1992

Steady-state oscillations of linear and nonlinear systems

Christopher A. Tucher

STEADY-STATE OSCILLATIONS

OF LINEAR AND NONLINEAR SYSTEMS

by

Christopher A. Tucker

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in
Mechanical Engineering

Approved by:

Professor J. S. Torok - Thesis Advisor

Professor R. B. Hetnarski

Professor A. B. Engel (Mathematics Department)

Professor C. W. Haines (Department Head)

DEPARTMENT OF MECHANICAL ENGINEERING
COLLEGE OF ENGINEERING
ROCHESTER INSTITUTE OF TECHNOLOGY
ROCHESTER, NEW YORK

MAY 1992

I, Christopher Anthony Tucker, hereby grant permission to Wallace Memorial Library of Rochester Institute of Technology to reproduce my thesis in whole or in part. Any reproduction will not be for commercial use or profit.

May 12, 1992

ACKNOWLEDGEMENTS

I would like to sincerely thank Dr. Joseph Torok, my thesis advisor, for encouragement, guidance, valuable suggestions, and for always being available for assistance. His dedication towards his students' work have been unmatched throughout my years at RIT. He is a great mathematician, professor and friend.

I am deeply grateful to Dr. Charles Haines, Dr. Richard Hetnarski and Dr. Alejandro Engel for their expertise and helpful suggestions.

Last but not least, I dearly thank my wife Cynthia, for added motivation and for being so patient and understanding at times when I spent many continuous hours working on this project. I know she had her hands full with our daughter Kristen. I dedicate this thesis to Cynthia and Kristen, my joys.

ABSTRACT

In this paper, an efficient algorithm is developed for the identification of stable steady-state solutions to periodically forced linear and nonlinear dynamical systems. The developed method is based on mapping techniques introduced by Henri Poincare' and the theory of one-parameter transformation groups. The algorithm successfully identifies initial conditions which give rise to strictly periodic orbits. The technique is demonstrated on selected problems associated with linear as well as nonlinear systems.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF FIGURES (continued)

LIST OF SYMBOLS

A,B,C ....Constant

A_t .... Time dependent operator

[A], [B], [K]....Square matrix

[I] ....Identity square matrix

c ....Viscous damping constant

F ....Force function

$\vec{F}$ ....Force vector

f ....Periodic input function

$\vec{f}$ ....Periodic forcing vector

G ....Forward advance transformation

G′ .... Derivative of **G** for 1-D system

D**G**..............Partial derivative of **G**

J ....Jacobian operator

J**G** ....Jacobian of **G** function

J$\mathbb{P}$ ....Jacobian of $\mathbb{P}$ function

$\mathbb{H}$ ....System chracteristic Transfer function

x,y ....Rectangular coordinates, distances

$\dot{x}$,$\dot{y}$ ....Time derivative of coordinates, x,y

$\ddot{x}$ ....Time derivative of $\dot{x}$

x_0,y_0 .... Initial conditions for x,y

x^* .... Initial value that gives periodic solution,x_0

$\vec{x}$,$\vec{y}$ ....Position vector

$\vec{g}(x)$,$\vec{y}(x)$ .....Vector functions

x^h,y^h .... Series solution expansion

k ....Spring stiffness constant

m ....Mass

$\mathbb{M}$ .... Manifold, n-dimensional

N ....Number of time interval

$\mathbb{P}$ ....Poincare map

$\mathbb{P}'$ .... Derivative of Poincare Map for 1-D system

D$\mathbb{P}$ ....Partial derivative of $\mathbb{P}$

LIST OF SYMBOLS (continued)

$\mathbb{R}$ ....Real space

T ....Period

t ....Time

t_0 .... Initial condition of t

τ ....Period: Periodic time

Φ ....Forward advance transformation function

Φ_0 .... Trajectory at time t_0

Φ_t .... Trajectory at time "t"

Φ_T ....Trajectory at time t = T (period)

Φ_{nT} ....Trajectory at time t = nT

ω ....Circular frequency of forced vibration

ω_n ....Natural frequency

Ω ....Driving frequency of system

ϵ ....Small element, accuracy error parameter

β ....System parameter constant

ζ ....Viscous damping factor

U ....Infinitesimal Generator operator

v ....Velocity

$\dot{v}$ ....Acceleration

x^h, y^h .... Series solution

$\{x(t)\}$ ....Displacement vector

$\{y(t)\}$ ....State vector

$\{F(t)\}$ ....Force vector

$\{\dot{x}(t)\}$ ....Velocity vector

$\vec{u}(t)$ ....Forcing vector

$\frac{\partial}{\partial x}$ ....Partial derivetive operator w.r.t x

$\prod$ .... Mapping

I INTRODUCTION

Oscillatory motion is an important aspect in the fields of physics and engineering. Periodic motion is common in most physical systems. Some examples include the motion of planets, the earth around the sun, the moon around the earth, the movement of bodies of water (ocean waves), all repeating their motion after a specified time. The analysis of oscillations is an important part of mechanical vibration, and is an essential design criterion that is necessary in almost all structural and mechanical systems in present day engineering design.

Any attempt to design a mechanical system usually begins with a prediction of its performance. Linear vibration analysis has been adequate for most applications. However, because of the current high demand for greater system performance, the application of linear analysis sometimes results in failures. Many of these failures are a result of nonlinear effects in systems that were designed under the assumption of linear behavior. Nonlinear analysis now receives considerable attention in an effort to understand phenomena not predicted by traditional linear analysis.

Physical systems are modeled by differential equations. Based on the nature of the differential equation, the system can be classified as linear or nonlinear. There are many characteristics which distinguish between the solutions of linear and nonlinear differential equations. For example, the fundamental system of solutions exists only for linear differential equations [1]. This implies that if certain basic solutions are known, the general solution will be a linear combination of these fundamental solutions. However, it is more often than not impossible to analytically solve nonlinear differential equations. Consequently, because of the difficulty involved, approximation methods and qualitative analyses of the solutions become important in studying the nature of nonlinear oscillations [2].

Linear analysis is a rather mature subject. It is a unified theory based on concepts and results from linear algebra and its generalization, functional analysis. The principle of superposition allows linear differential equations to be solved analytically. All solutions can be constructed from the fundamental solutions which are exponential functions [2,14]. This limits the type of behavior encountered in linear systems. Its utility in solving a vast multitude of physical problems, however, remains unsurpassed.

The analysis of nonlinear systems is a richer topic in comparison to the standard linear theory. The lack of a unified theory that would encompass nonlinear analysis allows for considerable variation in system properties and qualitative behavior. Not only do nonlinear systems behave differently from linear ones, the system response may at first seem unintuitive. Limit cycles, for example, are unique to some dissipative nonlinear systems. Limit cycles are isolated periodic solutions which attract a dense subset of the state space [6]. Other phenomena include amplitude instabilities, catastrophes and chaotic behavior [1,6].

The focus of this investigation is the steady state behavior of linear and nonlinear systems. Steady state behavior is understood to be the long-term response of a system due to external forcing. The attention will be restricted to periodic behavior, which occurs universally in linear as well as nonlinear systems. In particular, a general method for the determination of period solutions will be proposed and examined. The proposed method is constructive, that is, it yields actual results. In addition, the idea is applicable to linear and nonlinear systems without modification.

Some attention has been focused on the determination of steady state solutions [3-6]. Typical methods of analysis for steady state periodic response (for a given initial state x_0 at time t_0) entail integrating the governing matrix equations until the response becomes periodic. This means that the transient response becomes negligible. For lightly-damped systems the analysis is exceedingly slow, and could be prohibitive, as it must extend over far too many periods. Also, it is hard to tell whether a stable orbit exists and what its period T is, or whether or not the response will end up at a singular point. This method is called the brute-force approach.

Aprille and Trick [3] developed a series of algorithms for the determination of periodic solutions associated with problems in nonlinear circuit analysis. Their proposed method was apparently successful, but the outlined procedure is cumbersome. It requires integration of the system equations together with the coupled variational equations. The variational equations constitute a linearization of the system about a specified solution. Hence, n+1 analyses are required for each iterate where n is the dimension of the state space.

A more systematic approach has been developed, one that rapidly determines initial conditions which give rise to strictly periodic solutions. The methodology has been automated by the use of a symbolic computation program, MAPLE. This generalized approach is briefly outlined below.

The integration of the system equations up to a fixed time defines a family of point transformations, parameterized by the time variable, mapping the state space into itself. The method of solution requires the use of Lie group theory. This allows the construction of the global transformation equations with the characteristic infinitesimal generator of the group [11]. The solutions generated by such a Lie series representation constitute a generalization of solutions obtained for linear, constant coefficient systems [14]. Recall that the fundamental matrix solution is expressible as a series expansion of a matrix-valued exponential function. The primary motivation for developing Lie series solutions to differential equations is that complete solutions to the problem are generated for arbitrary initial conditions. With the availability of computational and symbolic mathematics programs [9], the series solution of differential equations are much more feasible now [11].

BACKGROUND

There are two kinds of dynamical systems that are encountered in vibration analysis, autonomous systems and forced systems which are called nonautonomous. For a nonautonomous system, the independent variable t (time) is present in the forcing function of the system differential equation. The function depends explicitly on t. The first order system given below is nonautonomus:

$$\dot{x} = f(x,t) \tag{1.1}$$

On the other hand, autonomous system differential equations have no explicit dependence on t. The differential equation in (1.2) is an example of an autonomous first order system:

$$\begin{aligned} \dot{x} &= f(x,y) \\ \dot{y} &= g(x,y) \end{aligned} \tag{1.2}$$

In this investigation we will propose a general method to find periodic solutions of nonautonomous system equations. Although the methodology is completely general, the discussion will focus separately on first-order systems, higher-order systems, linear and nonlinear systems.

Nonautonomous Systems

Consider the nonautonomous first-order equation

$$\dot{x} = f(x,t) \tag{1.3}$$

where f is periodic in t of period T, and is continuous in t and x. The analysis of periodic solutions is a nontrivial problem. It is essentially a two-point boundary value problem in which the solution to (1.3) on the interval [0,T] must satisfy the boundary condition

$$x(0) = x(T) \tag{1.4}$$

This type of problem can be solved using the shooting method for boundary value problems. But this technique would be cumbersome at best. Integrating both sides of equation (1.3)

$$x(T) = \int_0^T f(x,\tau)d\tau + x(0) \tag{1.5}$$

We can express the above problem in terms of a mapping

$$x_0 = \varphi(x_0) \tag{1.6}$$

where

$$x_0 = x(0) \tag{1.6a}$$

$$\varphi(x_0) = \int_0^T f(x, \tau)\, d\tau + x_0 \tag{1.6b}$$

and x(t) satisfies equation (1.3) for $0 \le t \le T$.

One approach to finding the periodic solution of equation

(1.6) is by means of the Newton Raphson iteration

$$x_0^{i+1} = x_0^i - [I - d\varphi(x_0^i)]^{-1}[x_0^i - \varphi(x_0^i)] \qquad (1.7)$$

where

$$d\varphi(x_0^i) = \left.\frac{\partial x(T;x_0)}{\partial x_0}\right|_{x_0^i} \qquad (1.8)$$

We will review the concept of a Poincare' Mapping and in the process show how to use it for finding periodic solutions of nonautonomous systems.

Poincare' Mapping for Nonautonomous Systems

Consider the nonautonomous system

$$\dot{\vec{x}} = f(\vec{x},t) \qquad \vec{x} \in \mathbb{R}^n, \quad f \in C^1(\mathbb{R}\times\mathbb{R}^n \to \mathbb{R}^n) \qquad (1.9)$$

With a simple association of variables $\theta \equiv t$, we convert to the autonomous system

$$\left.\begin{array}{l}\dot{\vec{x}} = f(\vec{x},\theta) \\ \dot{\theta} = 1\end{array}\right| \quad \text{vector field on } \mathbb{R}^{n+1} \qquad (1.10)$$

Hence any general results for autonomous systems on $\mathbb{R}^m$, $m \geq 2$ will hold for nonautonomous systems as well.

Of particular (and practical) interest is when $f(\vec{x},t)$ is periodic in t. That is,

$$f(\vec{x},t) = f(\vec{x},t + T) \tag{1.11}$$

for each fixed $\vec{x}$. In any case, the phase space for the system in equation (1.9) is n+1 dimensional:

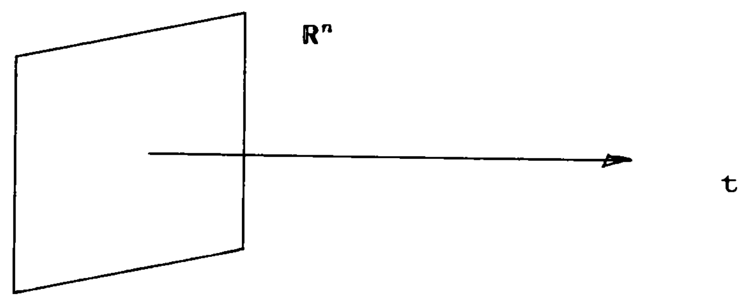


FIG 1-1. Phase space

The phase curves (sometimes called integral curves for nonautonomous systems) are smooth curves in $\mathbb{R}^{n+1}$:

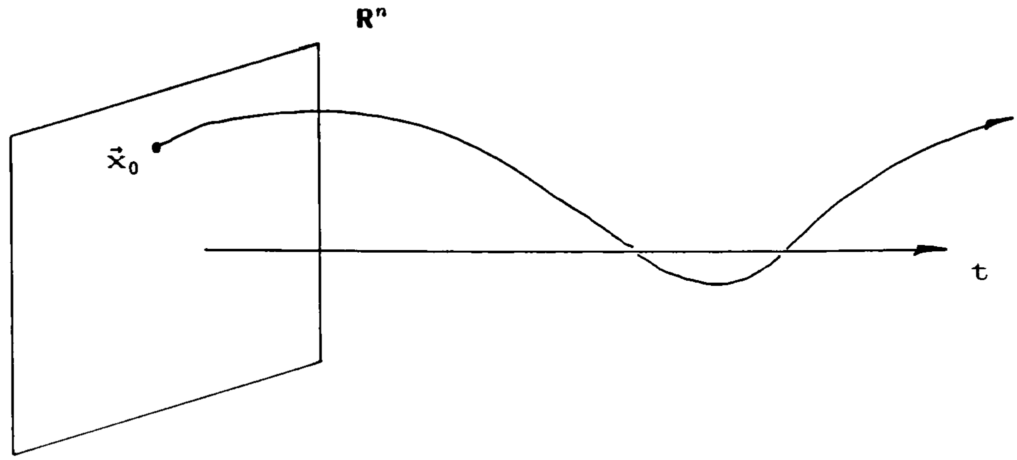


FIG 1-2. Phase curve

The projection of the phase space onto $\mathbb{R}^n$ constitutes the reduced phase space or state space of system (1.10).

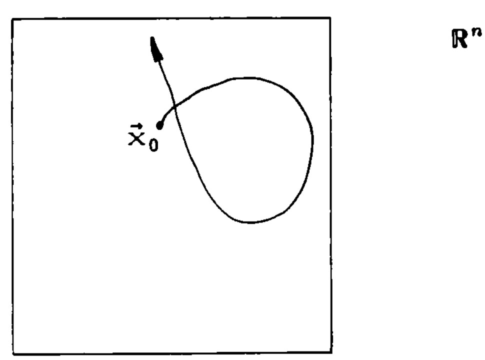


FIG 1-3. Reduced phase space/state space

In the reduced phase space, trajectories can intersect. This is a typical feature of nonautonomous systems. We can still define a solution:

$$\varphi_t(t_0,\vec{x}_0) \quad \in \quad \mathbb{R}^n$$

i.e.

$$\frac{d}{dt}[\varphi_t(t_0,\vec{x}_0)] = f(t,\varphi_t(t_0,\vec{x}_0)) \qquad (1.12)$$

with $\varphi_{t_0}(t_0,\vec{x}_0) = \vec{x}_0$

The graph of this solution $\left(\varphi_t(t_0,\vec{x}_0),t\right)$ defines a phase flow on $\mathbb{R}^{n+1}$

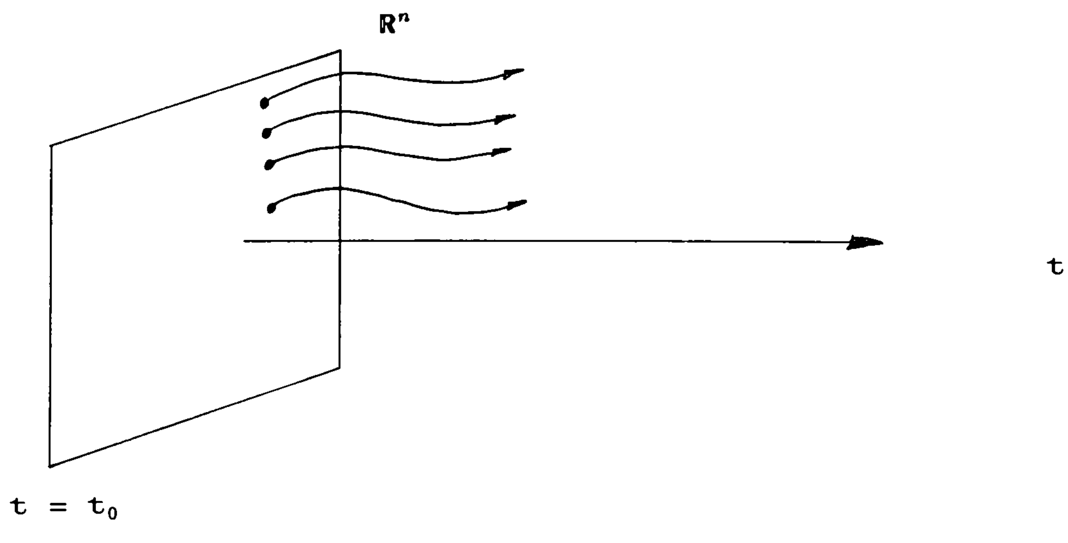


FIG 1-4. Phase flow

For example, given the simple first order system

$$\dot{x} = f(x,t), \quad x \in \mathbf{R}$$

the state space consists of the real line. The integral curves define a phase flow on $\mathbf{R} \times \mathbf{R}$

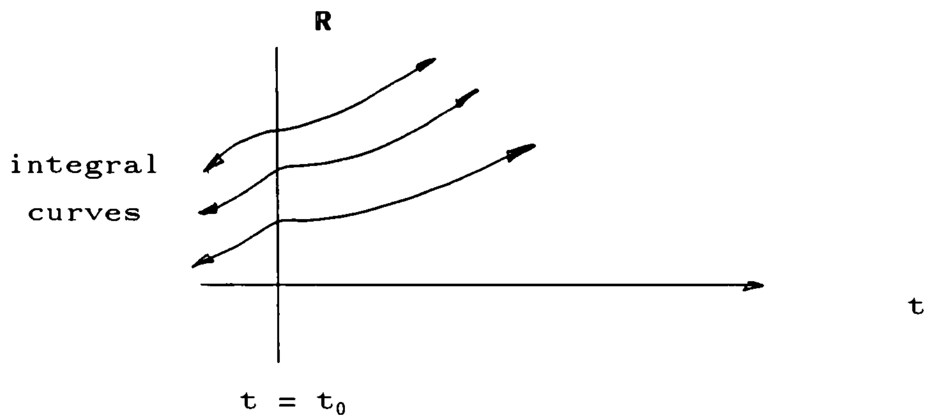


FIG 1-5. Integral curves defining a phase flow

<u>Poincare' Mapping</u>

Let $\mathbf{M}$ be an n-dimensional manifold in $\mathbf{R}^{n+1}$. If the t-axis is a transversal to $\mathbf{M}$, then (for the purposes of dynamics) $\mathbf{M}$ is a <u>Poincare "Surface" of Section</u>.

Given any (fixed) $t = t_0$, the flow $\left(\varphi_t(t_0,\vec{x}_0),t\right)$ defines a mapping (locally, near the t-axis)

$$\prod: \mathbf{R}^n \times \{t_0\} \rightarrow \mathbf{M}$$

The mapping is defined by integrating initial conditions forward until the integral curves intersect the manifold $\mathbf{M}$.

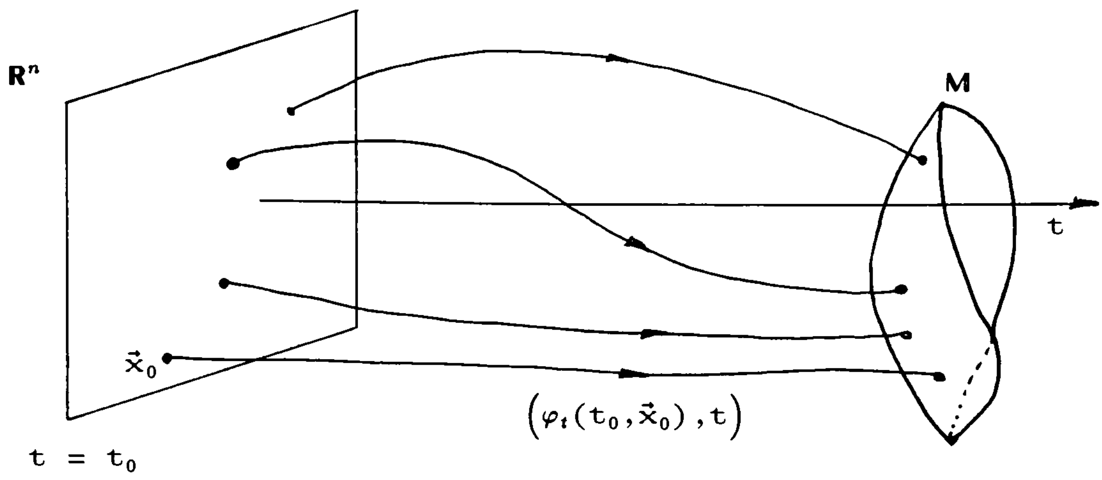


FIG 1-6. Integration of integral curves from $\mathbb{R}^n \to \mathbb{M}$

For practical purposes, $\mathbb{M}$ is taken as an isomorphic copy of $\mathbb{R}^n$:

$$\mathbb{M} \equiv \mathbb{R}^n \times \{t_1\} \qquad \text{for some } t = t_1$$

Hence the mapping is defined by integrating initial conditions forward until the trajectories intersect the hyperplane $\mathbb{M} = \mathbb{R}^n \times \{t_1\}$. Specifically,

$$\prod(\vec{x}_0) = \left(\varphi_{t_1}(t_0,\vec{x}),t_1\right) \tag{1.13}$$

Poincare' Section

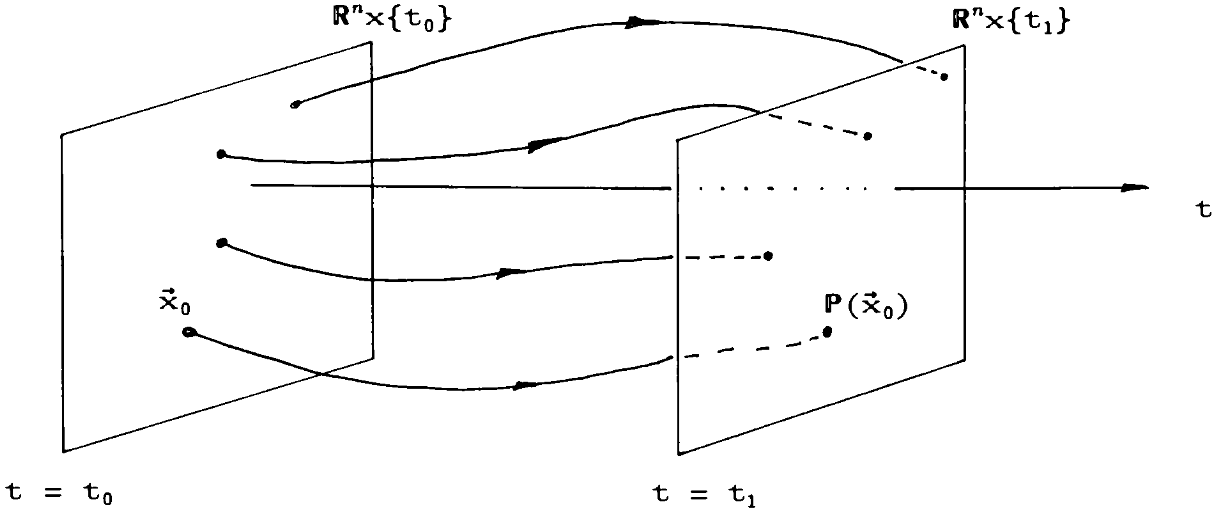


FIG 1-7. Projection operators showing mapping on $\mathbb{R}^n$

Using projection operators, the continuous dynamics are reduced to the action of a mapping on $\mathbb{R}^n$:

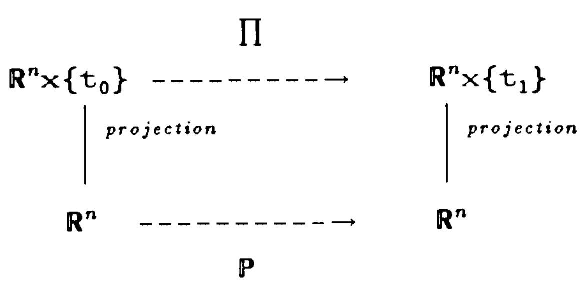


FIG 1-8. Mapping of $\mathbb{P}$ in the state space

This diagram defines a mapping $\mathbb{P}$ on the state space $\mathbb{R}^n$.

$\mathbf{P}: \mathbf{R}^n \rightarrow \mathbf{R}^n$ can also be considered as a forward-advance mapping.

Letting now the initial condition be arbitrary (dropping the subscript)

$$\mathbf{P}(\vec{x}) = \varphi_{t_1}(t_0,\vec{x}) \tag{1.14}$$

[note that the "initial" time and final have been fixed]

The Poincare' Mapping (corresponding to a Euclidean surface of section) reduces the investigation of the dynamics to the analysis of n-dimensional maps. The following observations can be made:

- The behavior of the flow is preserved by the mapping. That is, convergence or divergence of trajectories can be investigated.
- The "dimension" of the problem is effectively reduced by one.

The Poincare' Mapping is most useful in studying periodic solutions, limit sets, and asymptotic behavior. This paper will focus on its use in the investigation of periodic solutions.

Periodic Solutions

Consider $\dot{\vec{x}} = f(t,\vec{x})$ (on $\mathbf{R}^n$)

with $f(t + T,\vec{x}) = f(t,\vec{x})$ for each $\vec{x} \in \mathbf{R}^n$

We can convert to an equivalent autonomous system (using $\theta = \omega t$)

$$\left.\begin{aligned} \dot{\vec{x}} &= (f(\tfrac{\theta}{\omega},\vec{x}) \\ \dot{\theta} &= \omega \end{aligned}\right| \qquad \text{where } \omega = \frac{2\pi}{T}$$

Now $f(\frac{\theta}{\omega},\vec{x})$ is 2π-periodic in θ.

