
 5020     FORMAT(/' LIMIT CYCLE :'/
     >              ,' MAXIMUM OSCILLATION = ',1PG18.5,' +/- ',G18.5/
     >              ,' MINIMUM OSCILLATION = ',1PG18.5,' +/- ',G18.5///)
        ELSE
          AVMX=-999.
          AVMN=-999.
          SDMX=-999.
          SDMN=-999.
```

```
          WRITE(6,5030)
 5030     FORMAT(/' LIMIT CYCLE WAS NOT REACHED ')
        ENDIF
C
C
        XJNK=-999.
        WRITE(3'IREC) (XJNK,I=1,2)
        IREC=IREC+1
        WRITE(3'IREC) DRAV,DRSD
        IREC=IREC+1
        WRITE(3'IREC) FNAV,FNSD
        IREC=IREC+1
        WRITE(3'IREC) AVMX,SDMX
        IREC=IREC+1
        WRITE(3'IREC) AVMN,SDMN
        IREC=IREC+1
        CLOSE(UNIT=3)
        RETURN
C
        END
```

276