To investigate the periodic solutions, the integral curves are "tracked" at multiples of the forcing period

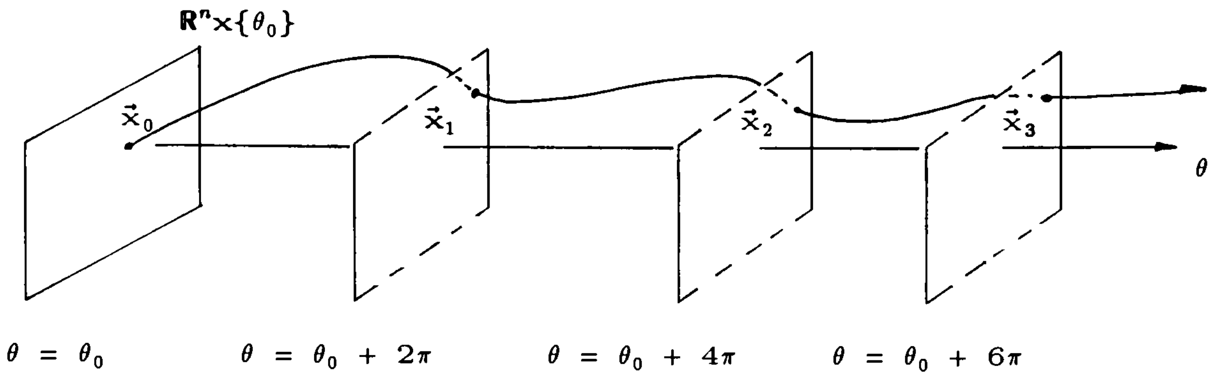


FIG 1-9. Integral curve motion at different values of θ

But since the forcing function $f(\frac{\theta}{\omega},\vec{x})$ is 2π-periodic, we need to concentrate only on the single forcing period $\theta_0 < \theta < \theta_0 + 2\pi$ and keep track of the images:

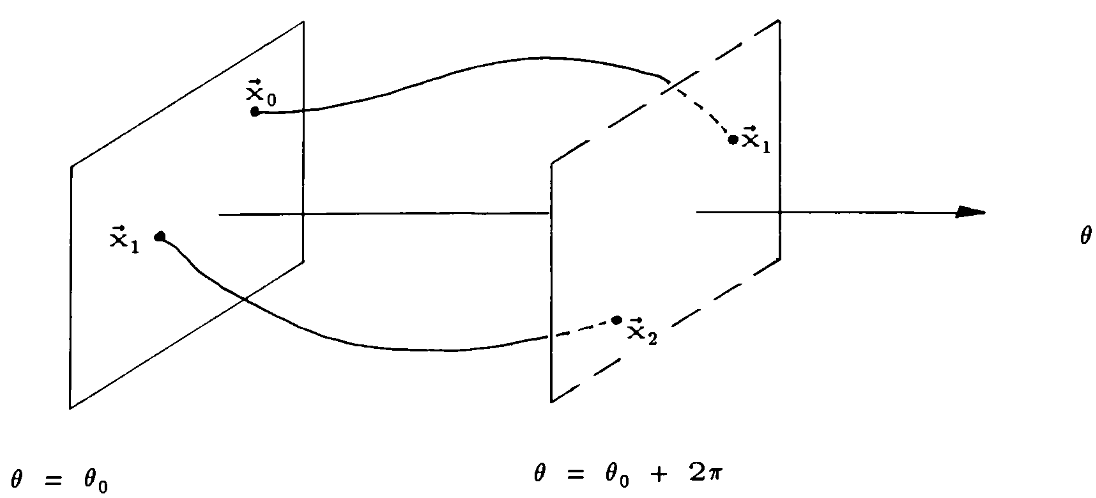


FIG 1-10. Image of a single forcing period

That is we re-start the dynamics with a "new" initial condition each time. The integral curves (solutions) are effectively tracked by deducing the Poincare' Mapping

$$\mathbf{P}: \mathbf{R}^n \rightarrow \mathbf{R}^n$$

[Each point is integrated forward over the interval θ to $\theta_0+2\pi$]

<u>Theorem</u>: Given $\dot{\vec{x}} = f(t,\vec{x})$ in $\mathbf{R}^n$, $f(t,\vec{x})$ T periodic. The system has a T-periodic solution if the associated Poincare' Mapping, $\mathbf{P}$, has a fixed point $\vec{x}_p$.

Proof: Clearly, a T-periodic solution results in a fixed point of the Poincare' Mapping. Suppose $\mathbf{P}(\vec{x}_p) = \vec{x}_p$ for some $\vec{x}_p \in \mathbf{R}^n$. This means that

$$\varphi_{T+t_0}(t_0,\vec{x}_p) = \vec{x}_p \quad \text{for the solution} \quad \varphi_t(t_0,\vec{x}_p).$$

$$\text{But } \frac{d}{dt}[\, \varphi_{T+t}(t_0,\vec{x}_p)] = f\Big(t+T,\ \varphi_{T+t}(t_0,\vec{x}_p)\Big)$$

$$= f\Big(t,\ \varphi_{T+t}(t_0,\vec{x}_p)\Big)$$

Hence $\varphi_t(t_0,\vec{x}_p)$ and $\varphi_{T+t}(t_0,\vec{x}_p)$ are two solutions with the same initial condition $\vec{x}_p$. By uniqueness of solutions,

$$\varphi_t(t_0,\vec{x}_p) = \varphi_{T+t}(t_0,\vec{x}_p)$$

Since the continuous dynamics is reduced to the action of a mapping, the rich collection of <u>fixed-point</u> <u>theorems</u> can be utilized to investigate periodic solutions.

This paper will show an efficient technique developed, based on the theory of Poincare' Mapping, that identifies initial conditions associated with periodic solutions for forced linear and nonlinear dynamic systems.

The algorithm developed for locating periodic solutions to linear and nonlinear systems will be reviewed. The process uses modifications to the method of analysis for determining steady state periodic response, with techniques of Poincare' Mapping [4] and the Infinitesimal Generator associated with Lie Series [8,11]. The. main part of the review is outlined in two chapters. Chapter three starts with one dimensional (1-D) linear system analysis and extends the analysis for higher dimension linear systems. Chapter four discusses the analysis of I-D nonlinear systems and continues on to the analysis of higher dimension nonlinear systems.

The use of the symbolic computation mathematics program Maple, will be used for its speed in calculating solutions for differential equations and the generation of series solution expansions. Discussions on how the algorithm is applied and examples will be reviewed.

II DYNAMICAL SYSTEMS

If at any time the output of a system depends on some past input, the system is referred to as dynamic. A dynamical system can be defined as one for which the response of the system will vary with time when it is disturbed or acted upon by some external excitation. An example of a dynamic system is a vibrating spring-mass-damper system shown in the figure below.

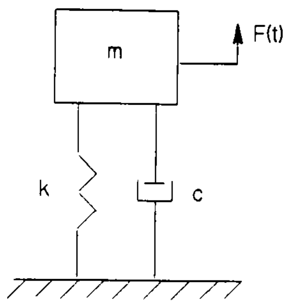


FIG 2-1. Mass-spring-damper system

This dynamic behavior is typically defined by the nonautonomous differential equation

$$m\ddot{x} + c\dot{x} + kx = F(t) \qquad (2.0)$$

Here the dynamic action is the movement of the parts (system response) and can usually be seen or felt. A steadily applied periodic force will cause a vibration that continues with time, with characteristics that can be determined by both the system parameters and those from the input.

Other systems have dynamic characteristics that are governed by the same mathematical differential equations as those used for describing mechanical vibrating systems. Some examples include:

1. Electrical circuits, composed of resistive, capacitive and inductive elements that will oscillate (fluctuate) under the proper type of excitation.

2. Ecological systems. The population of a species of insects or mammals in a given region can vary from year to year because of factors such as the number of predators (and the interaction between predators and prey), disease, weather conditions and food supply.

3. Flow of traffic. Different types of traffic disturbances can result in dynamic characteristic behavior of human beings behind the wheel which can be modeled by differential equations.

The type of information that one wants to know about a dynamic system is essentially the same regardless of the physical details of the system. It is important to know :

1. How the system responds with time for any particular type of disturbance.
2. How long it will take for the dynamic action to dissipate if the disturbance is applied only briefly and then removed
3. Whether the system is stable or if its oscillations will increase in magnitude with time after the disturbance has been removed.

The objective of this investigation is to examine the steady state behavior of linear and nonlinear dynamical systems. The concept of Poincare' Mapping in conjunction with forward advance transformation, will be used to show the effectiveness of the method developed for seeking periodic solutions of these systems.

OBSERVATIONS

An autonomous system is one in which the dependent variable time t, does not appear explicitly in the system equation. A one dimensional (1-D) autonomous system cannot have periodic solutions (except constant solutions). Consider a 1-D autonomous system given by

$$\dot{x} = F(x) \tag{2.1}$$

The state space of this system is simply the real line:

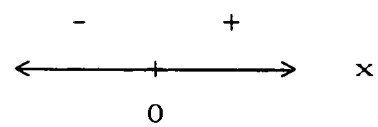


FIG 2-2. Real Line

To maintain a periodic solution, a trajectory must "reverse" direction. But it can't do so without $\dot{x} = 0$. But if $F(x) = 0$, we have an equilibrium point and the point can't move from there.

So the next level is a 1-D non-autonomous system, NA (one where the independent variable, t, appears explicitly):

$$\dot{x} = F(x,t) \tag{2.1a}$$

In particular, suppose that system (2.1a) is driven by a periodic forcing. That is,

$$\dot{x} = F(x) + f(t) \tag{2.2}$$

where f(t) is a periodic input function with period T. The following graphs in figure 2-3 show examples of periodic inputs

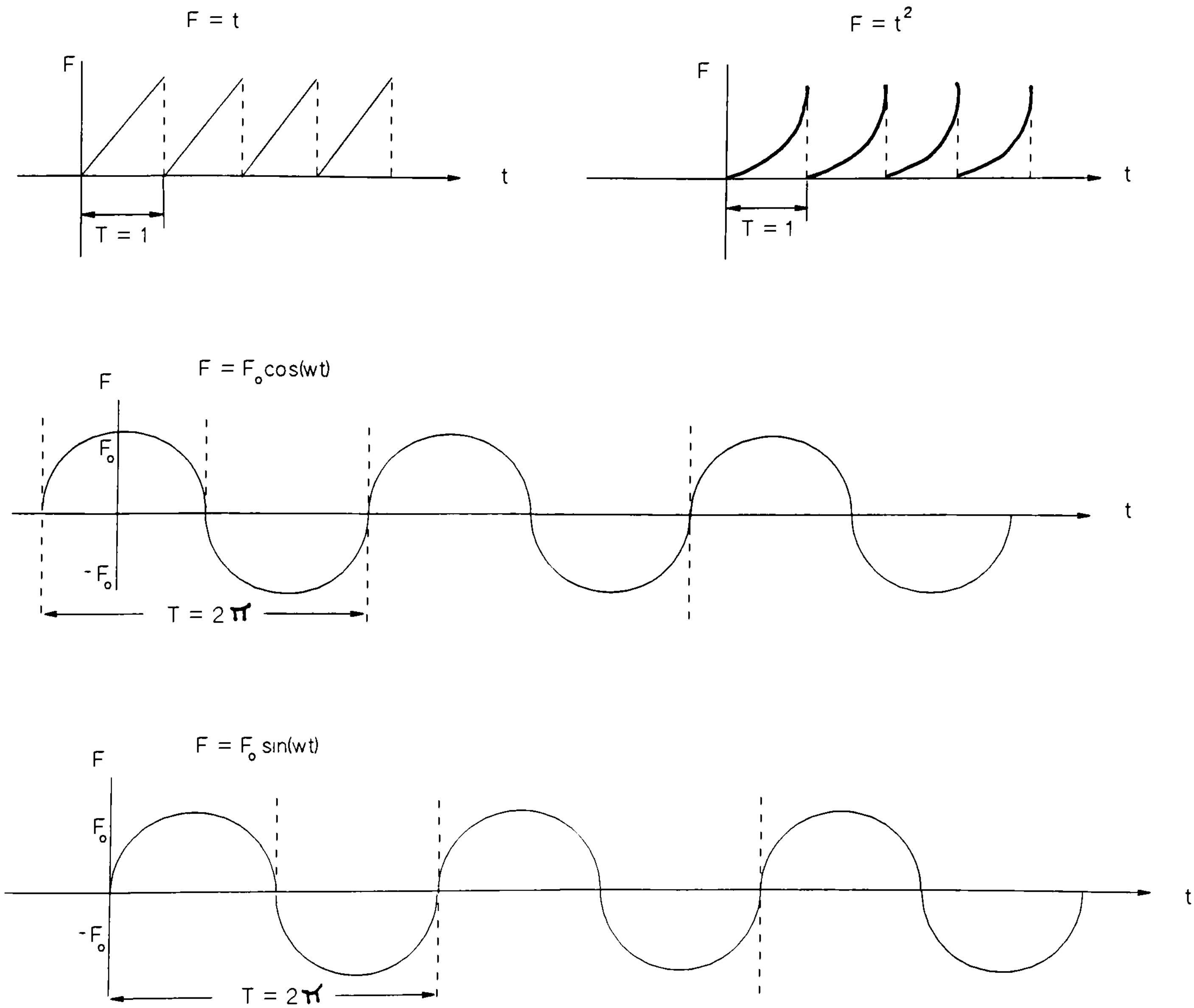


FIG 2-3. Periodic forcing functions

How does one find a periodic solution to eq. (2.2), if it exists? Is there an initial condition, x_0, such that

$$x(0) = x(T) = x_0$$

The objective of this investigation is to develop a systematic way of locating an initial condition that repeats after a period T. As an illustration, consider the forced system given by

$$\dot{x} = x + \sin(t) \tag{1}$$

Is there an initial condition, x_0, such that

$$x_0 = x(0) = x(2\pi)?$$

This analysis is detailed in Case Study # 1 in the 1-D Linear Systems section, using ordinary differential equation techniques [7]. The initial condition giving rise to a periodic solution is found to be

$$x_0 = -\frac{1}{2}$$

Indeed, the general solution is

$$x(t) = -\frac{1}{2}[\sin(t) + \cos(t)] + x_0 e^t + \frac{1}{2}e^t \tag{2.3}$$

When $t = 2\pi$ is substituted into the general solution, it can be shown that for $x_0 = -\frac{1}{2}$,

$$x(0) = x(2\pi).$$

Substituting $t = 2\pi$ into (2.3), we find that

$$x(2\pi) = -\frac{1}{2}[\sin(2\pi) + \cos(2\pi)] + x_0 e^{2\pi} + \frac{1}{2}e^{2\pi} \qquad (2.4)$$

Evaluating the sine, cosine and exponential terms,

$$x(2\pi) = -\frac{1}{2}[0 + 1] + x_0(535.492) + \frac{1}{2}(539.492) \qquad (2.5)$$

which simplifies to

$$x(2\pi) = -\frac{1}{2} + x_0(535.492) + \frac{1}{2}(539.492) \qquad (2.6)$$

Substituting the particular initial condition $\left(x_0 = -\frac{1}{2}\right)$ into (2.6),

$$x(2\pi) = -\frac{1}{2} + (-\tfrac{1}{2})(535.492) + \frac{1}{2}(539.492) \qquad (2.7)$$

$$= -\frac{1}{2}$$

Which again simplifies to the original value for the initial condition,

$$> \quad x(2\pi) = -\frac{1}{2} = x_0$$

PROCESS FOR LOCATING PERIODIC SOLUTION

To locate periodic solutions, we need to develop the concept of <u>Poincare' Mapping</u>. Consider the initial value problem

$$\dot{\vec{x}} = \vec{F}(x) + \vec{f}(t), \quad x(0) = x_0 \qquad (2.8)$$

where $\vec{f}(t+T) = \vec{f}(t)$ "periodic forcing"

Let us "solve" the problem and imagine advancing the solution forward in time to $t = T$. That is, suppose $x(t)$ is a solution of equation (2.8). Keeping track of the flow in state space, the initial value $\vec{x}_0$ is advanced forward to some point $\vec{x}_1 = \vec{x}(T)$. Consider a 1-D example

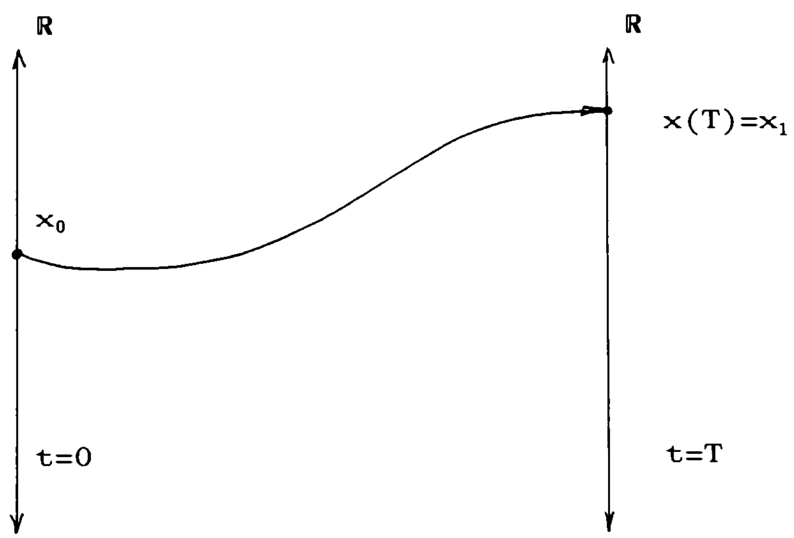


FIG 2-4. 1-D state space

So effectively, the point x_0 is "mapped" to some other point $x_1 = x(T)$. Now, if we allow the <u>initial</u> point to be <u>arbitrary</u>, say just x, in time " T " this point is sent to some unique point in the state space. The dynamical system thus defines a one-to-one mapping from the state space to itself.

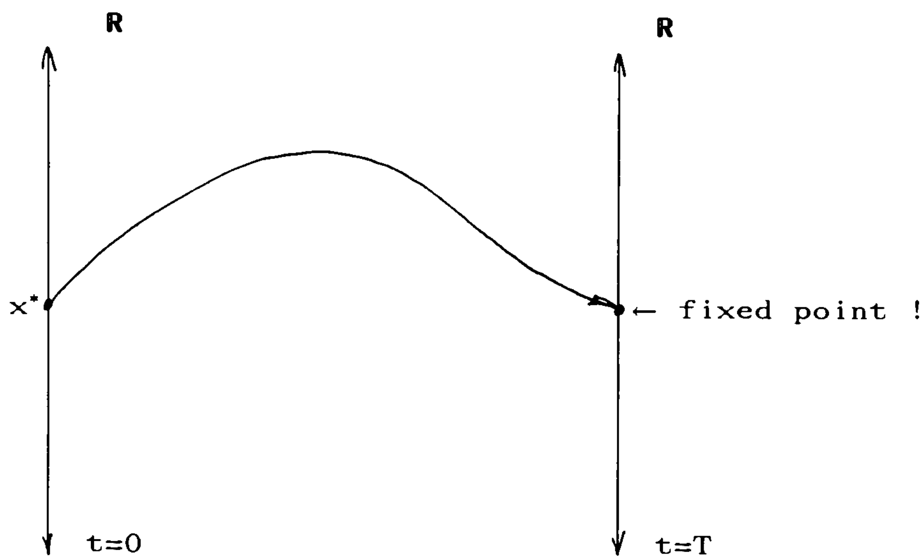


FIG 2-5. 1-D State space

Thus, if a point returns to itself after t = T, it is a <u>fixed point</u> of this forward advance mapping. This means that a particular initial condition, say x^*, gives rise to a solution of period T. Keeping track of solutions and initial conditions is notationally cumbersome. Some <u>notation</u> is needed to keep track of the operations. The solution to

$$\dot{\vec{x}} = \vec{F}(\vec{x}) + \vec{f}(t), \qquad \vec{x}(0) = \vec{x}_0$$

is denoted by

$$\Phi_t(\vec{x}_0) \equiv \text{solution trajectory at time "t",} \qquad (2.9)$$

$$\text{starting at } \vec{x}_0 \text{ !}$$

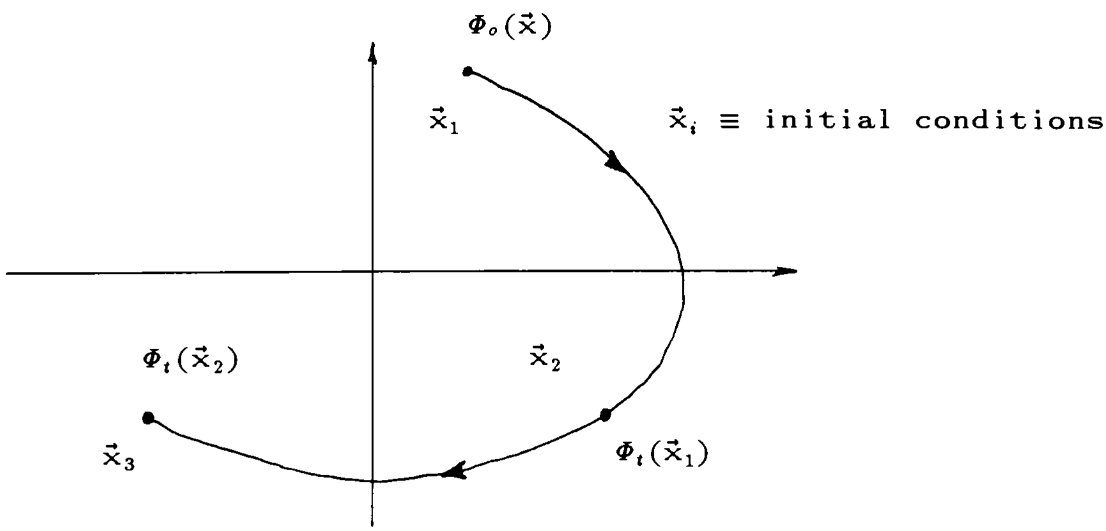
We can denote the solution for <u>arbitrary</u> initial conditions by dropping the subscript. So Φ_t = solution of the Initial Value Problem (IVP), starting at $\vec{x}$, $\Phi_t(\vec{x})$. i.e.

$$\frac{d}{dt}[\Phi_t(\vec{x})] = \vec{F}\Big(\Phi_t(\vec{x})\Big) + f(t) \qquad (2.10)$$

$$\text{with } \Phi_0(\vec{x}) = \vec{x}$$

$\Phi_t(x)$ effectively defines the flow in the state space.

For two dimensional systems, the flow is depicted in Figure 2-5



Note that $\Phi_0(\vec{x}) = \vec{x}$.

FIG 2-6. 2-D state space

$\Phi_t(x)$ describes the evolution of each point in the state space. Thus, $\Phi_t(x)$ is the forward advance mapping (at time "t") starting at $\vec{x}$. This allows us to express with a single symbol a solution in time, starting at an arbitrary point $\vec{x}$.

Examples:

(1) $\dot{x} = x$ general solution: $x(t) = Ae^t$.
Now $x(0) = A$, so the forward advance map is

$$\Phi_t(x)=xe^t \tag{2.11}$$

It is important to keep in mind that now 'x' represents an arbitrary initial condition.

(2) $\dot{x} = -x + t$ general solution: $x(t) = t-1 + Ce^{-t}$

$$x(0) = -1 + C$$

$$\Rightarrow C = x(0) + 1$$

Eliminating the constant of integration results in the general solution

$$x(t) = t - 1 + (x_0 + 1)e^{-t}$$

Letting the initial condition be arbitrary, the forward advance mapping is explicitly given by

$$\Phi_t(x) = t - 1 + (x+1)e^{-t} \tag{2.12}$$

That is, after a time t any initial value $x \in \mathbb{R}$ gets mapped to

$$(t-1) + (x+1)e^{-t}$$

(3) $\ddot{x} + x = 0$ general solution:

$$x(t) = A\sin t + B\cos t$$

$$= \dot{x}(0)\sin t + x(0)\cos t$$

As a first order system:

$$\dot{x} = y$$
$$\dot{y} = -x$$

the solution can be written in the form

$$\begin{Bmatrix} x(t) = y(0)\sin t + x(0)\cos t \\ y(t) = y(0)\cos t - x(0)\sin t \end{Bmatrix}$$

Letting $\begin{Bmatrix} x \\ y \end{Bmatrix}$ be arbitrary initial conditions, we have

$$\vec{\Phi}_t(\vec{x}) = \begin{bmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{bmatrix} \vec{x} \qquad (2.13)$$

or

$$\Phi_t(\vec{x}) = [A_t]\vec{x} \qquad (2.13a)$$

where A_t is a time-dependent operator.

Phase Space:

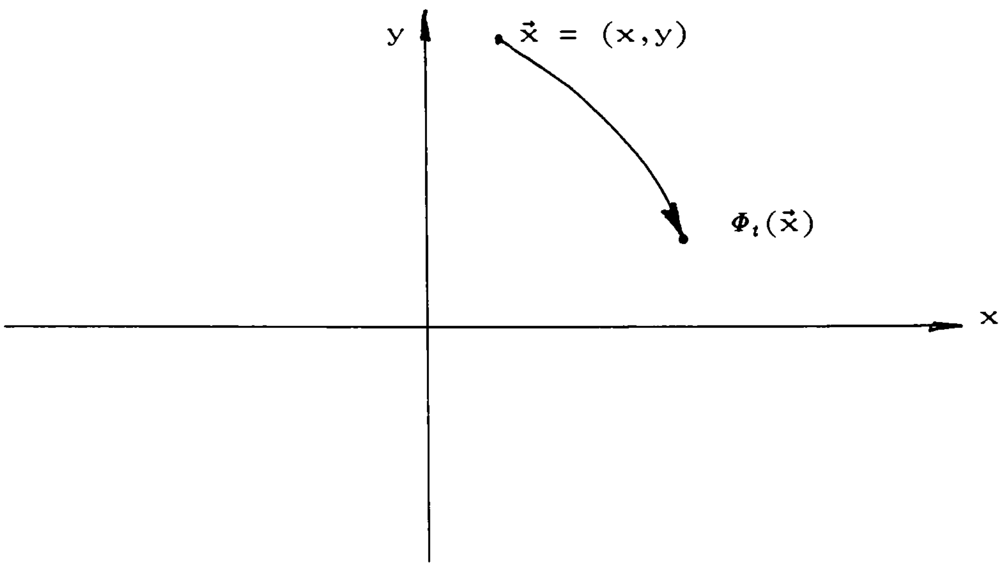


FIG 2-7. Phase space

So the evolution over a fixed time interval is thought of as a time-advance mapping. Now we can define Poincare' Mapping:

POINCARE' MAPPING

Consider the flow, $\Phi_t(\vec{x})$, now associated with a dynamical system. If we fix the time advance to some specific value, say $t = T$, then $\Phi_T(\vec{x})$ becomes simply a mapping (time is fixed) that takes every point in the state space to some other point (freeze trajectory points at $t = T$). That is, every initial condition in the phase space has traveled (via the synopsis) to some point at $t=T$. This flow effectively defines the Poincare' Mapping. We denote it by

$$\mathbf{P}(x) = \Phi_T(x) \tag{2.14}$$

(For instance, in example # 2, $\mathbb{P}(x) = T - 1 + (x+1)e^{-T}$)

We have now redefined the problem in terms of a mapping. There are now three things that can be accomplished with this mapping:

(1) Determine limit sets (if they exist).

(2) Find periodic solutions.

(3) Examine stability of the system.

Asymptotic Behavior

To find where a particular point (initial condition) ends up as $t \to \infty$, repeatedly apply the map. Let x be an arbitrary initial condition. Its asymptotic behavior is obtained by iterating the associated Poincare′ mapping:

$$\Phi_T(x) = \mathbb{P}(x)$$

$$\Phi_{2T}(x) = \mathbb{P}[\mathbb{P}(x)] \qquad \leftarrow\text{--image after T, of point starting at } \mathbb{P}(x)$$

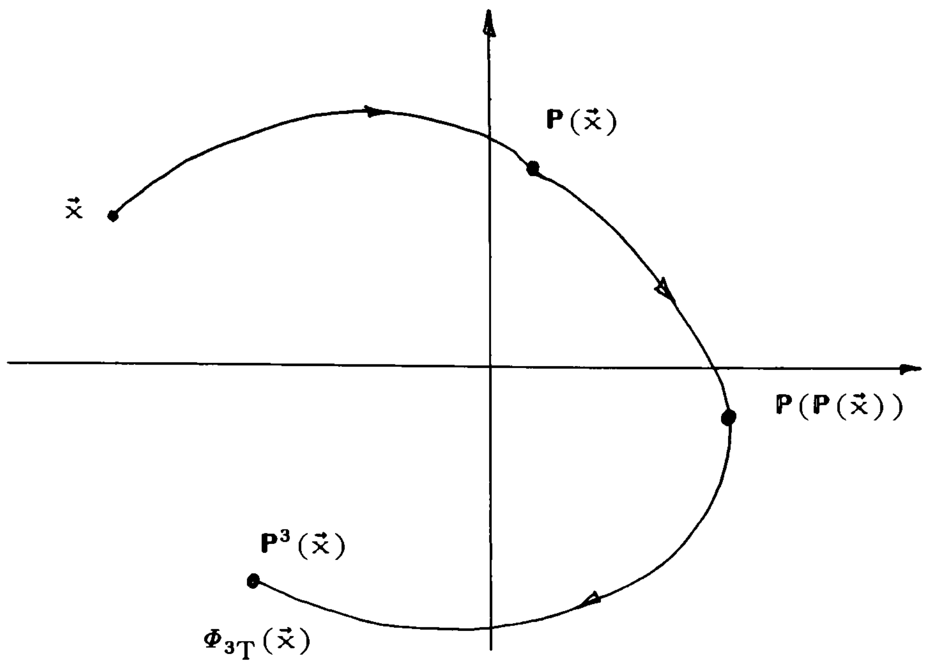
$$\Phi_{3T}(x) = \mathbb{P}(\mathbb{P}[\mathbb{P}(x)]) \tag{2.15}$$

$$= \mathbb{P}^3(x) \qquad \leftarrow\text{-------- composition}$$

$$\Phi_{nT}(x) = \mathbb{P}^n(x) = \mathbb{P}.\mathbb{P}...(\mathbb{P}(x)) \qquad \leftarrow\text{---- composition}$$

The long-term behavior is determined by computing the limit

$\lim_{n\to\infty} \mathbb{P}^n(x)$, if it exists.



||

"image" of x after t = 3T .

FIG 2-8. Phase space plot of Poincare' mapped point

So determination of asymptotic behavior reduces to iteration of the associated Poincare' Map.

Example: Let $\mathbb{P}(x) = T-1 + (x+1)e^{-T}$ (T fixed)

Let $x = x_0$, then $x_1 = \mathbb{P}(x_0)$

$$x_2 = \mathbb{P}(x_1) = \mathbb{P}(\mathbb{P}(x_0))$$
$$x_3 = \mathbb{P}(x_2) = \mathbb{P}(\mathbb{P}(\mathbb{P}(x_0)))$$
$$\vdots$$
$$x_{n+1} = \mathbb{P}(x_n) \qquad (2.16)$$

x_∞ = limit after infinite time (if it exists)

Example: Find the limit of any solution to

$$\dot{x} = -x + t \qquad \text{(period } T = 1)$$

The initial condition giving rise to a periodic limit solution was found to be

$$x_0 = 0.582$$

In fact, all initial conditions converge to this value in the limit. See Chapter III, 1-D Linear Systems, Case Study 2 for problem detail.

<u>Periodic Solutions</u>

The main focus of this investigation is the determination of periodic solutions (if they exist). Periodic means that the solution <u>repeats</u> itself. This requires finding a <u>fixed point</u> of the Poincare' Map. Thus x^* gives rise to a periodic solution if

$$\vec{x}^* = \mathbb{P}(x^*) = \Phi_T(x^*) \qquad (2.17)$$

i.e, after time T, x^* returns to x^*.

The <u>Poincare' Mapping</u> is the tool we will use to locate periodic solutions. If the Poincare mapping can be constructed, these periodic solutions are readily computed. In most cases, however, the Poincare' mapping must be approximated. This important aspect will be discussed in the subsequent chapters.

III LINEAR SYSTEMS

An important characteristic to know about a system is whether it is linear or nonlinear. This will influence the solution methods used to analyze the system equations. A linear system is defined as one in which the dependent variables describing the system must be either of first or zero power (absent), and contains no products of the dependent variables [2]. For the system in the figure below,

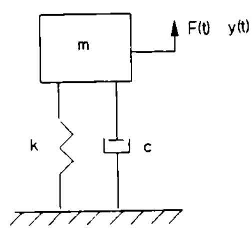


FIG 3-1. Mass-spring-damper system

the equation of motion is described by a linear differential equation. The differential equation of the system is considered linear,

$$m\frac{d^2}{dt^2}y(t) + c\,\frac{d}{dt}y(t) + ky(t) = F(t) \qquad (3.1)$$

A primary attribute of a linear system is the associated superposition principle, thereby allowing the use of analytical techniques such as Modal Analysis and Fourier Analysis [2].

The relation between the excitation F(t) and response y(t) of equation (3.1) can be described by the following block diagram:

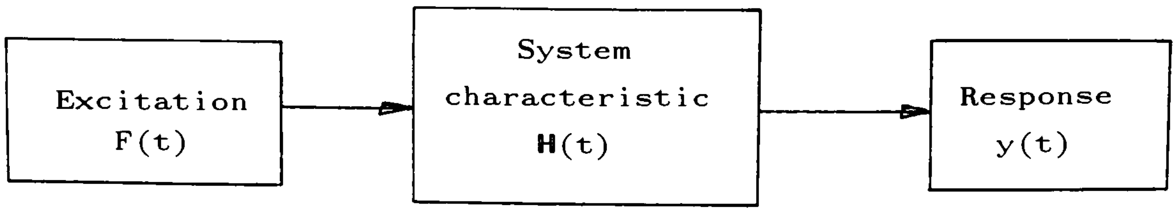


FIG 3-2. Block diagram representation of system

where $\mathbb{H}(t)$ is the system characteristic in the form of the linear differential operator

$$\mathbb{H}(t) = m \frac{d^2}{dt^2} + c \frac{d}{dt} + k \qquad (3.2)$$

A simple way of testing whether a system is linear or nonlinear is by use of the superposition principle. The principle states that the response produced by simultaneous applications of two different forcing functions or inputs is the sum of two individual responses. If $y_1(t)$ is a solution response of the system to the excitation $F_1(t)$, and $y_2(t)$ is the response to another excitation $F_2(t)$ applied to the system. In terms of the above linear operator notation, we can write

$$F_1(t) = \mathbb{H}[y_1(t)], \quad F_2(t) = \mathbb{H}[y_2(t)], \qquad (3.3)$$

and

$$F_3(t) = c_1 F_1(t) + c_2 F_2(t), \qquad (3.4)$$

where c_1 and c_2 are arbitrary constants. You can also write

$$F_3(t) = \mathbb{H}_3[y_3(t)] = c_1 \mathbb{H}[y_1(t)] + c_2 \mathbb{H}[y_2(t)], \qquad (3.5)$$

which is also a solution of the linear system. In terms of the operator notation expression,

$$\mathbf{H}[c_1y_1 + c_2y_2] = c_1\,\mathbf{H}[y_1] + c_2\,\mathbf{H}[y_2] \tag{3.6}$$

represents the statement that the operator **H** is linear, implying that the superposition principle holds true for the system whose characteristics are described by **H**. If on the other hand,

$$F_3(t) = \mathbf{H}_3[y_3(t)] \neq c_1\,\mathbf{H}[y_1(t)] + c_2\,\mathbf{H}[y_2(t)] \tag{3.7}$$

the system is considered nonlinear.

So, for linear systems that have several inputs, the response to several inputs can be calculated by dealing with one input at a time and then adding the results. As a result of the principle of superposition, complicated solutions to linear differential equations can be derived as a sum of simple solutions.

FUNDAMENTAL SOLUTIONS / FUNDAMENTAL MATRIX

Consider the vibrating system in figure 3.1. The system is acted upon by an excitation force F(t), and the system behavior is defined by the displacement y(t) of mass m. Using Newton's second law, it can be shown that the system's displacement must satisfy the differential equation (3.1)

$$m\frac{d^2}{dt^2}y(t) + c\,\frac{d}{dt}y(t) + ky(t) = f(t) \tag{3.8}$$

where the coefficients m, c and k are constants. The standard procedure in State Variable Analysis is to put the system equations into simultaneous first order form [10]. The simplest way to do this is to define a new state variable, the velocity v(t). That is,

$$v(t) = \frac{d}{dt} y(t) \tag{3.9}$$

Now, substituting for $\frac{d^2}{dt^2}y(t)$ $\left(\text{which is } \frac{d}{dt}v(t)\right)$ and $\frac{d}{dt}y(t)$, the system equation can be expressed as

$$m\frac{d}{dt}v(t) + c\, v(t) + ky(t) = f(t) \tag{3.10}$$

or

$$\frac{d}{dt}v(t) = -\frac{c}{m}v(t) - \frac{k}{m}y(t) + \frac{1}{m}f(t) \tag{3.11}$$

The second order system (3.8) is reduced to two first order equations. These are State Equations defining the mechanical system:

$$\frac{dx}{dt} = v(t) \tag{3.12}$$

$$\frac{d}{dt}v(t) = -\frac{c}{m}v(t) - \frac{k}{m}x(t) + \frac{1}{m}f(t) \tag{3.13}$$

In matrix form,

$$\begin{bmatrix} \frac{d}{dt}x(t) \\ \frac{d}{dt}v(t) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \begin{bmatrix} x(t) \\ v(t) \end{bmatrix} + \begin{bmatrix} 0 & 1 \\ \frac{1}{m} & 0 \end{bmatrix} \begin{bmatrix} f(t) \\ 0 \end{bmatrix} \tag{3.14}$$

Higher order systems can be transformed in a similar manner. Given

$$y^{(n)}(t) + a_{n-1}\, y^{(n-1)}(t) + \ldots\ a_1\dot{y}(t) + a_0 y(t) = f(t) \qquad (3.15)$$

the state variables are defined as

$$x_1 = y$$
$$x_2 = \dot{y}$$
$$x_3 = \ddot{y}$$
$$x_n = y^{(n-1)}$$

The derivatives of $x_1, \ldots, x_{n-1}$ are obtained from the first n-1 equations. The derivative of x_n is obtained from the original differential equation (3.11).

FUNDAMENTAL MATRIX

Another method of solving for the response of multi-degree-of freedom linear system (very convenient for numerical computation) is the use of the fundamental matrix.

If [A] is a constant n X n matrix then the power series for any [A] is represented by the following [14],

$$e^{[A]t} = \sum_{k=0}^{\infty} \frac{t^k}{k!}[A]^k = [I] + [A]t + \frac{t^2}{2!}[A]^2 + \frac{t^3}{3!}[A]^3 \ldots\ldots (3.16)$$

The displacement vector $\{x(t)\}$ and the velocity $\{\dot{x}(t)\}$ of an n-degree of freedom system define the state of the system. They can be arranged in a 2n-dimensional vector of the form

$$\{y(t)\} = \begin{Bmatrix} \{x(t)\} \\ \{\dot{x}(t)\} \end{Bmatrix} \tag{3.17}$$

Similarly, you can introduce the 2n-dimensional forcing vector

$$\{F(t)\} = \begin{Bmatrix} \{0\} \\ \{f(t)\} \end{Bmatrix} \tag{3.18}$$

where $\{y(t)\}$ is known as the state vector and $\{F(t)\}$ is the force vector. The equation of motion of an n-degree of freedom linear system can be written in the general matrix form,

$$\{\dot{y}(t)\} = [A]\{y(t)\} + [B]\{F(t)\} \tag{3.19}$$

where [A] and [B] are 2n X 2n matrices of coefficients, depending on the nature of the system.

To obtain a solution of the above equation, first consider the homogenous equation,

$$\{\dot{y}(t)\} = [A]\{y(t)\} \tag{3.20}$$

This matrix equation is similar in structure to the scaler first-order differential equation. Letting $\{y(0)\}$ be the initial state vector, the solution of the homogeneous equation above can be verified to be

$$\{y(t)\} = e^{[A]t}\{y(0)\} \tag{3.21}$$

where $e^{[A]t}$ is the series matrix as defined previously in equation (3.16),

$$e^{[A]t} = [I] + [A]t + \frac{t^2}{2!}[A]^2 + \frac{t^3}{3!}[A]^3 \ldots \quad (3.16)$$

Looking at the nonhomogenous equation (3.19) introduce a 2n X 2n matrix [K(t)], premultiply the equation by [K(t)], and obtain

$$[K(t)]\{\dot{y}(t)\} = [K(t)][A]\{y(t)\} + [K(t)][B]\{F(t)\} \quad (3.22)$$

Now

$$\frac{d}{dt}\{[K(t)]\{y(t)\}\} = [\dot{K}(t)]\{y(t)\} + [K(t)]\{\dot{y}(t)\} \quad (3.23)$$

so equation (3.22) can be written as

$$\frac{d}{dt}\Big\{[K(t)]\{y(t)\}\Big\} - [\dot{K}(t)]\{y(t)\} =$$

$$[K(t)][A]\{y(t)\} + [K(t)][B]\{F(t)\} \quad (3.24)$$

Next, choose [K(T)] so as to satisfy

$$[\dot{K}(t)] = - [A][K(t)] \quad (3.25)$$

which has the solution

$$[K(t)] = e^{-[A]t}[K(0)] \quad (3.26)$$

For convenience, we choose [K(0)] as the identity matrix, or

$$[K(0)] = [I] \tag{3.27}$$

so that equation (3.26) reduces to

$$[K(t)] = e^{-[A]t} \tag{3.28}$$

From equations (3.28) and (3.16) we observe that the matrices [K(t)] and [A] commute (same 2n X 2n order), or

$$[A][K(t)] = [K(t)][A] \tag{3.29}$$

Substituting equation (3.29) into equation (3.25), we can see the matrix [K(t)] also satisfies

$$[\dot{K}(t)] = -[K(t)][A] \tag{3.30}$$

so equation (3.24) can be reduced to

$$\frac{d}{dt}\{[K(t)]\{y(t)\}\} = [K(t)][B]\{F(t)\} \tag{3.31}$$

So to complete the solution of equation (3.19), you have to solve equation (3.31) above. Integrating equation (3.31) yields

$$[K(t)]\{y(t)\} = [K(0)]\{y(0)\} + \int_0^t [K(\tau)][B]\{F(\tau)\}d\tau$$

$$= \{y(0)\} + \int_0^t [K(\tau)][B]\{F(\tau)\}d\tau \tag{3.32}$$

premultiplying equation (3.32) by $[K(t)]^{-1}$, yields the solution of the nonhomogenous equation (3.19) in the form,

$$\{y(t)\} = [K(t)]^{-1}\{y(0)\} + \int_0^t [K(t)]^{-1}[K(\tau)][B]\{F(\tau)\}d\tau$$

$$= e^{[A]t}\{y(0)\} + \int_0^t e^{[A](t-\tau)}[B]\{F(\tau)\}d\tau \qquad (3.33)$$

Equation (3.33) contains the same solution as in equation (3.24) for the homogenous case. Since both the homogenous and particular solutions are present, this is the complete solution for equation (3.19).

FORCED SOLUTIONS

The behavior determined by a forcing function is called a forced response and that, due to initial energy storage, is the natural response. The time between the starting and the ending of the natural response is the transient response. After the natural response has become negligibly small, conditions are said to be in steady state.

The differential equation of motion for a second order linear system (mass-damper-spring) with arbitrary forcing f(t) is

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = f(t) \qquad (3.34)$$

where the excitation force, f(t), is chosen to be harmonic. The simplest form is

$$f(t) = A\cos(\omega t) \qquad (3.35)$$

where ω is the excitation frequency or the driving frequency. Because the excitation force is harmonic, it can be shown that the steady state response is also harmonic and has the same frequency ω [2]. Inserting the expression for f(t) into the differential equation (3.34) and divide through by m to separate the higher order term of x. The steady state solution has the form

$$x(t) = C_1 \sin \omega t + C_2 \cos \omega t \tag{3.36}$$

Inserting this into equation (3.34) results in

$$\ddot{x}(t) + 2 \zeta \omega_n \dot{x} + \omega_n^2 x(t) = \frac{A}{m} \cos \omega t \tag{3.37}$$

where ζ is the viscous damping factor and ω_n is the natural frequency of undamped oscillation. The steady state solution is readily expressed as

$$x(t) = \frac{A k^{-1}}{[1-(\omega/\omega_n)^2]^2 + (2\zeta\omega/\omega_n)^2} \left\{ \frac{2\zeta}{\omega_n} \sin \omega t + \left(1 - \left(\frac{\omega}{\omega_n}\right)^2\right) \cos \omega t \right\} \tag{3.38}$$

where

$$A = \text{Constant}$$

$$\zeta = \frac{c}{2m\omega_n}$$

$$\omega_n = \sqrt{\frac{k}{m}}$$

1-D Linear Systems

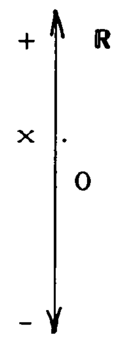


FIG 3-3. State space: $\mathbb{R}$, the real line

Equation:

$$\dot{x} = ax + f(t) \tag{3.39}$$

where f(t) is periodic. that is, $f(t + T) = f(t)$.

$T \equiv$ forcing period.

For a 1-D linear system we can solve equation (3.39) exactly, and then find the exact formula for the Poincare' Map.

Two methods of solution:

1. This is a first order linear equation

$$x(t) = e^{at}x(0) + \int_0^t e^{-a\tau}f(\tau)\, d\tau \tag{3.40}$$

2. Use Laplace Transform.

<u>Poincare' Map</u>

Recall that the Poincare' map assigns <u>state</u> values at t = 0 to state values at t = T (end of period)

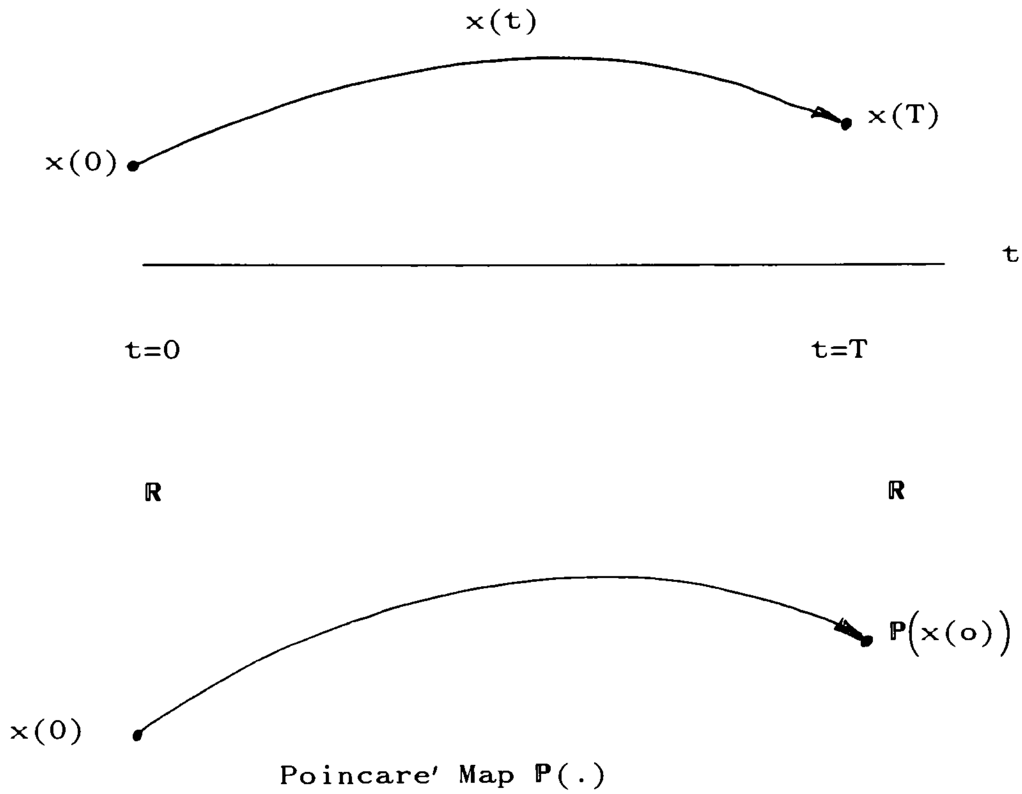


FIG 3-4. Poincare' mapping of state values

$$x(0) \quad \Rightarrow \quad \mathbb{P}\big(x(0)\big) \quad = \quad x(T)$$

⇑ but this depends on x(0), so we have a mapping.

Going back to the exact solution of the differential equation (3.40) and substituting t=T,

$$x(T) = e^{aT} x(0) + \int_0^T e^{-a\tau} f(\tau) d\tau \qquad (3.41)$$

This equation determines the Poincare' map. Given x(0), we now have an explicit expression for x(T), i.e, x(0) is mapped to x(T) by the dynamics of the problem. The dynamics is completely defined by equation (3.41). So allowing the initial condition to be arbitrary, that is, setting x(0) to be x, we find that the point x goes to (after one period)

$$e^{aT} x + \int_0^T e^{-a\tau} f(\tau) d\tau \qquad (3.42)$$

Hence the Poincare' map (corresponding to the forcing period T) is

$$\mathbf{P}(x) = e^{aT} x + \int_0^T e^{-a\tau} f(\tau) d\tau \qquad (3.43)$$

This holds for any forcing function of period T.

Finally, to determine the periodic solution to our problem, it means that we are looking for a fixed point of the Poincare' map $\mathbb{P}(x)$. That is, a periodic solution is obtained by finding an initial condition x^* such that

$$x^* = \mathbf{P}(x^*) \qquad (3.44)$$

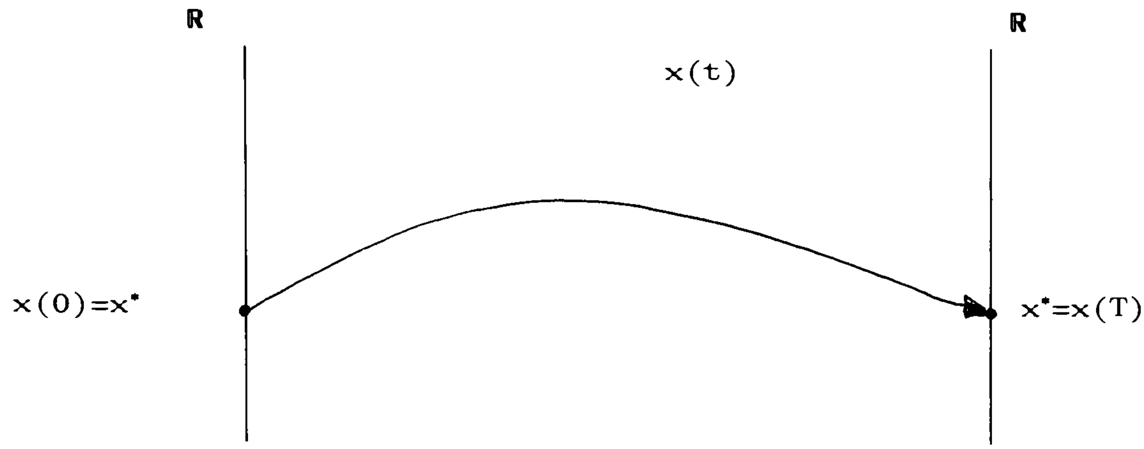


FIG 3-5. Poincare' mapping of a fixed initial point x^*

Thus if $x^* = \mathbb{P}(x^*)$, a <u>fixed point</u> of the Poincare' map, then $x(0)=x^*$ is the initial condition that gives rise to a <u>periodic solution</u> to the differential equation.

The following example problems are solved, using this technique to find the <u>Poincare' map</u> and the <u>fixed point</u>. Plots of the responses showing periodic solution also follows each example.

1. $\dot{x} = -x + t$, T=1 (period) CASE STUDY # 2

2. $\dot{x} = x + t^2$, T=1 CASE STUDY # 3

3. $\dot{x} = -2x + 4\sin^2(t)$, $T=\pi$ CASE STUDY # 4

CASE STUDY # 1

$$\dot{x} = x + \sin(t) \tag{1}$$

Objective: Is there an initial condition x_0, such that $x_0 = x(0) = x(2\pi)$?

Solution:

integrating factor- $e^{\int -dt} = e^{-t}$

multiply both sides of equation (1) by e^{-t}

$$e^{-t}\left(\frac{dx}{dt} - x\right) = e^{-t}[\sin(t)]$$

$$\frac{d}{dt}\left(xe^{-t}\right) = e^{-t}\sin(t)$$

integrating

$$xe^{-t} = \int e^{-t}\sin(t)\, dt$$

$$xe^{-t} = -\frac{e^{-t}}{2}[\sin(t) + \cos(t)] + C$$

$$x(t) = -\tfrac{1}{2}[\sin(t) + \cos(t)] + Ce^{t} \tag{2}$$

Define constant C in terms of initial condition x_0. At t=0, $x(0) = x_0$, so from (2),

$$x_0 = -\frac{1}{2}[0 + 1] + C$$

$$x_0 = -\frac{1}{2} + C$$

$$C = x_0 + \frac{1}{2}$$

$$x(t) = -\frac{1}{2}[\sin(t) + \cos(t)] + x_0 e^t + \frac{1}{2}e^t \qquad (3)$$

this is the solution for arbitrary x_0.

When $t = \tau$ (for this case $\tau = 2\pi$) is $x(2\pi) = x_0$?

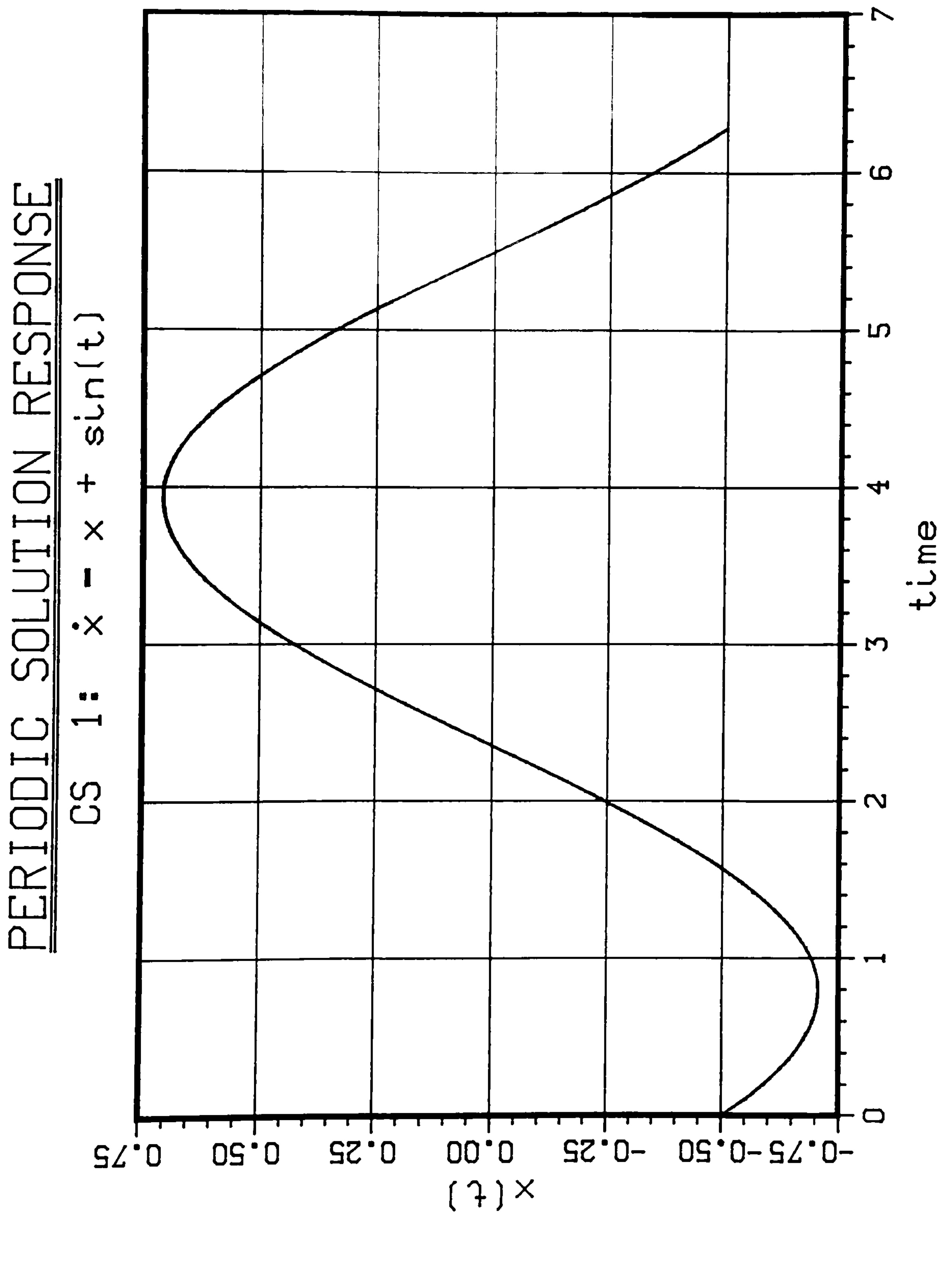
$$x_0 = x(2\pi) = -\frac{1}{2}[\sin(2\pi) + \cos(2\pi)] + x_0 e^{2\pi} + \frac{1}{2}e^{2\pi}$$

$$x_0 = -\frac{1}{2}[0 + 1] + x_0 e^{2\pi} + \frac{1}{2}e^{2\pi}$$

$$x_0 - x_0 e^{2\pi} = -\frac{1}{2} + \frac{1}{2}e^{2\pi}$$

$$x_0 = \frac{(-\frac{1}{2} + \frac{1}{2}e^{2\pi})}{(1 - e^{2\pi})} = \frac{267.2459}{-534.4918} = -\frac{1}{2}$$

$x_0 = -\frac{1}{2}$ generates a periodic solution


PERIODIC SOLUTION RESPONSE
CS 1: ẋ = x + sin(t)
x(t)
0.75
0.50
0.25
0.00
-0.25
-0.50
-0.75
time
0
1
2
3
4
5
6
7

CASE STUDY # 2

Given: $$\dot{x} = -x + t \tag{1}$$

Objective: Find the limit of any solution ?

Solution:

integrating factor- $e^{\int dt} = e^t$

multiply both sides of equation (1) by e^t

$$e^t\left(\frac{dx}{dt} + x\right) = e^t\ t$$

$$\frac{d}{dt}\left(xe^t\right) = e^t\ t$$

integrating

$$xe^t = \int e^t\ t\ dt$$

$$xe^t = (t-1)e^t + C$$

$$x(t) = (t-1) + Ce^{-t} \tag{2}$$

Define constant C in terms of initial condition x_0. At $t=0$, $x(0) = x_0$, so from (2),

$$x_0 = (0 - 1) + C$$

$$x_0 = -1 + C$$

$$C = x_0 + 1$$

$$x(t) = t - 1 + x_0 e^{-t} + e^{-t} \quad (3)$$

$$\Phi_t(x) = t - 1 + xe^{-t} + e^{-t} \quad (4)$$

this is the solution for arbitrary x.

The Poincare' map is given by

$$\underline{P(x)} = \Phi_T(x) = T - 1 + xe^{-T} + e^{-T} \quad (5)$$

Solving for initial condition x_0 from (3),

$$x_0 = t - 1 + x_0 e^{-t} + e^{-t}$$

$$x_0 - x_0 e^{-t} = t - 1 + e^{-t}$$

$$x_0(1 - e^{-t}) = t - 1 + e^{-t}$$

$$x_0 = \frac{(t - 1 + e^{-t})}{(1 - e^{-t})}$$

when t = T (period),

$$x_0 = \frac{(T - 1 + e^{-T})}{(1 - e^{-T})}$$

for this case T = 1,

$$x_0 = \frac{(1 - 1 + e^{-1})}{(1 - e^{-1})}$$

$$x_0 = \frac{(e^{-1})}{(1 - e^{-1})} = x_1 = 0.5820$$

Now let $x = x_0$ and substitute into $\mathbb{P}(x)$ equation (5) to get the next value x_1 with t = T,

$$x_1 = \mathbb{P}(x_0) = T - 1 + \left(0.5820\right) e^{-T} + e^{-T}$$

again T = 1,

$$x_1 = \mathbb{P}(x_0) = 1 - 1 + \left(0.5820\right)e^{-1} + e^{-1}$$

$$x_1 = e^{-1}\left(1 + 0.5820\right)$$

$$x_1 = \frac{1.5820}{e} = \frac{1.5820}{2.7183} = 0.5820$$

similarly,

$$x_2 = \mathbb{P}(x_1) = T - 1 + \left(0.5820\right) e^{-T} + e^{-T}$$

again T = 1,

$$x_2 = \mathbb{P}(x_1) = 1 - 1 + \left(0.5820\right)e^{-1} + e^{-1}$$

$$x_2 = e^{-1}\left(1 + 0.5820\right)$$

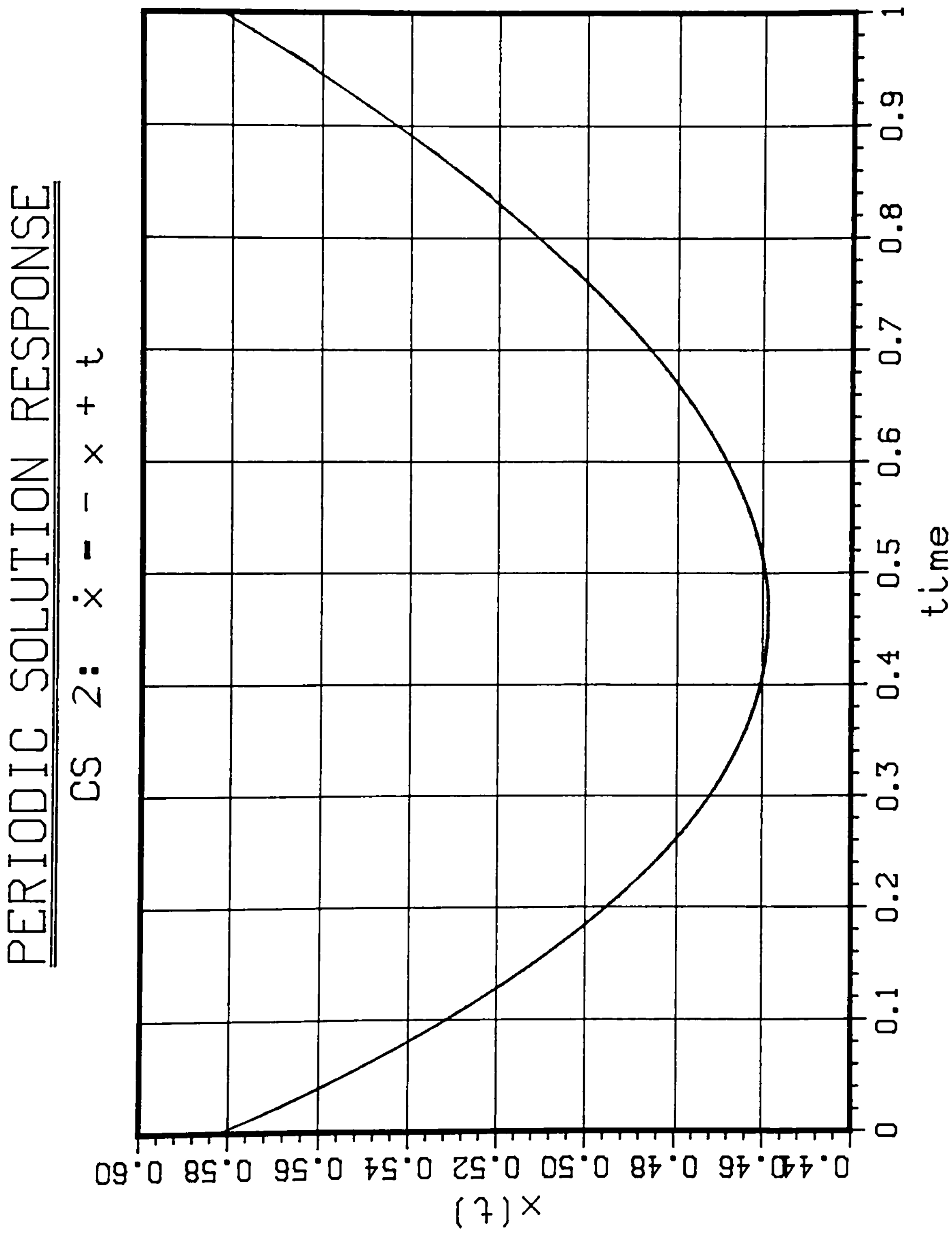
$$x_2 = \frac{1.5820}{e} = \frac{1.5820}{2.7183} = 0.5820$$

similarly,

$$x_3 = \mathbf{P}(x_2) = 0.5820$$

so,

$$x_0 = x_1 = x_2 = x_3 = .. = 0.5820 \quad \rightarrow \text{limit}$$


PERIODIC SOLUTION RESPONSE
CS 2: ẋ = - x + t
x(t)
0.40 0.42 0.44 0.46 0.48 0.50 0.52 0.54 0.56 0.58 0.60
time
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

CASE STUDY # 3

Given: $$\dot{x} = x + t^2 \tag{1}$$

Objective: Find the limit of any solution ?

Solution:

integrating factor- $e^{\int -dt} = e^{-t}$

multiply both sides of equation (1) by e^{-t}

$$e^{-t}\left(\frac{dx}{dt} + x\right) = e^{-t} t^2$$

$$\frac{d}{dt}\left(xe^{-t}\right) = e^{-t} t^2$$

integrating

$$xe^{-t} = \int e^{-t} t^2 dt$$

From CRC handbook,

$$x(t) = - t^2 - 2t - 2 + Ce^t \tag{2}$$

Define constant C in terms of initial condition x_0. At t=0, $x(0) = x_0$, so from (2),

$$x_0 = -2 + C$$

$$C = x_0 + 2$$

$$x(t) = -t^2 - 2t - 2 + [x_0 + 2]e^t \qquad (3)$$

this is the solution for arbitrary x_0.

When $t = \tau$ (period),

$$\mathbb{P}(x) = -\tau^2 - 2\tau - 2 + [x_0 + 2]e^\tau \qquad (4)$$

Solving for initial condition x_0 (3),

$$x_0 = t^2 - 2t - 2 + x_0 e^t + 2e^t$$

$$x_0 - x_0 e^t = -t^2 - 2t - 2 + 2e^t$$

$$x_0(1 - e^t) = -t^2 - 2t - 2 + 2e^t$$

$$x_0 = \frac{(-t^2 - 2t - 2 + 2e^t)}{(1 - e^t)}$$

when $t = \tau$,

$$x_0 = \frac{(-\tau^2 - 2\tau - 2 + 2e^\tau)}{(1 - e^\tau)}$$

for this case $\tau = 1$,

$$x_0 = \frac{(-1 - 2 - 2 + 2(2.7182818)}{(1 - 2.7182818)}$$

$$x_0 = \frac{(-5 + 5.43656)}{(-1.7182818)}$$

$$x_0 = -0.25406 = x_1 = -0.25406$$

Now substitute x_0 into $\mathbb{P}(x)$ equation (4) to get the next value x_2 with $t = \tau$,

$$x_2 = \mathbb{P}(x_1) = -\tau^2 - 2\tau - 2 + [x_0 + 2]e^{\tau}$$

again $\tau = 1$,

$$x_2 = \mathbb{P}(x_1) = -1 - 2 - 2 + [-0.25406 + 2]2.7182818$$

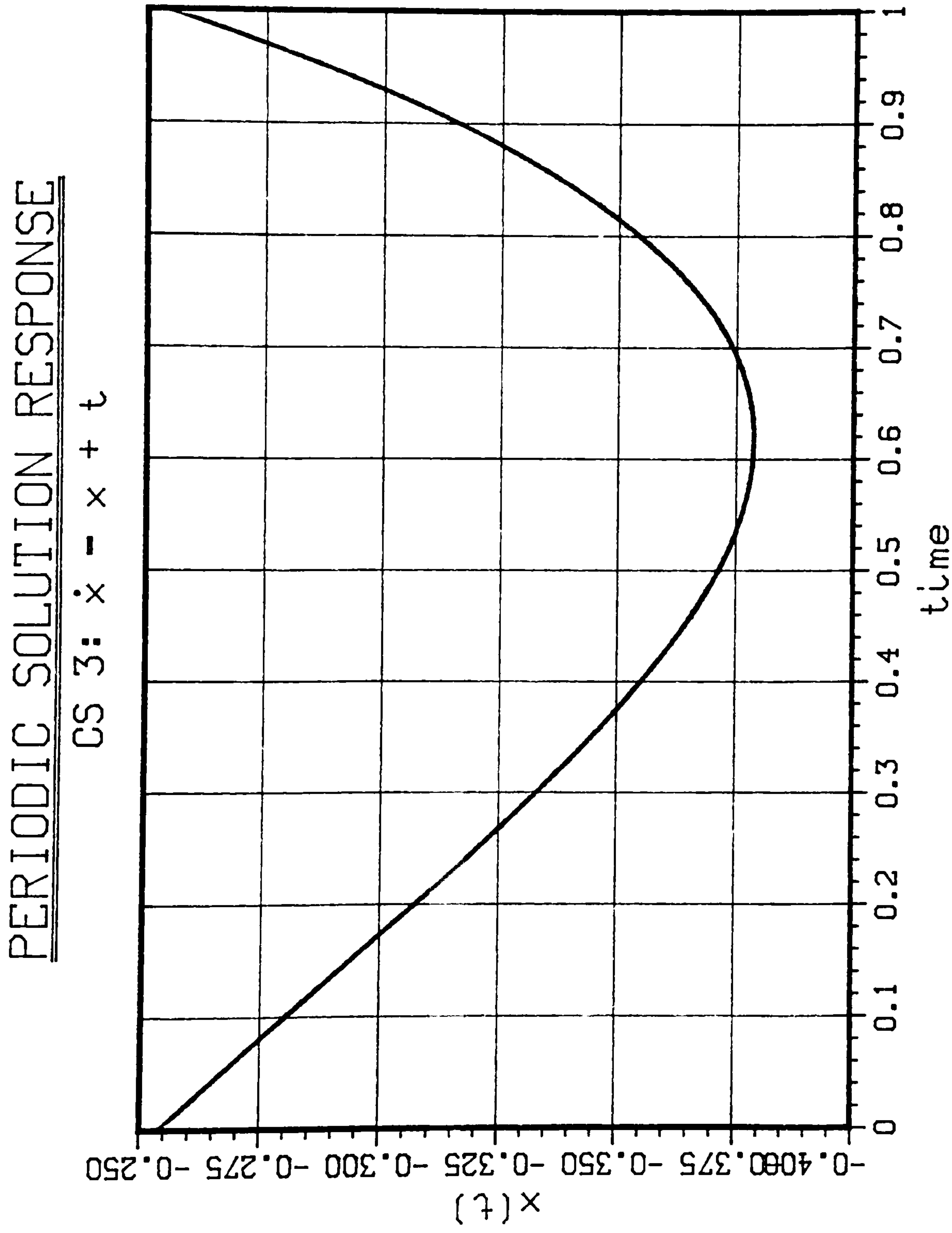
$$x_2 = -0.2505$$

similarly,

$$x_3 = \mathbb{P}(x_2) = -0.2505$$

so,

$$x_0 = x_1 = x_2 = x_3 = .. = -0.2505 \quad \rightarrow \text{limit}$$


PERIODIC SOLUTION RESPONSE
CS 3: ẋ = x + t
x(t)
-0.250
-0.275
-0.300
-0.325
-0.350
-0.375
-0.400
time
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

CASE STUDY # 4

$$\dot{x} = -2x + 4\sin^2(t) \qquad (1)$$

Objective: Find the periodic solution, i.e. the initial condition (IC), such that $x_0 = x(0) = x(\pi)$.

Solution:

From Maple [9] the solution to (1) is

$$x(t) = \frac{3}{2} - \frac{e^{-2t}}{2} + x(0)e^{-2t} - \cos^2 t - \sin t\bullet\cos t \qquad (2)$$

which is the solution for arbitrary x_0. At the initial condition, $x(0) = x_0$, solve (2) for x_0

$$x(0) = x_0 = \frac{3}{2} - \frac{e^{-2t}}{2} + x_0 e^{-2t} - \cos^2 t - \sin t\bullet\cos t$$

For t = T (period) = π

$$x_0 = \frac{3}{2} - \frac{e^{-2(\pi)}}{2} + x_0 e^{-2(\pi)} - \cos^2(\pi) - \sin(\pi)\bullet\cos(\pi)$$

$$x_0 - x_0 e^{-2(\pi)} = \frac{3}{2} - \frac{e^{-2(\pi)}}{2} - \cos^2(\pi) - \sin(\pi)\bullet\cos(\pi)$$

$$x_0\left(1 - e^{-2\pi}\right) = \frac{3}{2} - \frac{e^{-2\pi}}{2} - (-1)^2 - (0)\bullet(-1)$$

$$x_0\left(1 - e^{-2\pi}\right) = \frac{3}{2} - \frac{e^{-2\pi}}{2} - 1 = \frac{1}{2} - \frac{e^{-2\pi}}{2} = \frac{1}{2}\left(1 - e^{-2\pi}\right)$$

$$x_0 = \frac{\frac{1}{2}\left(1 - e^{-2\pi}\right)}{\left(1 - e^{-2\pi}\right)} = \frac{1}{2}$$

Using the Poincare' map , $x_1 = x_0$, $x_2 = \mathbf{P}(x_1)$
substitute the value for x_1 into equation (2)

$$x_2 = \frac{3}{2} - \frac{e^{-2t}}{2} + (\tfrac{1}{2})e^{-2t} - \cos^2 t - \sin t \bullet \cos t$$

After another period, $t = T = \pi$

$$x_2 = \frac{3}{2} - \frac{e^{-2\pi}}{2} + (\tfrac{1}{2})e^{-2\pi} - \cos^2(\pi) - \sin(\pi)\bullet\cos(\pi)$$

$$x_2 = \frac{3}{2} - \frac{e^{-2\pi}}{2} + (\tfrac{1}{2})e^{-2\pi} - (-1)^2 - (0)\bullet(-1)$$

$$x_2 = \frac{3}{2} - \frac{e^{-2\pi}}{2} + (\tfrac{1}{2})e^{-2\pi} - 1$$

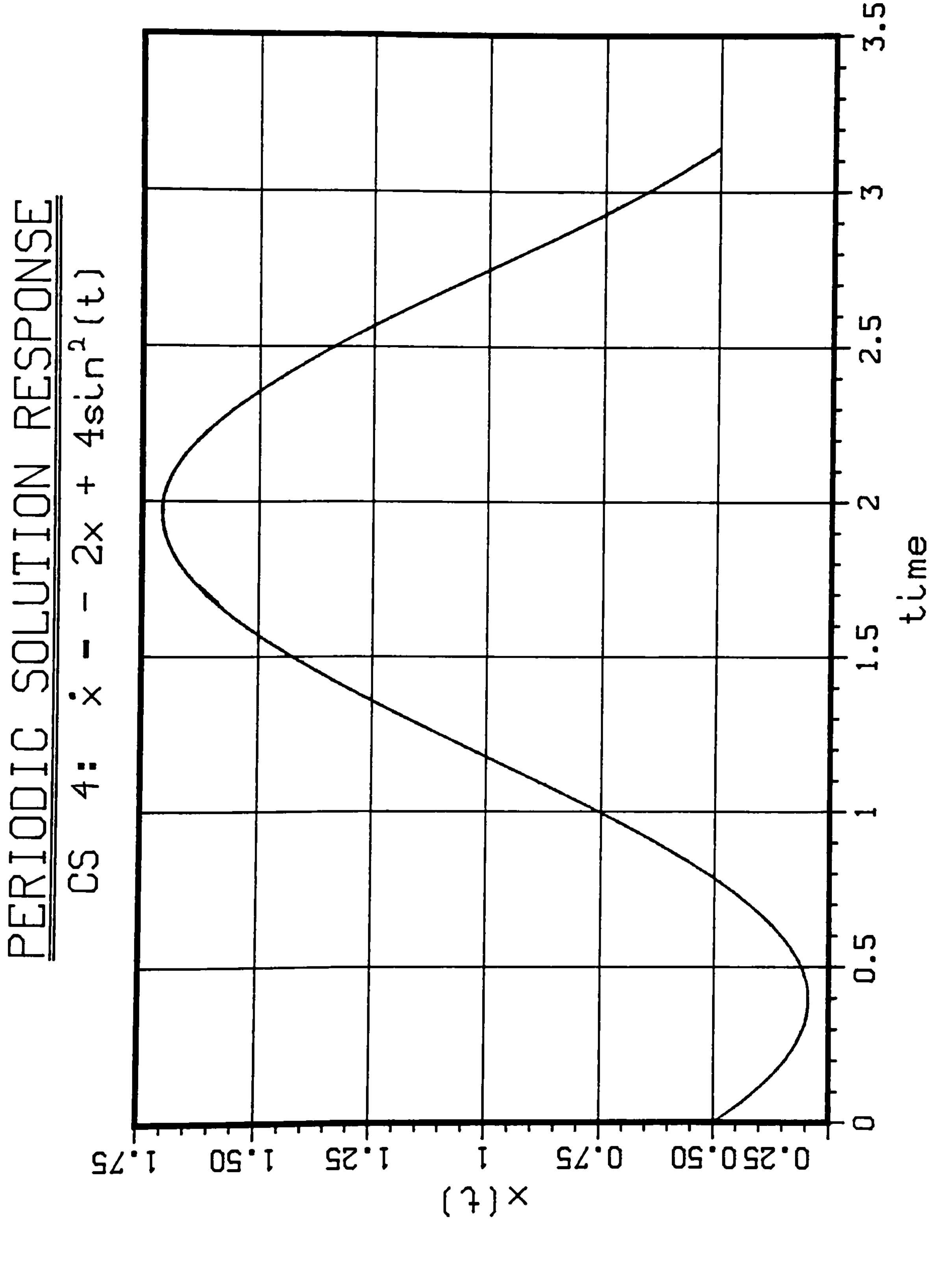
$$x_2 = \frac{1}{2}$$

Since

$$x_0 = x_1 = x_2 \ldots = \frac{1}{2}$$

we can conclude that we have found a <u>Periodic solution</u> which is also a <u>fixed point</u>

$$\Rightarrow \quad x_0 = x_1 = x_2 \ldots = \frac{1}{2} \quad \leftarrow \underline{\text{fixed point}}$$


PERIODIC SOLUTION RESPONSE
CS 4: ẋ = − 2x + 4sin²(t)
x(t)
0.25
0.50
0.75
1
1.25
1.50
1.75
time
0
0.5
1
1.5
2
2.5
3
3.5

Higher Dimensional Systems

Now looking at the equation for a 2-D linear system (mass spring),

$$m\ddot{x} + kx = f(t) \tag{3.45}$$

where f(t) is periodic, $f(t + T) = f(t)$

$T \equiv$ forcing period

For a 2-D system, the approach for solving is similar to that for a 1-D system. The difference is that now there is a system of equations instead of one equation. There are two methods for solving systems of linear 1^{st} order equations:

[1] Fundamental solutions

The 2^{nd} order linear system for a mass spring setup is

$$m\ddot{x} + kx = \sin(t) \tag{3.46}$$

Convert the equation to two linear 1^{st} order state equations by using,

$$y = \dot{x} \tag{3.47}$$

Equation (3.45) then becomes

$$m\dot{y} + kx = \sin(t) \tag{3.48}$$

So the system of 1^{st} order equations are,

$$\left\{ \begin{array}{l} y = \dot{x} \\ \\ \ddot{y} = \frac{\sin(t)}{m} - \frac{kx}{m} \end{array} \right\} \qquad (3.49)$$

[2] Use Laplace Transform.

Remember that the Poincare' map assigns state values at t=0 to state values at t = T (end of period).

$$\{x(0)\} \rightarrow \left\{\mathbb{P}\big(x(0)\big)\right\} = \{x(T)\} \qquad (3.50)$$

The symbolic computational program Maple [9] is used to solve the system of equation for the responses in equation (3.49). For the system with m = 1 and k = 2, the solution for the system equation from Maple is,

$$\begin{cases} x(t) = \sin(t) + y(0)\dfrac{\sin\sqrt{2}\,t}{\sqrt{2}} - \dfrac{\sin\sqrt{2}\,t}{\sqrt{2}} + x(0)\cos\sqrt{2}\,t \\ y(t) = \cos(t) - 2x(0)\dfrac{\sin\sqrt{2}\,t}{\sqrt{2}} + y(0)\cos\sqrt{2}\,t - \cos\sqrt{2}\,t \end{cases} \qquad (3.51)$$

Substituting t = T

$$\begin{cases} x(T) = \sin(T) + y(0)\dfrac{\sin\sqrt{2}\,T}{\sqrt{2}} - \dfrac{\sin\sqrt{2}\,T}{\sqrt{2}} + x(0)\cos\sqrt{2}\,T \\ y(T) = \cos(T) - 2x(0)\dfrac{\sin\sqrt{2}\,T}{\sqrt{2}} + y(0)\cos\sqrt{2}\,T - \cos\sqrt{2}\,T \end{cases} \tag{3.52}$$

This equation will be used to construct the Poincare' map. Given x(0) and y(0), we now have an explicit expression for the set {x(T), y(T)}. So letting the initial condition be <u>arbitrary</u>, that is, just let x(0) be x and y(0) be y, we find that after one period the point {x,y} is mapped to

$$\left| \begin{matrix} \sin(T) + y(0)\dfrac{\sin\sqrt{2}\,T}{\sqrt{2}} - \dfrac{\sin\sqrt{2}\,T}{\sqrt{2}} + x(0)\cos\sqrt{2}\,T \\ \cos(T) - 2x(0)\dfrac{\sin\sqrt{2}\,T}{\sqrt{2}} + y(0)\cos\sqrt{2}\,T - \cos\sqrt{2}\,T \end{matrix} \right| \tag{3.53}$$

Thus the associated Poincare' map is given by

$$\mathbb{P}\{\vec{x}\} = \begin{Bmatrix} x(T) \\ y(T) \end{Bmatrix} \tag{3.54}$$

This mapping holds for any forcing period T. So to find the periodic solution to the problem at hand, it means that we are looking for a fixed point of the Poincare' map $\mathbb{P}\{\vec{x}\}$. That is,

$$\begin{Bmatrix} x \\ y \end{Bmatrix}^{*} = \mathbb{P}\{\vec{x}^{*}\} \tag{3.55}$$

Thus, if $\{\vec{x}\}^* = \mathbf{P}\{\vec{x}^*\}$ i.e. fixed point, then $\{x(0),y(0)\} = \{\vec{x}\}^*$ is the initial condition that gives rise to a periodic solution to the set of linear 1st order equations.

Back to equation (3.53) with $T = 2\pi$ $x(0) = x_0$, $y(0) = y_0$, you end up with the equation

$$
\begin{aligned}
x_0 &= -\ 0.36295 + 0.36295y_0 - 0.85822x_0 \\
y_0 &= \ \ 1.85822 - 0.72590x_0 - 0.85822y_0
\end{aligned}
\tag{3.56}
$$

Solving equation (II.12) simultaneously, yields

$$
\begin{aligned}
x_0 &= 0 \\
y_0 &= 1.0
\end{aligned}
\tag{3.57}
$$

The above solution was plotted to verify that it gives a periodic solution. These plots can be seen in Case Study # 5.

The following example problems are solved using this technique. Detail of each problem along with plots of the responses verifying periodic solutions are also included.

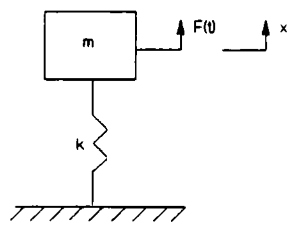
1. $\ddot{x} + 2x = \sin(t)$ — Case Study # 5

2. $\ddot{x} + 2\dot{x} + 2x = \sin(t)$ — Case Study # 6

3. $\left| \begin{aligned} m_1\ddot{x} + k_1x_1 + k_2(x_1-x_2) &= \sin(t) \\ m_2\ddot{x}_2 + k_2(x_2-x_1) &= 0 \end{aligned} \right|$ — Case Study # 7

4. $\left| \begin{aligned} \dot{x} &= x + y + 20\cos(\pi t) \\ \dot{y} &= \frac{x}{2} - y + 10\sin(2\pi t) \end{aligned} \right|$ — Case Study # 8

CASE STUDY # 5



Mass-spring system

EQM:

$$\ddot{x} = -2x + \sin(t) \qquad (\tau = 2\pi) \qquad (1)$$

<u>Objective</u>: Find periodic solution for the above system, that is the IC x_0 that will repeat itself after period τ.

<u>Solution</u>: Let $y = \dot{x}$, then equation (1) can be written as the system,

$$\left\{ \begin{array}{l} y = \dot{x} \\ \dot{y} = \sin(t) - 2x \end{array} \right\} \qquad (2)$$

From Maple, the program MSPR was written to solve the system of linear differential equations (2). The solution is,

$$\left\{ \begin{array}{l} x(t) = \sin(t) + \dfrac{y(0)\sin\sqrt{2}t}{2} - \dfrac{\sin\sqrt{2}t}{2} + x(0)\cos\sqrt{2}t \\ y(t) = \cos(t) - \dfrac{2x(0)\sin\sqrt{2}t}{2} + y(0)\cos\sqrt{2}t - \cos\sqrt{2} \end{array} \right\} \qquad (3)$$

Solving for $x(0)$ & $y(0)$ with $\tau = t = 2\pi$ from equation (3)

$$\left\{ \begin{array}{l} x(0) = -0.36295 + 0.36295y(0) - 0.85822x(0) \\ y(0) = 1.9=8522 - 0.72590x(0) - 0.85822y(0) \end{array} \right\} \quad (4)$$

simplifying equation (4)

$$\left\{ \begin{array}{l} 1.85822x(0) = -0.36295 + 0.36295y(0) \\ 1.85822y(0) = 1.8522 - 0.72590x(0) \end{array} \right\} \quad (5)$$

$$x(0) = -0.19532 + 0.19532\, y(0) \quad (5a)$$

substituting $x(0)$ into 2^{nd} equation in equation (5) yields

$$1.85822y(0) = 1.8522 - 0.72590[-0.19532 + 0.19532\, y(0)] \quad (6)$$

$$1.85822y(0) = 1.8522 + 0.14178 - 0.14178y(0) \quad (7)$$

$$1.9700y(0) = 2.0000$$

$$y(0) = 1.0$$

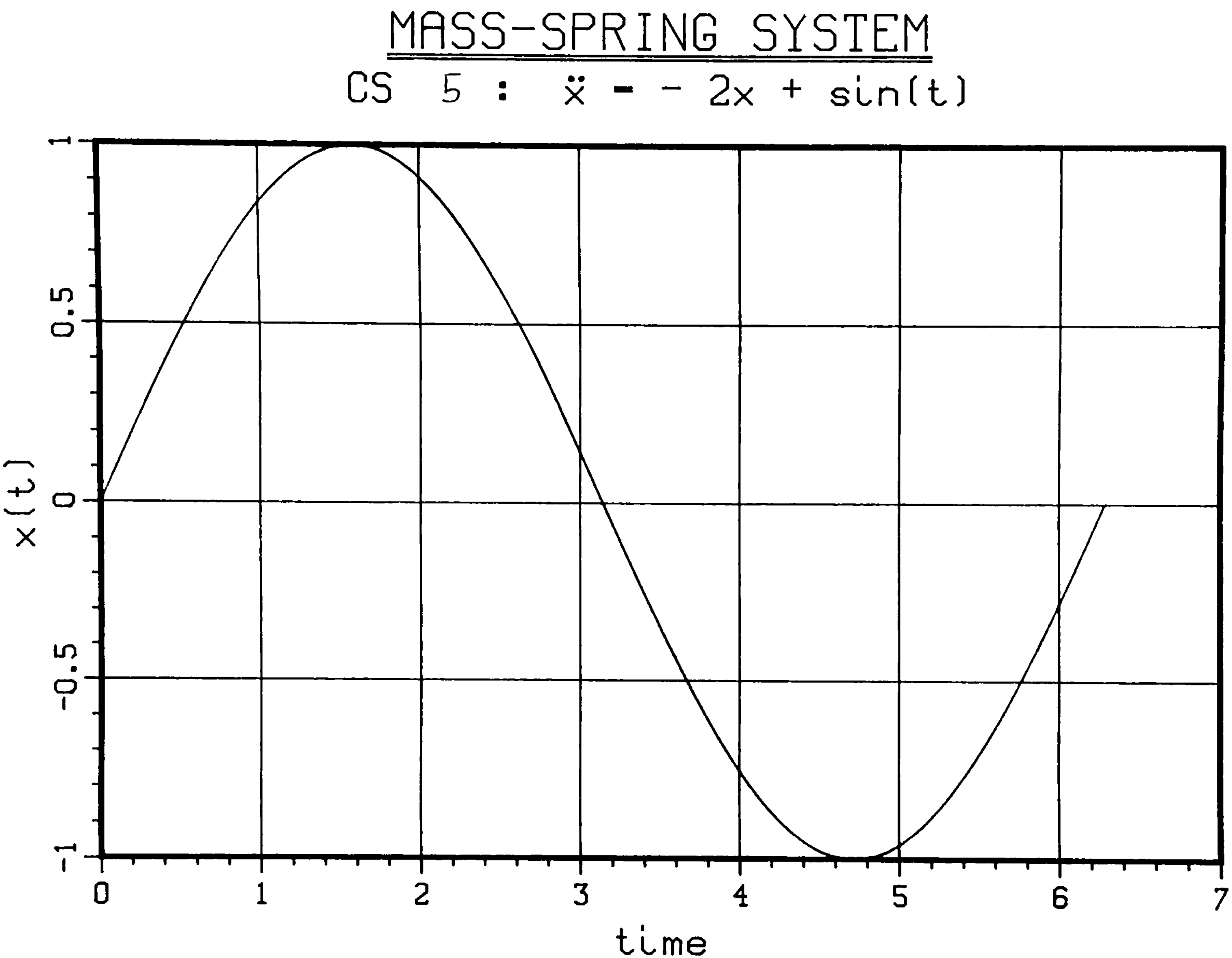
From equation (5a)

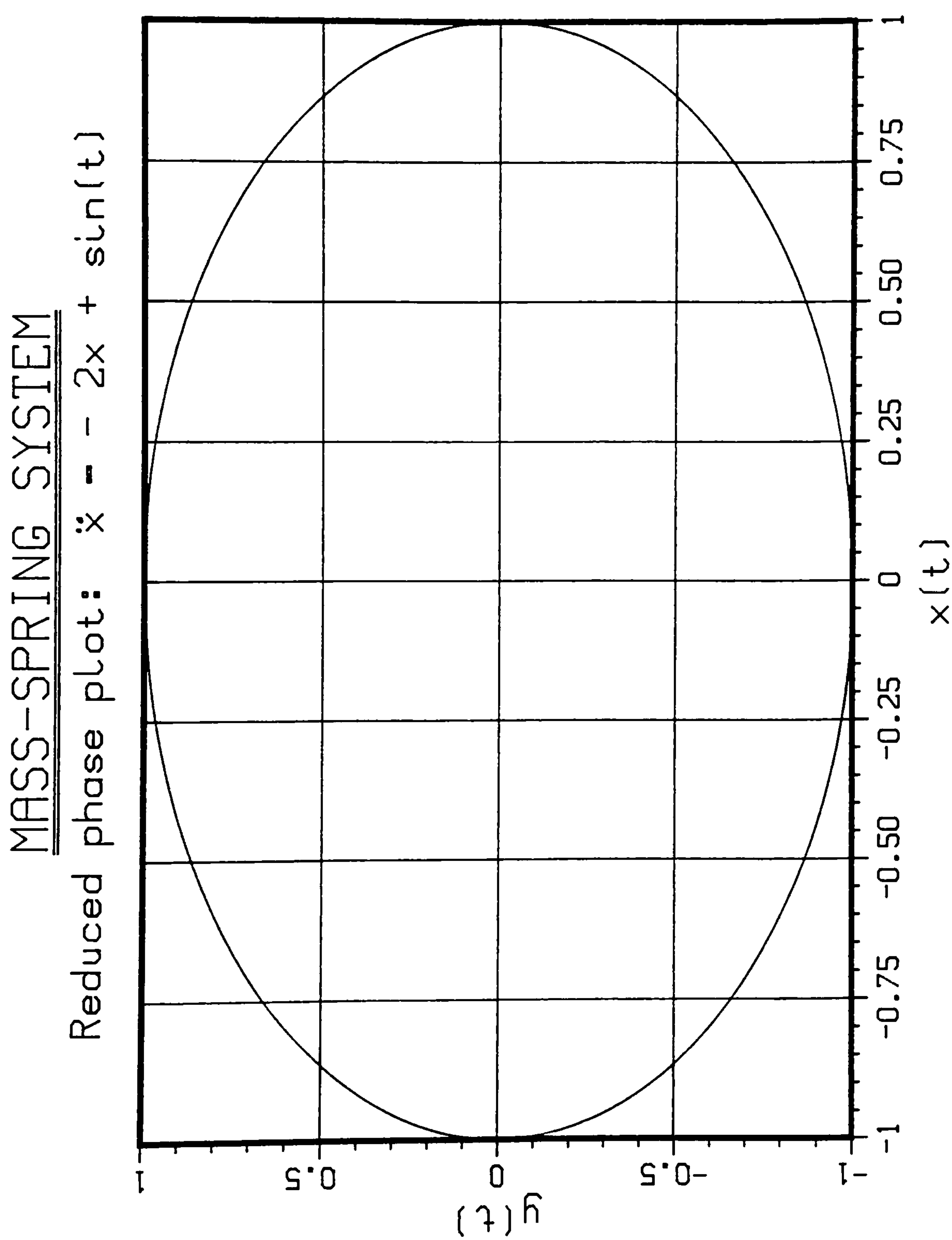
$$x(0) = 0.0$$

The initial condition giving rise to a periodic solution is

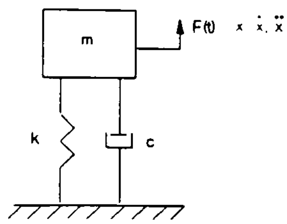
$$x(0) = 0.0$$
$$y(0) = 1.0$$

The following plots verify that the above values found give rise to a periodic solution.


MASS-SPRING SYSTEM
CS 5 : ẍ = - 2x + sin(t)
x(t)
1
0.5
0
-0.5
-1
0
1
2
3
4
5
6
7
time


MASS-SPRING SYSTEM
Reduced phase plot: ẍ = − 2x + sin(t)
x(t)
y(t)
-1
-0.75
-0.50
-0.25
0
0.25
0.50
0.75
1
-1
-0.5
0
0.5
1

CASE STUDY # 6



Mass-spring-damper system

EQM:

$$\ddot{x} = -2\dot{x} - 2x + \sin(t) \quad (\tau = 2\pi) \tag{1}$$

<u>Objective</u>: Find periodic solution for the above system, that is the initial condition x_0 that will repeat itself after period τ.

<u>Solution</u>: Let $y = \dot{x}$, then equation (1) can be written as the system,

$$\left\{ \begin{array}{l} y = \dot{x} \\ \dot{y} = \sin(t) - 2\dot{x} - 2x \end{array} \right\} \tag{2}$$

From Maple, the program MSPRD was written to solve the system of linear differential equations (2). The solution is,

$$\left\{ \begin{array}{l} x(t) = y(0)e^{-t}\sin(t) + \frac{1}{5} e^{-t}\sin(t) + x(0)e^{-t} + \frac{2}{5}e^{-t}\cos(t) \\ \qquad + x(0)e^{-t}\cos(t) + \frac{1}{5}\sin(t) - \frac{2}{5}\cos(t) \end{array} \right\} \tag{3}$$

$$\left\{ y(t) = -2x(0)e^{-t}\sin(t) - \tfrac{3}{5}e^{-t}\sin(t) - \tfrac{1}{5}e^{-t}\cos(t) \right.$$

$$\left. - y(0)e^{-t}\sin(t) + y(0)e^{-t}\cos(t) + \tfrac{2}{5}\cos(t) + \tfrac{1}{5}\sin(t) \right\} \qquad (4)$$

Now solve for $x(0)$ with $t = \tau = 2\pi$ from equation (3)

$$\left\{ x(0) = x(\tau) = y(0)e^{-\tau}\sin(\tau) + \tfrac{1}{5}\, e^{-\tau}\sin(\tau) + x(0)e^{-\tau} \right.$$

$$\left. + \tfrac{2}{5}e^{-\tau}\cos(\tau) + x(0)e^{-\tau}\cos(\tau) + \tfrac{1}{5}\sin(\tau) - \tfrac{2}{5}\cos(\tau) \right\}$$

simplifying [dropping $\sin(\tau = 2\pi) = 0$],

$$x(0) = \tfrac{2}{5}e^{-\tau}\cos(\tau) + x(0)e^{-\tau}\cos(\tau) - \tfrac{2}{5}\cos(\tau)$$

$$x(0) = \tfrac{2}{5}[1.8674(10^{-3})] + x(0)[1.8674(10^{-3})] - \tfrac{2}{5}$$

$$x(0)\ [1 - 1.8674(10^{-3})] = \tfrac{2}{5}[1.8674(10^{-3})] - \tfrac{2}{5}$$

$$0.998x(0) = -0.39925$$

$$x(0) = -0.39997$$

Similarly, solve equation (5) for $y(0)$ with $t = \tau = 2\pi$,

$$y(0) = y(\tau) = -2x(0)e^{-\tau}\sin(\tau) - \tfrac{3}{5}e^{-\tau}\sin(\tau) - \tfrac{1}{5}e^{-\tau}\cos(\tau)$$

$$- y(0)e^{-\tau}\sin(\tau) + y(0)e^{-\tau}\cos(\tau) + \tfrac{1}{5}\cos(\tau) + \tfrac{2}{5}\sin(\tau)$$

Simplifying [dropping $\sin(\tau = 2\pi) = 0$],

$$y(0) = -\tfrac{1}{5}e^{-\tau}\cos(\tau) + y(0)e^{-\tau}\cos(\tau) + \tfrac{1}{5}\cos(\tau)$$

$$y(0) = -\tfrac{1}{5}[1.8674(10^{-3})] + y(0)[1.8674(10^{-3})] + \tfrac{1}{5}$$

$$y(0)[1 - 1.8674(10^{-3})] = -\tfrac{1}{5}[1.8674(10^{-3})] + \tfrac{1}{5}$$

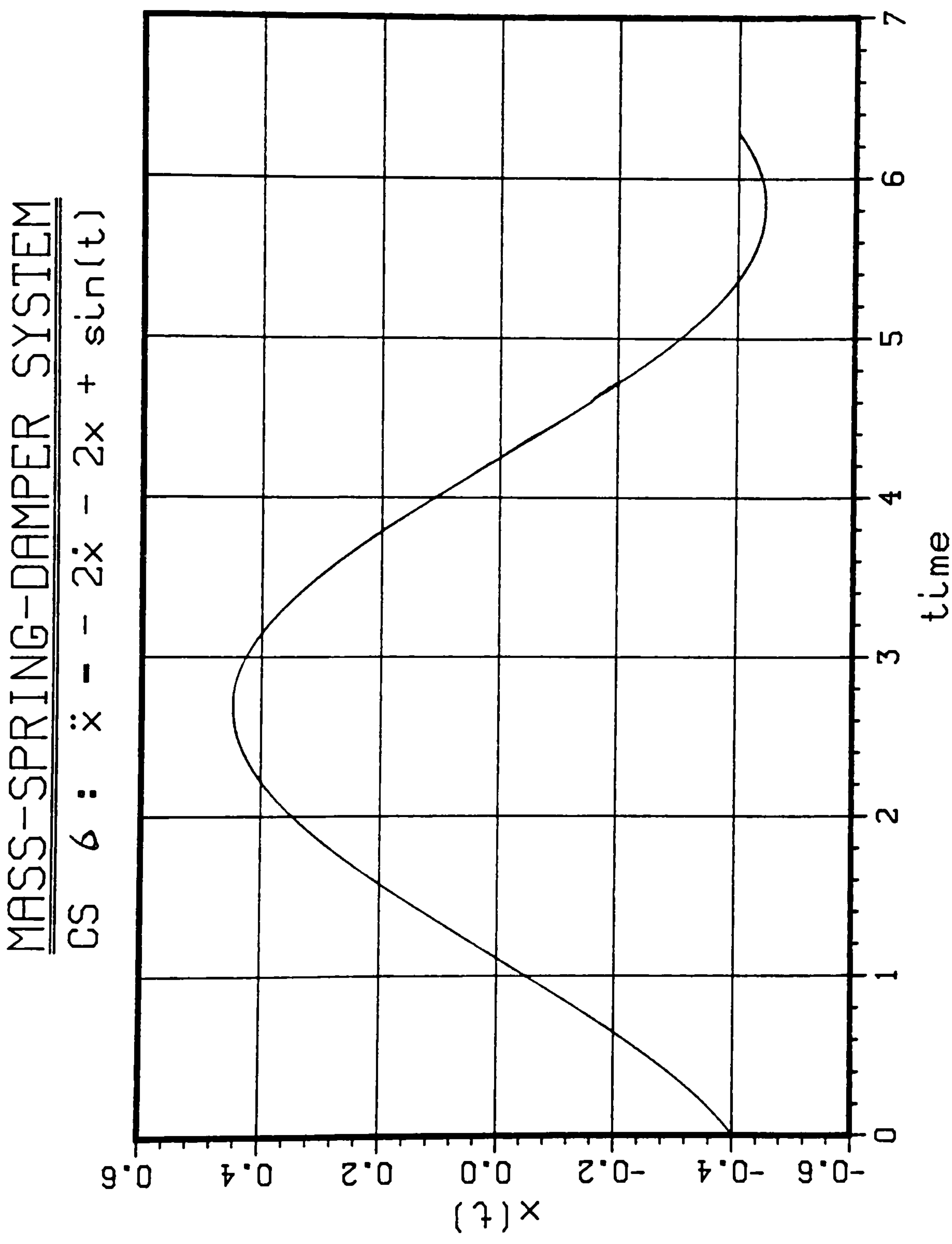
$$y(0)\ [.998] = 0.19963$$

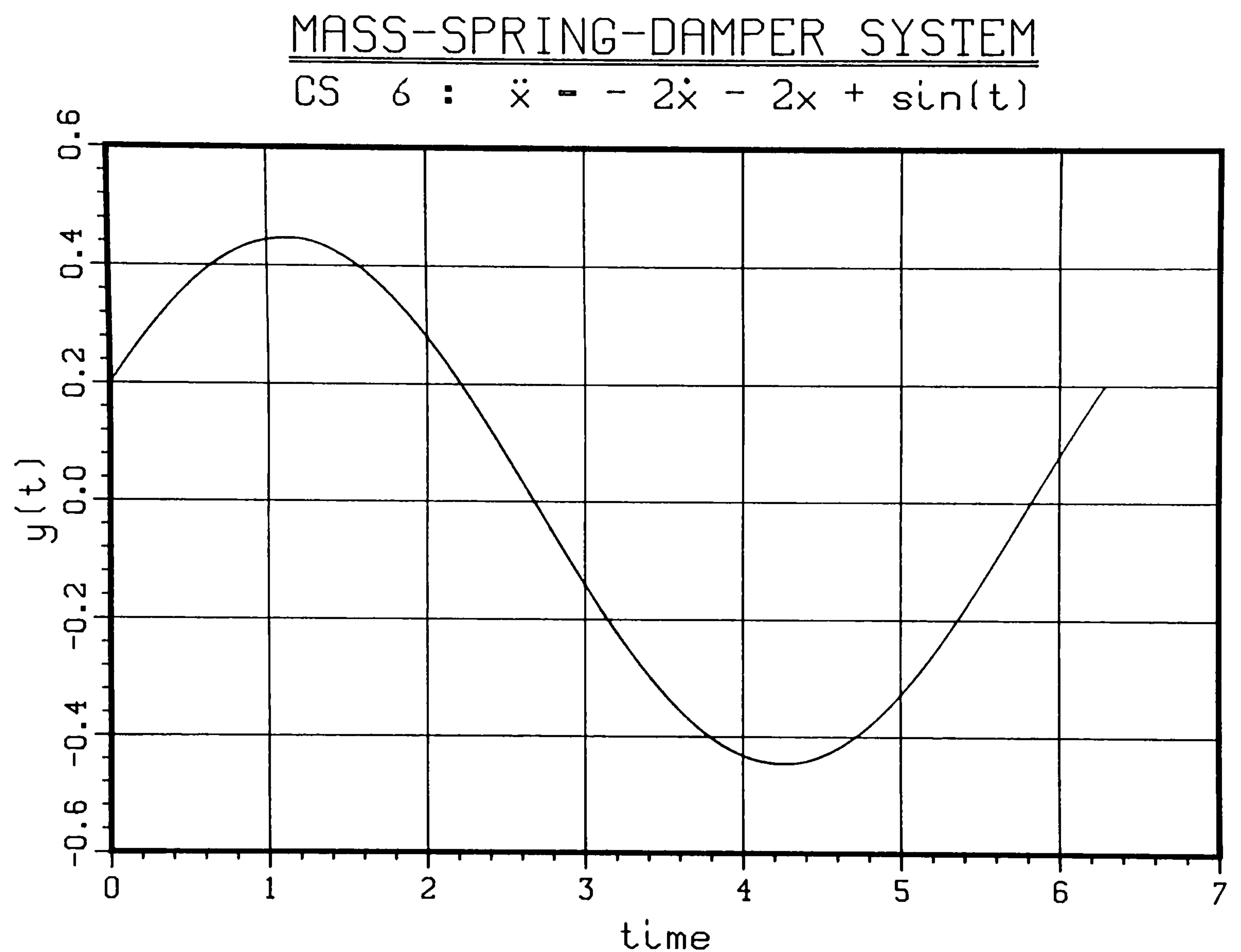
$$y(0) = 0.20$$

The initial condition giving rise to a periodic solution is

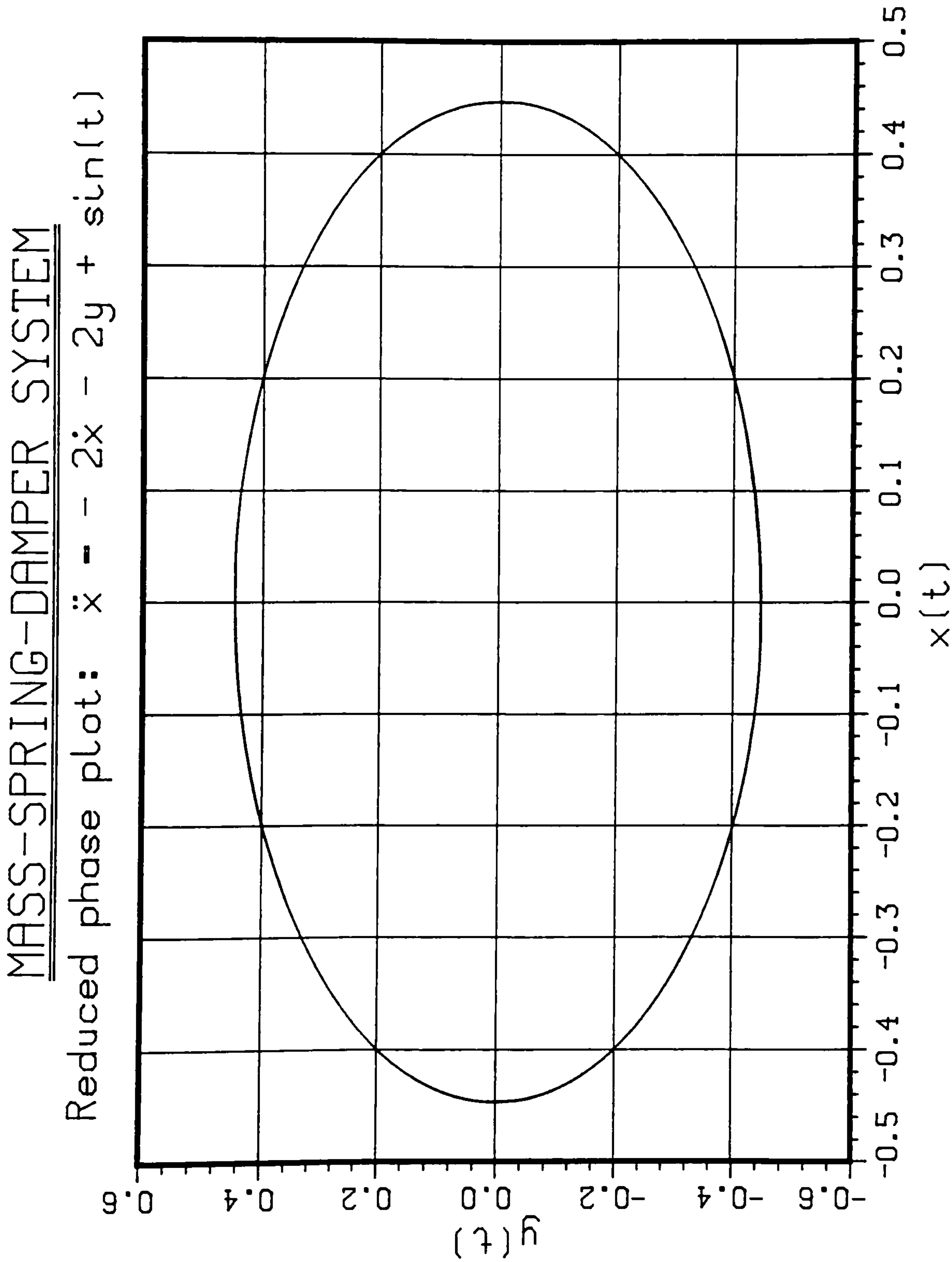
$$x(0) = -0.4$$
$$y(0) = 0.2$$

Plots of the above values verified the periodic solution.

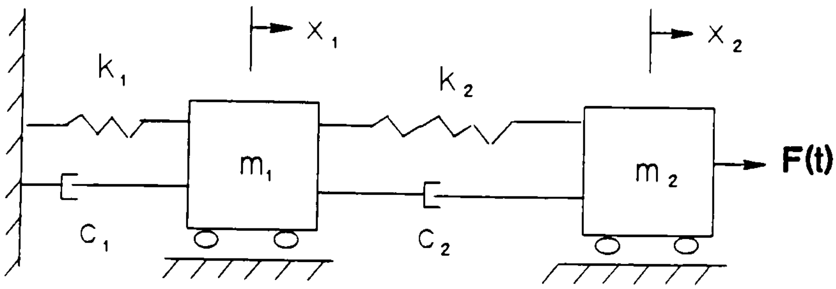

MASS-SPRING-DAMPER SYSTEM
CS 6 : ẍ = - 2ẋ - 2x + sin(t)
x(t)
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
0
1
2
3
4
5
6
7
time



MASS-SPRING-DAMPER SYSTEM
CS 6 : ẍ = - 2ẋ - 2x + sin(t)
y(t)
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
0
1
2
3
4
5
6
7
time


MASS-SPRING-DAMPER SYSTEM
Reduced phase plot: ẍ = - 2ẋ - 2y + sin(t)
x(t)
y(t)
-0.5
-0.4
-0.3
-0.2
-0.1
0.0
0.1
0.2
0.3
0.4
0.5
-0.6
-0.4
-0.2
0.0
0.2
0.4
0.6

CASE STUDY # 7



<u>Objective</u>: Find periodic solution for the above system, that is the Initial Conditions that will repeat itself after period τ.

<u>Solution</u>: From the differential equation of motion (EQM), then arrange the system equations in simultaneous first order form, by letting $\dot{x}_1 = v_1$ & $\dot{x}_2 = v_2$, state equations.

EQM:

$$-k_1x_1 - c_1\dot{x}_1 + k_2(x_2 - x_1) + c_2(\dot{x}_2 - \dot{x}_1) = m_1\ddot{x}_1 \quad (1)$$

$$F(t) - k_2(x_2 - x_1) - c_2(\dot{x}_2 - \dot{x}_1) = m_2\ddot{x}_2 \quad (2)$$

Rearranging,

$$m_1\ddot{x}_1 + (c_1 + c_2)\dot{x}_1 - c_2\dot{x}_2 + (k_1 + k_2)x_1 - k_2x_2 = 0 \quad (1a)$$

$$m_2\ddot{x}_2 - c_2\dot{x}_1 + c_2\dot{x}_2 - k_2x_1 + k_2x_2 = F(t) \quad (2a)$$

By letting,

$$\dot{x}_1 = v_1$$
$$\&$$
$$\dot{x}_2 = v_2$$

equations (1a) & (2a) become,

$$m_1\dot{v}_1 + (c_1 + c_2)v_1 - c_2v_2 + (k_1 + k_2)x_1 - k_2x_2 = 0 \tag{1a}$$

$$m_2\dot{v}_2 - c_2v_1 + c_2v_2 - k_2x_1 + k_2x_2 = F(t) \tag{2a}$$

The state equations are,

$$\left|\begin{aligned} \dot{x}_1 &= v_1 \\ \dot{x}_2 &= v_2 \\ \dot{v}_1 &= -\frac{(c_1 + c_2)}{m_1}v_1 + \frac{c_2}{m_1}v_2 - \frac{(k_1 + k_2)}{m_1}x_1 + \frac{k_2}{m_1}x_2 \\ \dot{v}_2 &= \frac{c_2}{m_2}v_1 - \frac{c_2}{m_2}v_2 + \frac{k_2}{m_2}x_1 - \frac{k_2}{m_2}x_2 + \frac{F(t)}{m_2} \end{aligned}\right| \tag{3}$$

In matrix form the above system equation can be represented by

$$\dot{\vec{x}} = [A]\vec{x} + [B]u(t) \tag{4}$$

For the case with <u>no damping</u>,

$$c_1 = c_2 = 0$$

Using the following values

$$k_2 = k_1 = 1$$
$$m_1 = m_2 = 1$$
$$F(t) = \sin(t)$$

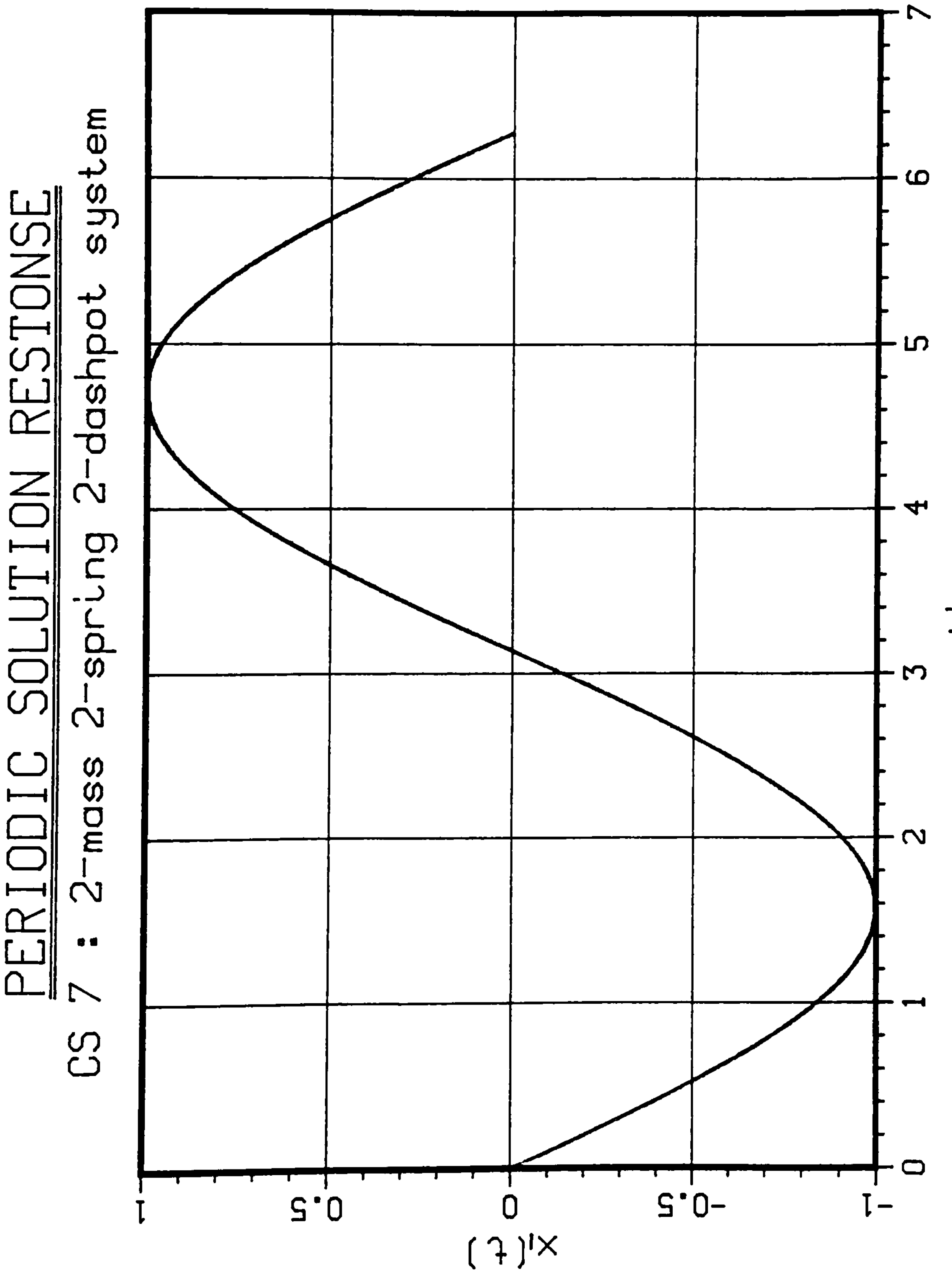
equation (3) become

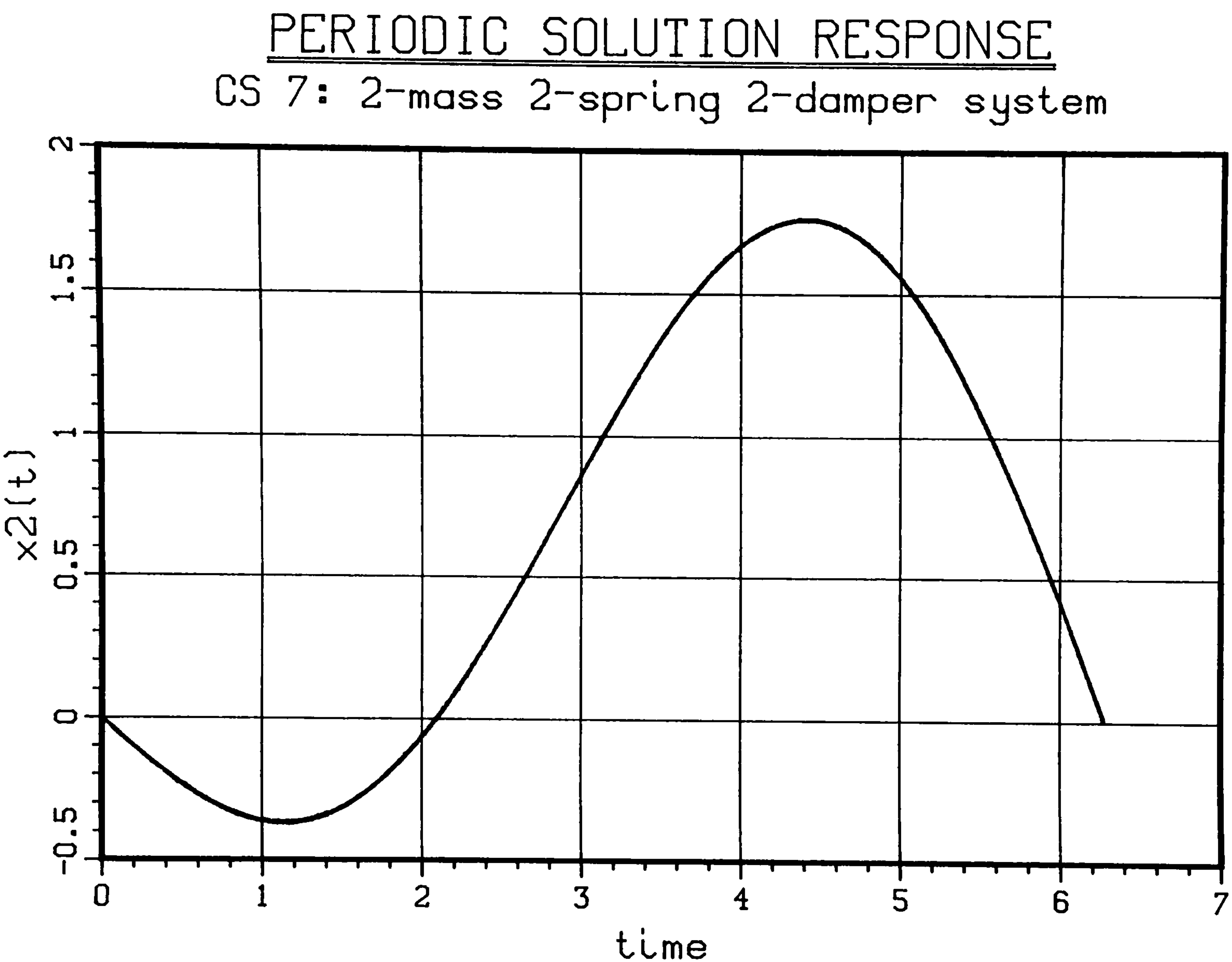
$$\left|\begin{aligned} \dot{x}_1 &= v_1 \\ \dot{x}_2 &= v_2 \\ \dot{v}_1 &= -2x_1 + x_2 \\ \dot{v}_2 &= x_1 - x_2 + \sin(t) \end{aligned}\right| \qquad (5)$$

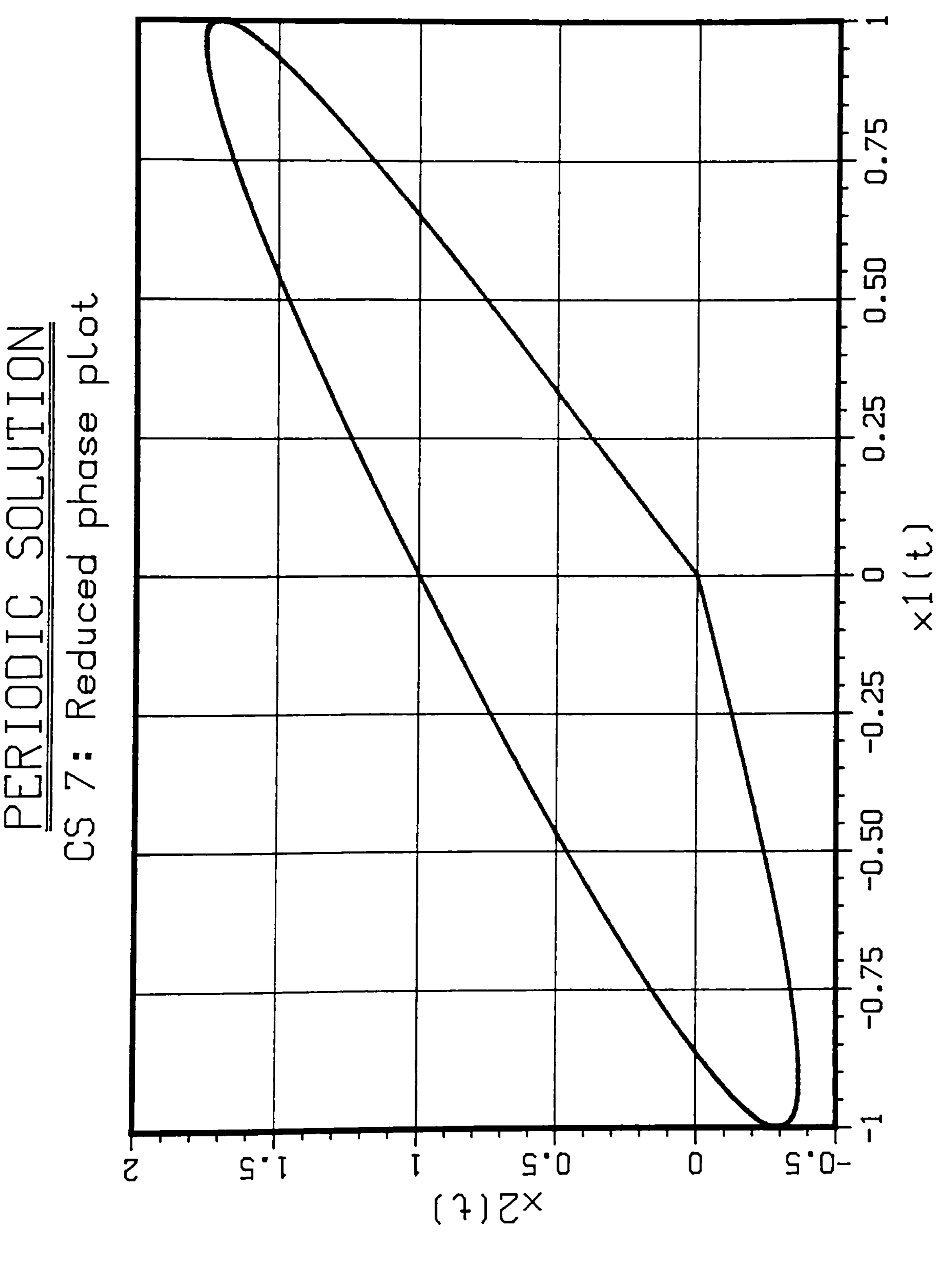
The program, Twomass, (Maple) was used to solve the system of linear first order equations in (5). The initial condition giving rise to a periodic solution for the system is,

$$\left\{ x_1(0) = 0.0 \; ; \; x_2(0) = 0.0 \; ; \; v_1(0) = -1.0 \; ; \; v_2(0) = -1.0 \right\}$$

The Assystant phase plotting software, was used to verify the periodic solution. Example plots are on the following pages.


PERIODIC SOLUTION RESTONSE
CS 7 : 2-mass 2-spring 2-dashpot system
time
x1(t)
0
1
2
3
4
5
6
7
1
0.5
0
-0.5
-1


PERIODIC SOLUTION RESPONSE
CS 7: 2-mass 2-spring 2-damper system
x2(t)
-0.5
0
0.5
1
1.5
2
time
0
1
2
3
4
5
6
7


PERIODIC SOLUTION
CS 7: Reduced phase plot
x1(t)
x2(t)
-1
-0.75
-0.50
-0.25
0
0.25
0.50
0.75
1
-0.5
0
0.5
1
1.5
2

CASE STUDY # 8

Given: The linear differential equation set.

$$\left|\begin{array}{l} \dot{x} = x + y + 20\cos(\pi t) \\ \dot{y} = \frac{x}{2} - y + 10\sin(2\pi t) \end{array}\right| \qquad (1)$$

Objective: Find periodic solution for the above system equations, that is the Initial Conditions that will repeat itself after the period τ.

Solution: From equation (1) above, Maple was used to solve for the solution x(t) and y(t)

$$x(t) = \frac{[\lambda\cosh(\lambda t) + \sinh(\lambda t)]x_0 + \sinh(\lambda t)y_o}{\lambda} +$$

$$\frac{1}{30\pi^2 + 16\pi^4 + 9}[320\pi^2 + 120\cosh(\lambda t) + \frac{1}{\lambda}(80\pi^3 + 480\pi^2$$

$$+ 180 + 120\pi)\sinh(\lambda t) + (-40\pi^2 - 60)\sin(2\pi t)]$$

$$y(t) = \frac{(\lambda^2 - 1)\sinh(\lambda t)x_0 + [\lambda\cosh(\lambda t) - \sinh(\lambda t)]y_0}{\lambda} +$$

$$\frac{1}{60\pi^2 + 32\pi^4 + 18}[(120 + 320\pi^2 + 240\pi)\cosh(\lambda t) +$$

$$\frac{(-160\pi^3 - 240\pi)}{\lambda}\sinh(\lambda t)] - \frac{40\pi\cos(2\pi t)}{3 + 8\pi^2} +$$

$$\frac{20\sin(2\pi t)}{3 + 8\pi^2} - \frac{20\cos(\pi t)}{2\pi^2 + 3}$$

where $\lambda = \sqrt{\frac{3}{2}}$

After substitution for the value of λ (1.224745), the above expressions for x(t) & y(t) simplifies to

$$\begin{aligned} x(t) = {} & 3.4063\sinh(1.224745t) + 1.7591\cosh(1.224745t) \\ & - 1.7591\cos(\pi t) + x_0\cosh(1.224745t) - 0.2440\sin(2\pi t) \\ & + 5.5263\sin(\pi t) + 0.8165x_0\sinh(1.224745t) \\ & + 0.8165\sinh(1.224745t)y_0 \end{aligned}$$

$$\begin{aligned} y(t) = {} & 2.4128\cosh(1.224547t) - 1.2519\sinh(1.224745t) \\ & + 0.4082x_0\sinh(1.224745t) + y_0\cosh(1.224745t) \\ & - 0.8165\sinh(1.224745t)y_0 - 0.8795\cos(\pi t) \\ & + 0.2440\sin(2\pi t) - 1.5333\cos(\pi t) \end{aligned}$$

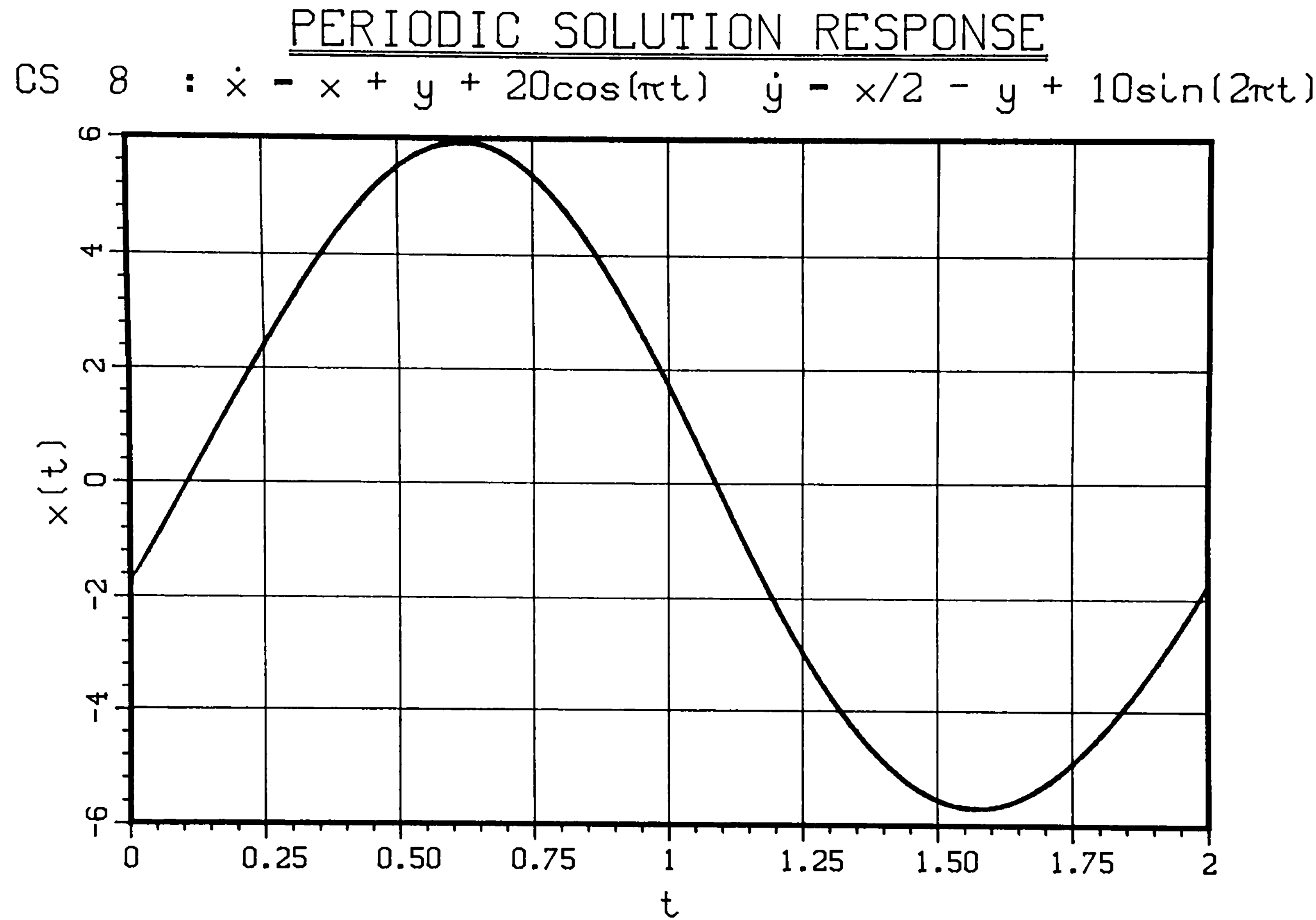
For this problem, with two separate forcing functions, the period is T = 2. Upon substitution for t = T in the above expression, and with $x(2) = x_0$ and $y(2) = y_0$, the requirement for a periodic solution reduces to

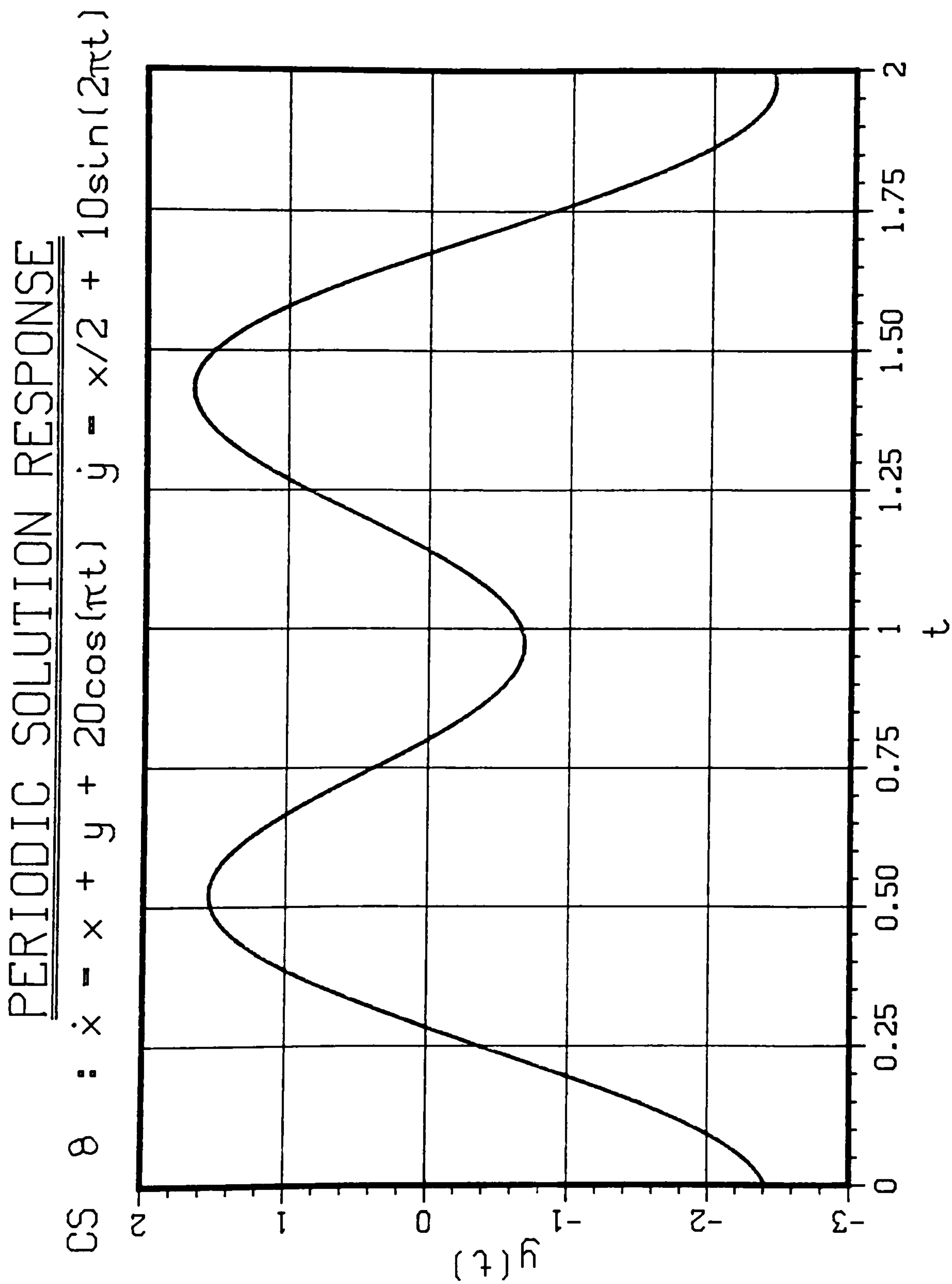
$$\begin{aligned} x_0 &= 28.0834 + 10.5276x_0 + 4.6933y_0 \\ y_0 &= 4.4684 + 2.3466x_0 + 1.1411y_0 \end{aligned}$$

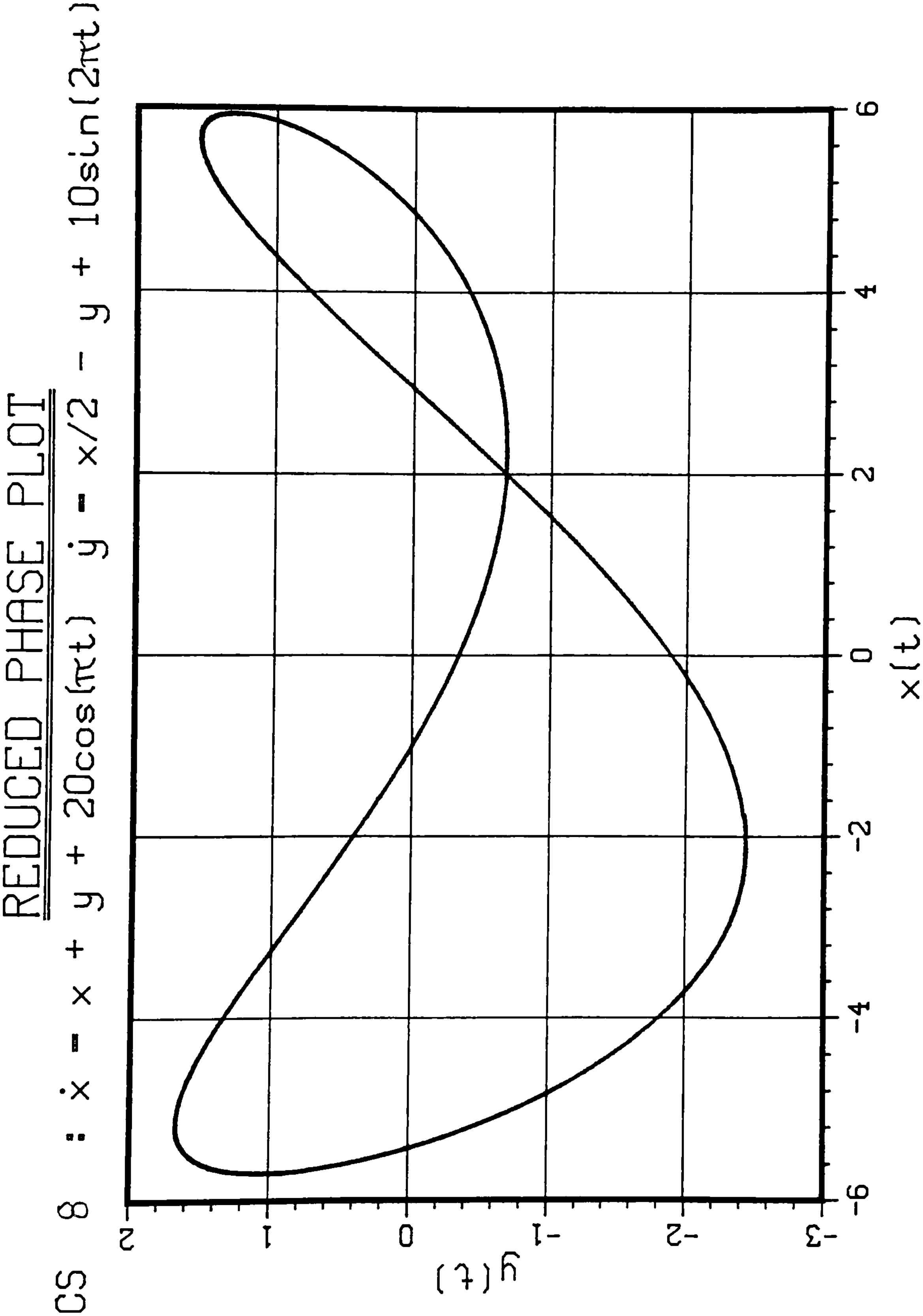
solving simultaneously, the initial condition giving rise to a periodic solution is

$$\left\{ \quad x_0 = -1.7571, \quad y_0 = -2.4128 \quad \right\}$$

Plots used to verify the periodic solution follow.


PERIODIC SOLUTION RESPONSE
CS 8 : ẋ = x + y + 20cos(πt) ẏ = x/2 − y + 10sin(2πt)
x(t)
6
4
2
0
-2
-4
-6
0
0.25
0.50
0.75
1
1.25
1.50
1.75
2
t


PERIODIC SOLUTION RESPONSE
CS 8 : ẋ = x + y + 20cos(πt)   ẏ = x/2 + 10sin(2πt)
2
1
0
-1
-2
-3
y(t)
0
0.25
0.50
0.75
1
1.25
1.50
1.75
2
t


REDUCED PHASE PLOT
CS 8 : ẋ = x + y + 20cos(πt)   ẏ = x/2 - y + 10sin(2πt)
x(t)
y(t)
-6
-4
-2
0
2
4
6
-3
-2
-1
0
1
2

IV NONLINEAR SYSTEMS

The study of nonlinear systems is more complicated than the study of linear systems, which can be attributed to the fact that the superposition principle (the ability to add linearly the responses of a system to various excitations) is not valid for nonlinear systems. This leads to an entirely different approach for handling nonlinear systems. Numerical methods are usually needed for solving nonlinear system equations. It should also be pointed out that the theory of nonlinear differential equations is not as complete as that for linear differential equations. In addition, it relies heavily on approximations based upon linear theory. There are circumstances where it is possible to use methods of linear theory in the study of nonlinear systems by examining the motion in the neighborhood of known motions, a process referred to as linearization. This is the basis of Lyapunov's First Method [2].

There are two basic approaches to solving nonlinear systems, the qualitative and the quantitative method. The qualitative approach is concerned with the general stability characteristics of the system in the area of a known solution, rather than with the explicit time history of the motion. On the other hand, the quantitative approach is also concerned with the time histories. Such solutions can be obtained by pertubation methods or by numerical integration.

This paper will investigate an area of the qualitative approach by focusing in the neighborhood of a known solution of the system, the periodic solution (if it exists). To begin with, we need to introduce the Infinitesimal Generator operator which can be used in the analysis.

INFINITESIMAL GENERATOR

Consider the nonlinear system

$$\dot{x} = F(\dot{x}) \tag{4.0}$$

The Infinitesimal Generator technique is based on the theory of continuous transformation groups and can be used to solve autonomous as well as non-autonomous systems of differential equations. The Infinitesimal Generator can conveniently be used to derive series solutions of nonlinear initial value problems [11]. The Infinitesimal Generator U is a differential operator and is defined by

$$U = F_1 \frac{\partial}{\partial x_1} + F_2 \frac{\partial}{\partial x_2} + \ldots + F_n \frac{\partial}{\partial x_n} \tag{4.1}$$

where F_i are the components of the right hand side of (4.0) and x_i are the state variables. For any arbitrary initial values x_i the solution x(t) can be developed as [11]

$$x^t_i = x_i + tUx_i + \frac{t^2}{2} U^2 x_i + \frac{t^3}{3!} U^3 x_i + \ldots \tag{4.2}$$

where x^t_i is the time advance of the i^{th} component ox x_i. Equation (4.2) is the series solution in time t of an autonomous system, the coefficients of which are functions of the arbitrary initial values x_0, that is

$$x(t) = x_0 + tUx_0 + \frac{t^2}{2} U^2 x_0 + \frac{t^3}{3!} U^3 x_0 + \ldots \tag{4.3}$$

Since U is a differential operator, it is necessary that the operator U^n must be applied to the variable x_i before specific initial values are submitted.

As an example consider the initial value problem

$$\dot{x} = -x^2 + t \qquad x(0) = \bar{x}. \tag{4.4}$$

Since this problem is nonautonomous, make a change of variables y = t, which gives

$$\dot{y} = 1, \qquad y(0) = t_0$$

So the equivalent autonomous system is given by

$$\begin{aligned} \dot{x} &= -x^2 + y & x(0) &= \bar{x} \\ \dot{y} &= 1 & y(0) &= t_0 \end{aligned} \tag{4.5}$$

Infinitesimal Generator Operator:

$$U = F_1 \frac{\partial}{\partial x} + F_2 \frac{\partial}{\partial y} \tag{4.6}$$

$$= (-x^2 + y) \frac{\partial}{\partial x} + 1 \frac{\partial}{\partial y}$$

$$= (-x^2 + y) \frac{\partial}{\partial x} + \frac{\partial}{\partial y}$$

We obtain:

$$Ux = -x^2 + y \tag{4.7}$$

$$U^2x = (-x^2 + y)(-2x) + 1 \tag{4.8}$$

$$U^2x = 2x^3 - 2xy + 1 \tag{4.9}$$

$$U^3x = (-x^2 + y)(6x^2 - 2y) + (-2x)$$

$$= -6x^4 + 8x^2y - 2y^2 - 2x \qquad (4.10)$$

$$U^4x = (-x^2 + y)(-24x^3 + 16xy - 2) + (8x^2 - 4y) \qquad (4.11)$$

And

$$Uy = 1 \qquad (4.12)$$

$$U^ky = 0 \quad \text{for all } k \geq 2 \qquad (4.13)$$

Maple software can be used to solve for the series expansion. The program code for doing the partial derivatives and calculating the expansion for a 2-D system of equation is the program VECFLD2D in Appendix A. All you have to do is define the two variables x_1 and x_2 and the two functions F_1 and F_2 (first order differential equations).

Finally:

with $h \equiv$ time advance

$$x(h) = x^h = x + (-x^2 + y)h + (2x^3 - 2xy + 1)\frac{h^2}{2!} +$$

$$(-6x^4 + 8x^2y - 2y^2 - 2x)\frac{h^3}{3!} + [U^4x]\frac{h^4}{4!} + \ldots \qquad (4.14)$$

$$y^h = y + 1(h) = y + h \qquad (4.15)$$

Now substitute initial conditions:

$$x^h = \bar{x} + (-\bar{x}^2 + t_0)h + (2\bar{x}^3 - 2\bar{x}t_0 + 1)\frac{h^2}{2!} + \ldots.$$

$$+ (-6\bar{x}^4 + 8\bar{x}^2t_0 - 2t_0^2 - 2\bar{x})\frac{h^3}{3!} + [U^4x]\frac{h^4}{4!} + \ldots. \qquad (4.16)$$

where $x = \overline{x}$, $y = t_0$

$$y^h = t_0 + h \tag{4.17}$$

x^h, y^h are values of the state variables after a <u>time-advance</u> of "h".

The series solution can now be expressed as a mapping

$$G(\overline{x},t_0,h) = \overline{x} + (-\overline{x}^2 + t_0)h + (2\overline{x}^3 - 2\overline{x}t_0 + 1)\frac{h^2}{2!} + \ldots.$$

$$+ (-6\overline{x}^4 + 8\overline{x}^2 t_0 - 2t_0^2 - 2\overline{x})\frac{h^3}{3!} + [U^4x]\frac{h^4}{4!} + \ldots. \tag{4.18}$$

with derivative

$$DG(\overline{x},t_0,y) = 1 + (-2\overline{x})h + (6\overline{x}^2 - 2t_0)\frac{h^2}{2!} + \ldots.$$

$$+ (-24\overline{x}^3 + 16\overline{x}t_0 - 2)\frac{h^3}{3!} + D_x[U^4x]\frac{h^4}{4!} + \ldots. \tag{4.19}$$

The fixed point was found to be

$$x_0 = 0.7867627\ldots$$

See Appendix A for the program Test.for that was used to solve for the fixed point of the Poincare' Map. The following figure FIG 4-1, shows the relation between the input function and the convergence of the initial guess to the fixed point.

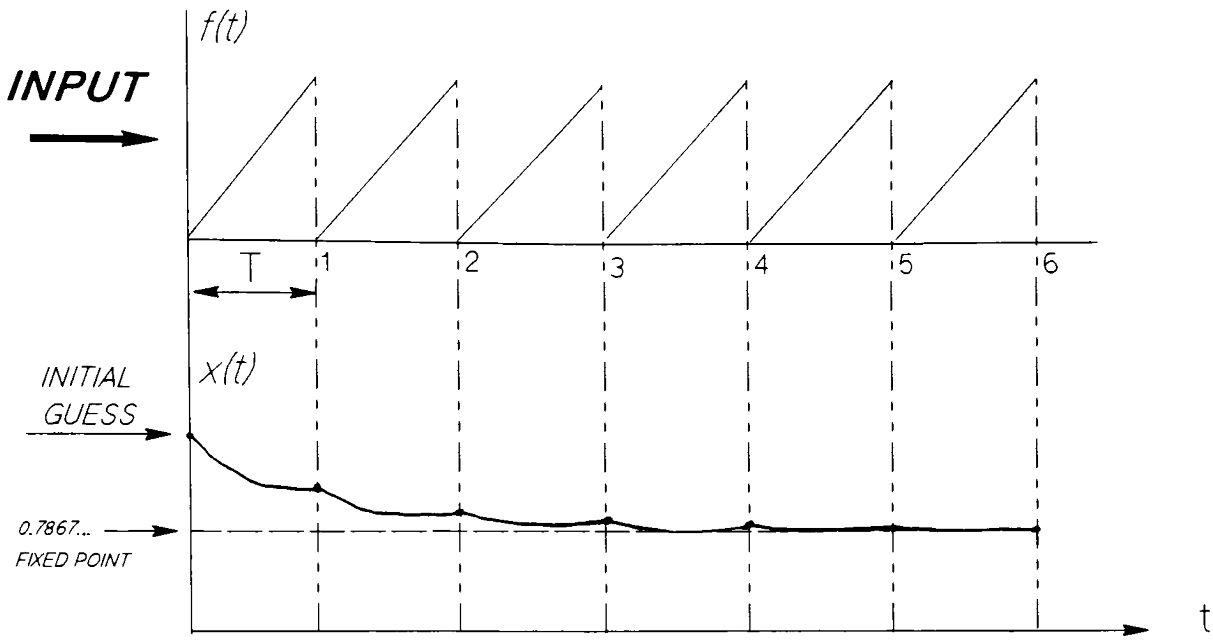


FIG 4-1. System input and output side by side while seeking periodic solution

Example: Solve using Infinitesimal Generator,

$$\dot{x} = -x^2 + tx \qquad x(0) = x_0$$

The initial value was found to be

$$x_0 = 0.5428$$

See example Case Study # 9, at the end of the 1-D Nonlinear Systems section, for detail analysis of this example problem.

POINCARE′ MAP DEVELOPMENT

The Poincare′ Map is developed as follows

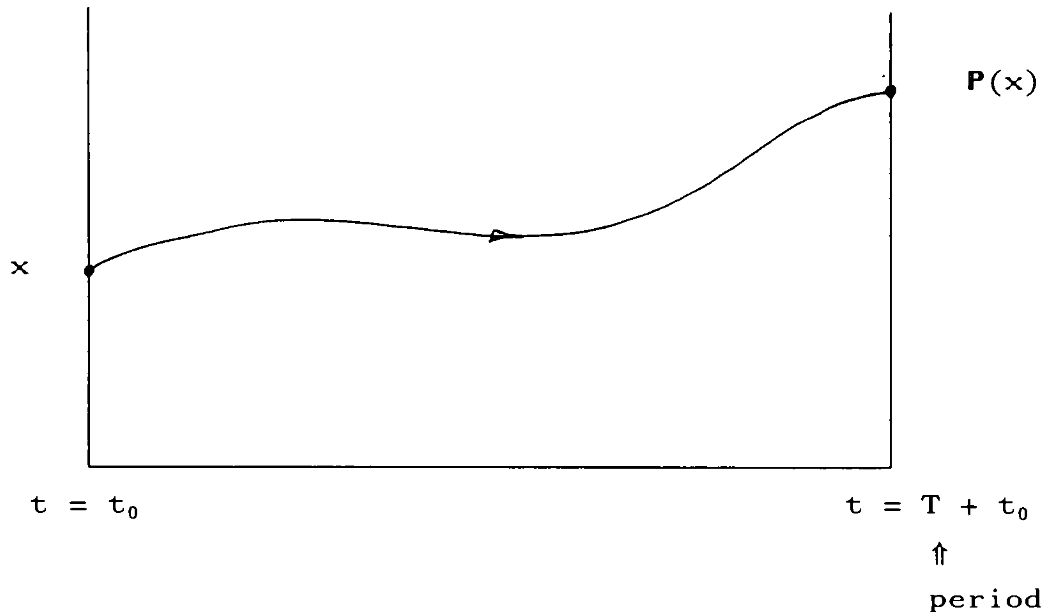


FIG 4-2. Mapping of point x from $t=t_0$ to t=T

Using the Infinitesimal Generator we can generate series solutions to the differential equations. Because of the truncation of higher order terms involved with the series scheme, accurate approximation of the equation solutions will be difficult. So the expressions are only valid on "small" intervals, certainly not out to a large time T.

The method used in dealing with this is to iterate the developed expressions over the subintervals, until we reach the entire period, as shown in the following figure. Here G(.) is the Φ_t (forward advance transformation) for small interval steps. That is, the solution is advanced forward in a sequence of small steps using an explicit series approximation:

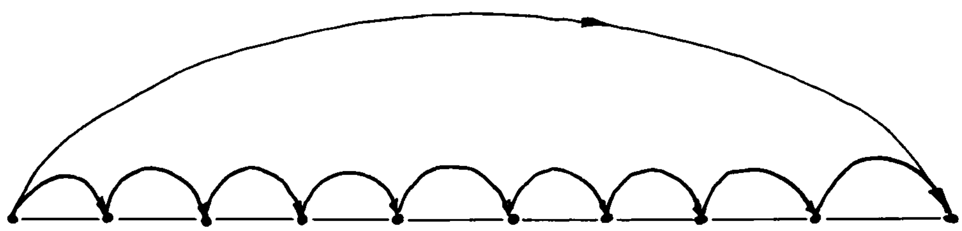


FIG 4-3. Forward mapping of x to the eventual Poincare' Mapping of x

This way we construct the Poincare' Map by iterating our formula through small steps.

Now, to apply Newton Raphson, we need $\mathbb{P}'(x)$, which is the derivative of the Poincare' Map expression. The Newton Raphson method involves generating a sequence $\{x_{k+1}\}$ which converges to a fixed point if $|\mathbb{P}'(x)| < 1$. The iteration scheme [15] is given by

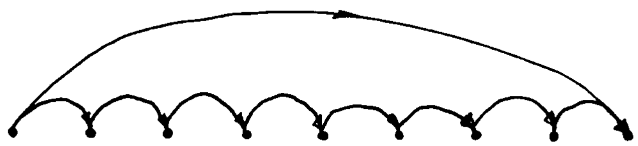
$$x_{k+1} = x_k - \frac{[x_k - \mathbb{P}(x_k)]}{[1 - \mathbb{P}'(x_k)]} \tag{4.20}$$

But $\mathbb{P}(x) = G(G(G(G(\ldots.G(x))\ldots.)))$ (4.21)

So using the Chain Rule,

$$\mathbb{P}'(x) = G'(\underbrace{G(G....(x)..)}_{(N-1)\ \text{iterate}}) * [\underbrace{G(G(G...(x)...)}_{(N-1)\ \text{iterate}}]' \tag{4.22}$$

If we define a sequence of points along the interval as



x_1 x_2 x_3 x_N x_{N+1} $N \equiv$ # of interval

$\Uparrow$

$\mathbb{P}(x_1)$

FIG 4-4. Forward advance mapping and the Poincare' Mapping of point and sequence of points

we get

$$\begin{aligned} x_2 &= G(x_1) \\ x_3 &= G(x_2) \\ &\cdot \\ &\cdot \\ x_{N+1} &= G(x_N) \end{aligned} \tag{4.23}$$

So $\mathbb{P}'(x_1) = G'(x_N)(x_N)'$ (4.24)

$$= G'(x_N)\Big(G(x_{N-1})\Big)'$$

$$= G'\Big(G(x_{N-1})\Big)\Big(G'(x_{N-1})\Big)(x_{N-1})'$$

$$= G'\Big(G(x_{N-1})\Big)[G(G(x_{N-2}))]'(x_{N-1})'$$

$$= G'(x_N)G'(G(x_{N-2}))G'(x_{N-2})\ (x_{N-1})'$$

$$= G'(x_N)G'(x_{N-1})G'(x_{N-2})\ldots G'(x_1)$$

Alternatively,

$$P(x) = G^n(x) \qquad \text{(n iterations)}$$

$$= G(G^{n-1}(x)) \tag{4.25}$$

So $P'(x) = G'(G^{n-1})(G^{n-1}(x))'$ (chain rule) (4.26)

Proceeding inductively,

$$P'(x) = G'(G^{n-1})G'(G^{n-2}(x))\ldots G'(G(x))G'(x) \tag{4.27}$$

So again,

$$P'(x) = G'(x_N)G'(x_{N-1})G'(x_{N-2})\ldots G'(x_2)G'(x_1) \Big|_{x = x_1} \tag{4.28}$$

This is the method we will use to seek periodic solutions for nonlinear systems. Given $\dot{\vec{x}} = \vec{F}(\vec{x},t)$, set up $G(x)$, the forward advance map for "small x intervals", and then calculate $G'(x)$. Apply Newton Raphson to get the best x value for that interval and then increment that x-value forward by the incremental amount. Keep repeating this process until x has been incremented (by the interval amount) over the entire period. The x-value at the end of the period is the fixed point for that system.

Outline of Methodology

Given: A dynamical system, from which we deduce a forward advance approximation $\mathbf{G}(x)$ and then estimate the Poincare' map as $\mathbf{P}(x) = \mathbf{G}^n(x)$.

Objective: To find x_0 such that $\mathbf{P}(x_0) = x_0$, that is establish a fixed point of $\mathbf{P}(x)$.

Now, once we have $\mathbf{P}(x)$ and $\mathbf{P}'(x)$, use Newton-Raphson (N-R) to iteratively find a fixed point:

$$x_{k+1} = x_k - \frac{[x_k - \mathbf{P}(x_k)]}{[1 - \mathbf{P}'(x_k)]} \tag{4.29}$$

Now, how does N-R work? Choose an initial guess x_{guess}, and use equation (4.29) until

$$|x_{k+1} - x_k| < \epsilon$$

where ϵ is some specified accuracy.

Now given x_k, a N-R iterate, how to actually compute x_{k+1} ?

Step 1. Set $x_{guess} = x_k$

Step 2. Determine $\mathbf{P}(x_k)$ and $\mathbf{P}'(x_k)$. That is compute

$$\mathbf{P}(x_k) = \mathbf{G}^n(x_k) = \mathbf{G}(\mathbf{G}(\ldots \mathbf{G}(x_k)\ldots)) \tag{4.30}$$

where $n = \frac{T}{h}$ in which T is the forcing period.

While using equation (4.30), save the intermediate values

$$z_1 = G(x_k)$$
$$z_2 = G^2(x_k) = G(z_1)$$
$$z_3 = G^3(x_k) = G(z_2)$$
$$\vdots$$
$$z_n = G(z_{1-n}) \equiv P(x_k)$$

$$(\text{note } z_{1+i} = G(z_i))$$

so that

$$P'(x_k) = G'(z_1)G'(z_2)G'(z_3)\ldots.G'(z_N) \qquad (4.31)$$

$$\left(z_{N+1} = G(x_n)\right) = x_{N+1}$$

Step 3. Now take P from equation (4.30) and take P' from equation (4.31) and compute

$$x_{k+1} = x_k - \frac{[x_k - P(x_k)]}{[1 - P'(x_k)]}$$

Step 4. Check $|x_{k+1} - x_k| < \epsilon$

- If ϵ is met go to Step 5
- If not, set $x_k = x_{k+1}$ then go to Step 2 and repeat process

Step 5. Output x_k

1-D Nonlinear Systems

For a 1-D nonlinear system, the first step is to rewrite the system equation as an autonomous system equation (if necessary) then apply the infinitesimal generator for the series expansion for the variables. Compute $\mathbf{P}(x)$ and $\mathbf{P}'(x)$ and iterate with Newton-Raphson algorithm to find the fixed point. As an example consider the nonlinear initial value problem

$$\dot{x} + \sin(x) = \sin(2\pi t), \qquad x(0) = x_0 \tag{4.32}$$

Since equation (4.32) is nonautonomous, the problem is rewritten as autonomous (by letting $y = t$)

$$\begin{aligned} \dot{x} &= -\sin(x) + \sin(2\pi y), & x(0) &= x_0 \\ \dot{y} &= 1, & y(0) &= 0 \end{aligned}$$

Recalling the infinitesimal generator operator

$$U = F_1 \frac{\partial}{\partial x_1} + F_2 \frac{\partial}{\partial x_2} + \,.\,. \tag{4.33}$$

which allows the construction of the solution in the series form

$$\phi_i = x_i + t\cdot Ux_i + \frac{t^2}{2!} U^2x_i + \frac{t^3}{3!} U^3x_i + \,.\,. \tag{4.34}$$

with the following identifications

$$\left.\begin{aligned} x_1 &= x \\ x_2 &= y \\ F_1 &= -\sin x_1 + \sin(2\pi)x_2 \\ F_2 &= 1 \end{aligned}\right| \tag{4.35}$$

the infinitesimal generator, from equation (4.33) is

$$U = \left(- \sin x_1 + \sin(2\pi x_2)\right) \frac{\partial}{\partial x_1} + \frac{\partial}{\partial x_2}$$

Operating on the variable x_1, the sequence of coefficients for the x_1 expansion is given by

$$Ux_1 = \left(- \sin x_1 + \sin(2\pi x_2)\right)$$

$$U^2x_1 = \left(\cos x_1 \sin x_1 - \cos x_1 \sin(2\pi x_2) + 2\pi\cos(2\pi x_2)\right)$$

$$U^3x_1 = \Big(\sin^3 x_1 - 2\sin^2 x_1 \sin(2\pi x_2) - \cos^2 x_1 \sin x_1 + \cos^2 x_1 \sin(2\pi x_2) + \sin x_1 \sin^2(2\pi x_2) - 2\pi\cos x_1 \cos(2\pi x_2) - 39.48\sin(2\pi x_2)\Big)$$

$$U^4x_1 = \Big(- 5.0\sin^3 x_1 \cos x_1 + 11\sin^2 x_1 \cos x_1 \sin(2\pi x_2) - 7\sin x_1 \cos x_1 \sin^2(2\pi x_2) + \cos^3(x_1)\sin x_1 - \cos^3 x_1 \sin(2\pi x_2) + \cos x_1 \sin^3(2\pi x_2) - 18.85\sin^2 x_1 \cos(2\pi x_2) + 18.85\sin x_1 \sin(2\pi x_2)\cos(2\pi x_2) + 2\pi\cos^2 x_1 \cos(2\pi x_2) + 39.45\cos x_1 \sin(2\pi x_2) - 248\cos(2\pi x_2)\Big)$$

The operation on the x_2 variable (y) yields

$$Ux_2 = 1 \text{ for all } k \geq 2$$

Recalling that the forward advance is done with small time increments of h (time advance), the series solution is written in terms of the mapping

$$G(x_1, x_2, h) = x_1 + \left(- \sin x_1 + \sin(2\pi x_2) \right)h + \frac{h^2}{2}\Big(\cos x_1 \sin x_1 - \cos x_1 \sin(2\pi x_2) + 2\pi\cos(2\pi x_2) \Big) + \frac{h^3}{6}\Big(\sin^3 x_1 - 2\sin^2 x_1 \sin(2\pi x_2) - \cos^2 x_1 \sin x_1 + \cos^2 x_1 \sin(2\pi x_2)$$

$$+ \sin x_1 \sin^2(2\pi x_2) - 2\pi \cos x_1 \cos(2\pi x_2) - 39.48\sin(2\pi x_2)\Big)$$

$$+ \frac{h^4}{24}\Big(- 5.0\sin^3 x_1 \cos x_1 + 11\sin^2 x_1 \cos x_1 \sin(2\pi x_2)$$

$$- 7\sin x_1 \cos x_1 \sin^2(2\pi x_2) + \cos^3(x_1)\sin x_1 - \cos^3 x_1 \sin(2\pi x_2)$$

$$+ \cos x_1 \sin^3(2\pi x_2) - 18.85\sin^2 x_1 \cos(2\pi x_2)$$

$$+ 18.85\sin x_1 \sin(2\pi x_2)\cos(2\pi x_2) + 2\pi\cos^2 x_1 \cos(2\pi x_2)$$

$$+ 39.45\cos x_1 \sin(2\pi x_2) - 248\cos(2\pi x_2)\Big) \qquad (4.36)$$

Applying the initial condition $y(0) = 0$ (for x_2), the terms containing x_2 vanish and the series expansion for the variable x_1 from equation (4.34) becomes

$$G(x_1, 0, h) = x_1 + \Big(- \sin x_1\Big)h + \frac{h^2}{2}\Big(\cos x_1 \sin x_1 + 2\pi\cos(2\pi x_2)\Big)$$

$$+ \frac{h^3}{6}\Big(\sin^3 x_1 - \cos^2 x_1 \sin x_1 - 2\pi\cos x_1\Big)$$

$$+ \frac{h^4}{24}\Big(- 5.0\sin^3 x_1 \cos x_1 + \cos^3(x_1)\sin x_1 - 18.85\sin^2 x_1$$

$$+ 2\pi\cos^2 x_1 - 248\cos(2\pi x_2)\Big) \ldots\ldots \qquad (4.37)$$

The partial derivative of **G** with respect to x_1 is

$$DG(x_1, 0, h) = 1.0 + \Big(- \cos\Big)h + \frac{h^2}{2}\Big(-\sin^2 x_1 + \cos^2 x_1\Big)$$

$$+ \frac{h^3}{6}\Big(5\sin^2 x_1 \cos x_1 - \cos^3 x_1 + 2\pi\sin x_1\Big) + \frac{h^4}{24}\Big(- 18.0\sin^2 x_1 \cos^2 x_1$$

$$+ 5\sin^4 x_1 - 50.27\sin x_1 \cos x_1\Big)\ldots \qquad (4.38)$$

Remember that $G(.)$ gives the forward advance transformation for small x interval steps (for small time advance). The $G(.)$ value obtained when the entire period of the forcing function has been reached gives the Poincare' map, $P(x)$. The Newton-Raphson technique is used, using $G(.)$ and $DG(.)$ at each small x interval step, throughout the forward transformations, constantly seeking the periodic value for that x, until the entire period is covered. The figure below illustrates the scheme of this method.

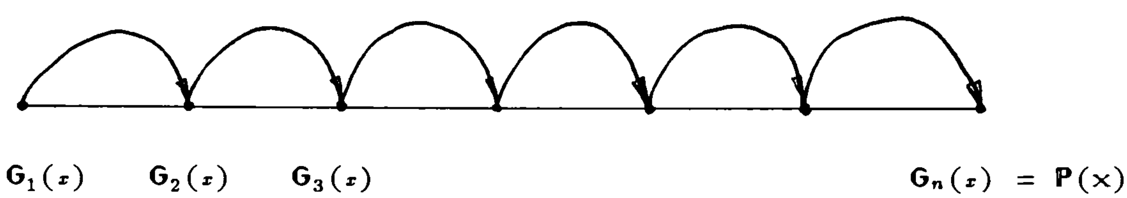


FIG 4-5. Forward advance mapping scheme

Because of the nonlinearity of the system, more than one periodic solution value was found when equations (4.36) & (4.37) (program VECFLD2 can calculate also) were entered in the program TEST.FOR. The periodic solutions for this example were found to be

$$x_0 = -0.1553, \quad 2.9863, \quad 6.1279, \quad 9.2695, \ldots..$$

which are all incremented by the value π. See the following Case Study # 10, for the plots verifying the above periodic solutions. The associated programs that do the calculations are located in Appendix A. The following problem is solved in a similar manner.

1. $\dot{x} = - x^3 + \sin(t)$ Case Study # 11

CASE STUDY # 9

$$\dot{x} = -x^2 + tx \qquad x(0) = x_0 \tag{1}$$

Objective: Find the periodic solution, i.e. IC, such that $x_0 = x(0) = x(T)$, where period T = 1.

Solution:

Since this problem is nonautonomous, make the change of variable y = t, which gives

$$\dot{y} = 1, \qquad y(0) = t_0$$

So the equivalent autonomous system is given by

$$\begin{aligned} \dot{x} &= -x^2 + xy & x(0) &= x_0 \\ \dot{y} &= 1 & y(0) &= t_0 \end{aligned} \tag{2}$$

Infinitesimal Generator Operator:

$$U = F_1\frac{\partial}{\partial x} + F_2\frac{\partial}{\partial y} \tag{3}$$

where $F_1 = (-x^2 + xy)$, $F_2 = 1$

$$U = (-x^2 + xy)\frac{\partial}{\partial x} + (1)\frac{\partial}{\partial y}$$

We obtain,

$$Ux = -x^2 + xy \tag{4}$$

$$U^2x = (-x^2 + xy)(-2x + y) + (1)(x)$$

$$= 2x^3 - 3x^2y + xy^2 + x \tag{5}$$

$$U^3x = (-x^2 + xy)(6x^2 - 6xy + y^2 + 1) + (1)(-3x^2 + 2xy)$$

$$= -6x^4 + 12x^3y - 7x^2y^2 - 4x^2 + xy^3 + 3xy \qquad (6)$$

$$U^4x = (-x^2 + xy)(-24x^3 + 36x^2y - 14xy - 8x + y^3 + 3y) +$$

$$(1)(12x^3 - 7x^2 + 3xy^2 + 3x)$$

$$= 24x^5 - 60x^4y + 6x^3y + 20x^3 - x^2y^3 - 3x^2y + 36x^3y^2 - 14x^2y^2 +$$

$$xy^4 + 6xy^2 - 7x^2 + 3x \qquad (7)$$

Also,

$$Uy = 1 \qquad (8)$$

$$U^ky = 0 \qquad \text{for all } k \geq 1 \qquad (9)$$

So,

$$x(h) = x + Uxh + U^2x\frac{h^2}{2} + U^3x\frac{h^3}{3!} + \ldots U^nx\frac{h^n}{n!} \qquad (10)$$

The series solution in terms of the mapping is

$$G(x,y,h) = x + (-x^2 + xy)h + (2x^3 - 3x^2y + xy^2 + x)\frac{h^2}{2} +$$

$$(-6x^4 + 12x^3y - 7x^2y^2 - 4x^2 + xy^3 + 3xy)\frac{h^3}{3!} + \ldots \qquad (11)$$

and,

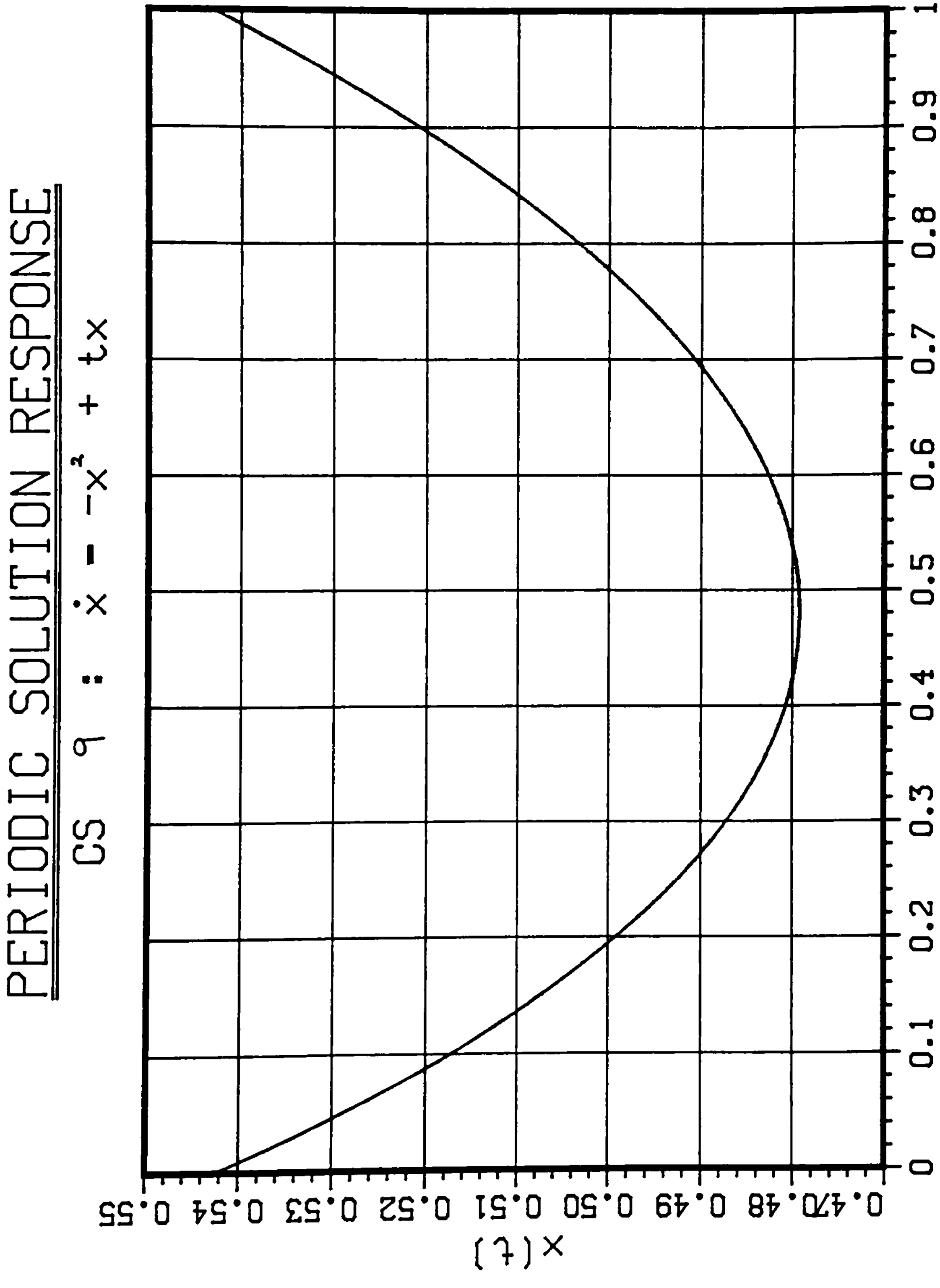
$$DG(x,y,h) = 1 + (-2x + y) + (6x^2 - 6xy + y^2 + 1)\frac{h^2}{2} + \ldots$$

$$(-24x^3 + 36x^2y - 14xy^2 - 8x + y^3 + 3y)\frac{h^3}{6} + \ldots \qquad (12)$$

Equations (11) and (12) are entered into the Fortran program TEST.FOR with period T=1, and the fixed point was found to be

$$x_0 = \quad 0.5428$$

The following plot verifies that the above value is a periodic solution.


PERIODIC SOLUTION RESPONSE
CS 9 : $\dot{x} = -x^2 + tx$
time
x(t)
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0.47 0.48 0.49 0.50 0.51 0.52 0.53 0.54 0.55

CASE STUDY # 10

$$\dot{x} + \sin(x) = \sin(2\pi t) \qquad (1)$$

Objective: Find the periodic solution of the above equation, such that $x_0 = x(0) = x(T)$, where the period $= 2\pi$.

Solution:

Since equation (1) is nonautonomous, it can be re-expressed as an autonomous one by letting $x = x_1$, and $t = x_2$. Equation (1) can be written as

$$\dot{x}_1 = -\sin(x_1) + \sin(2\pi x_2) \qquad (2)$$

equation (2) can now be replaced by two first order differential equations

$$\dot{x}_1 = -\sin(x_1) + \sin(2\pi x_3) \qquad x_1(0) = x1_0$$
$$x_2 = t \qquad x_2(0) = t_0$$

The series solution in terms of a mapping was calculated from the program VECFLD2 with the information $F_1 = -\sin(x_1) + \sin(2\pi x_2)$ and $F_2 = 1$ entered, and stored in the following subprograms

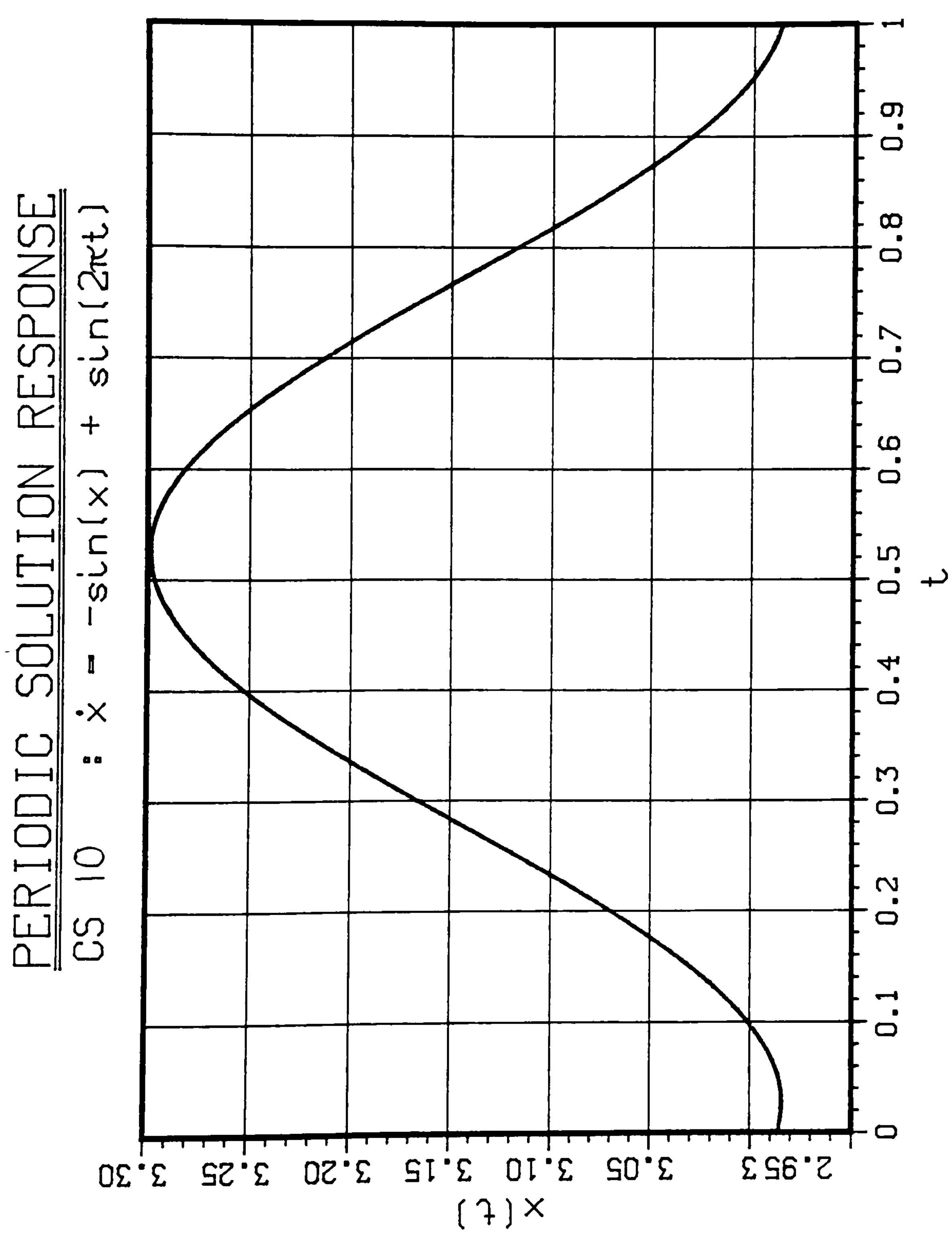
$x_1(h) = G_1(x_1, x_2, h)$ - subprogram [gnnxh.for]

$\frac{\partial G_1}{\partial x_1}(x_1, x_2, h)$ - subprogram [dgnnxh.for]

The above subprograms are called by the main Fortran program TEST.FOR to solve for the periodic solution value, with period $T = 2\pi$. Because of the nonlinearity of the system for this problem, more than one periodic solution was found

$$x_0 = -0.1553, \quad 2.9863, \quad 6.1279, \quad 9.2695, \ldots$$

which are incremented by the value π. Plots are displayed in the graphs that follow to verify the periodic solution found. Copies of all the programs used for the calculation are located in Appendix A.


PERIODIC SOLUTION RESPONSE
CS 10 : $\ddot{x} = -\sin(x) + \sin(2\pi t)$
t
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
x(t)
2.953
3.05
3.10
3.15
3.20
3.25
3.30

CASE STUDY # 11

$$\dot{x} = -x^3 + \sin(t) \qquad x(0) = x_0 \tag{1}$$

Objective: Find the periodic solution, i.e. IC, such that $x_0 = x(0) = x(T)$, where period $T = 2\pi$.

Solution:

Since this problem is nonautonomous, make the change of variable $y = t$, which gives

$$\dot{y} = 1, \qquad y(0) = t_0$$

So the equivalent autonomous system is given by

$$\begin{aligned} \dot{x} &= -x^3 + \sin(t) \qquad & x(0) &= x_0 \\ \dot{y} &= 1 \qquad & y(0) &= t_0 \end{aligned} \tag{2}$$

The series expansion was found to be from the program VECFLD2 to be

$$\mathbf{G}(x,y,h) = x + (-x^3 + \sin t)h + (3x^5 - 3x^2\sin t + \cos t)\frac{h^2}{2} + \left(-15x^7 + 21x^4\sin t - 6x(\sin t)^2 - 3x^2\cos t - \sin t\right)\frac{h^3}{3!} + \ldots \tag{3}$$

and,

$$\mathbf{DG}(x,y,h) = 1 - 3x^2h + (15x^4 - 6x\sin t)\frac{h^2}{2} + \left(-105x^6 + 84x^3\sin t - 6(\sin t)^2 - 6x\cos t\right)\frac{h^3}{6} + \ldots \tag{4}$$

Equations (3) and (4) are entered into two subprograms, and are used by the main Fortran program TEST.FOR to solve for the fixed point (periodic solution) with period $T=2\pi$. The periodic value was found to be

$$x_0 = - 0.71576$$

The above point was plotted to verify that it is a periodic solution. The graph of this value is on the following page. Also, copies of all computer programs used are in Appendix A.

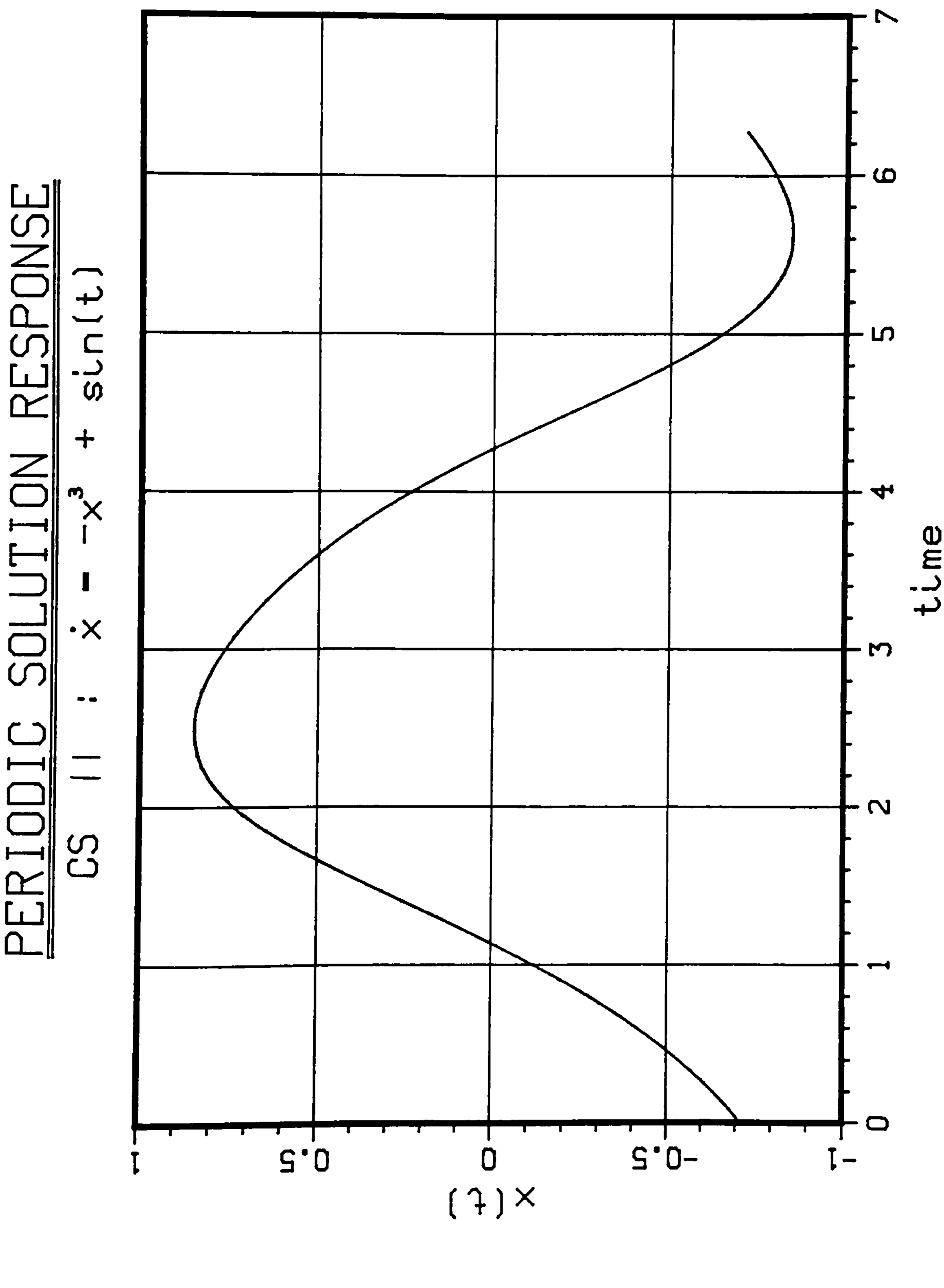
The problem was resolved with the damping term reduced to 0.1,

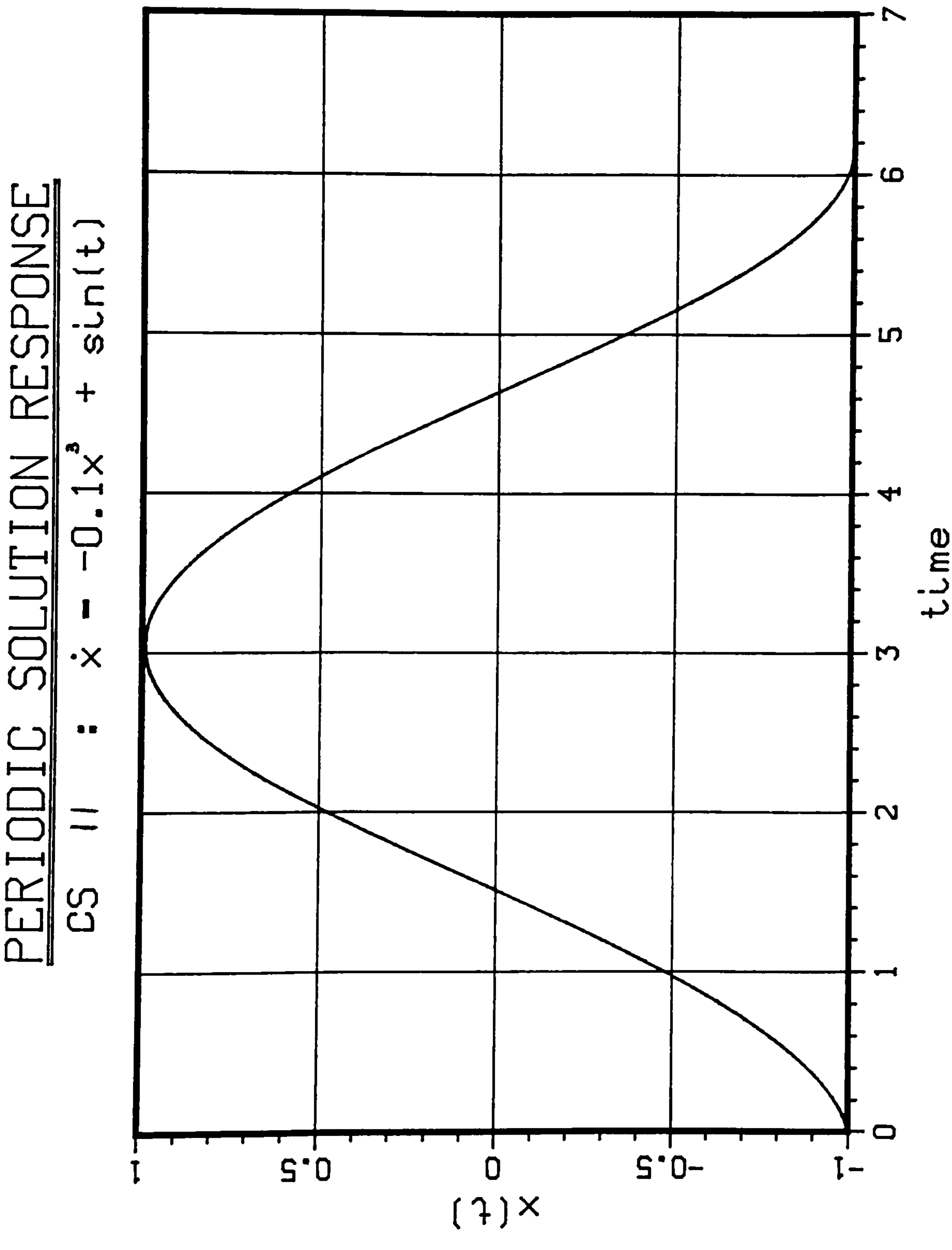
$$\dot{x} = - 0.1x^3 + \sin(t) \qquad x(0) = x_0 \tag{5}$$

The new periodic solution value was identified as

$$x_0 = - 0.9988$$

The graph for this new damping is displayed on the second plot.


PERIODIC SOLUTION RESPONSE
CS II : ẋ = −x³ + sin(t)
x(t)
1
0.5
0
-0.5
-1
time
0
1
2
3
4
5
6
7


PERIODIC SOLUTION RESPONSE
CS II : ẋ = -0.1x³ + sin(t)
time
x(t)
0
1
2
3
4
5
6
7
-1
-0.5
0
0.5
1

Higher Dimension Nonlinear Systems

For a 2-D nonlinear system, the same initial approach is taken as was done for the 1-D system, that is rewrite the system equation as an autonomous system equation (if necessary) then apply the infinitesimal generator for the series expansion for the variables. As an example consider, the nonlinear Duffing equation

$$\ddot{x} + \omega^2 x + \epsilon\omega^2(\alpha\dot{x} + \beta x^3) = F\cos(\Omega t) \qquad \epsilon \ll 1 \qquad (4.39)$$

where ϵ is the small parameter, ω is the natural frequency of the associated undamped, linear system. Ω is the driving frequency, α and β are given parameters of the system.

Since equation (4.39) is nonautonomous, the problem is re-expressed as an autonomous one by letting $x = x_1$, $\dot{x} = x_2$, $t = x_3$. Equation (4.39) can be written as

$$\dot{x}_2 + \omega^2 x_1 + \epsilon\omega^2(\alpha x_2 + \beta x_1^{\,3}) = F\cos(\Omega x_3) \qquad (4.40)$$

Using the following values for the parameters,

$$\omega = 1.0, \quad \epsilon = 0.1, \quad \alpha = 5.0, \quad \beta = 10.0, \quad F = 50.0, \quad \Omega = 2\pi$$

equation (4.40) can be replaced by the three first-order differential equations

$$\begin{aligned}
\dot{x}_1 &= x_2 & x_1(0) &= x1_0\\
\dot{x}_2 &= -\,0.5x_2 - x_1 - x_1^{\,3} + 50.0\cos(2\pi x_3) & x_2(0) &= x2_0\\
\dot{x}_3 &= 1 & x_3(0) &= x3_0
\end{aligned}$$

The infinitesimal generator operator for this third order system becomes

$$U = F_1 \frac{\partial}{\partial x_1} + F_2 \frac{\partial}{\partial x_2} + F_3 \frac{\partial}{\partial x_3} \qquad (4.41)$$

which allows the construction of the solution in the series form

$$\phi_i = x_i + t \cdot Ux_i + \frac{t^2}{2!} U^2 x_i + \frac{t^3}{3!} U^3 x_i + \ldots \qquad (4.42)$$

Using the following identifications

$$x_1 = x$$
$$\dot{x}_1 = x_2$$
$$x_3 = t$$
$$F_1 = x_2$$
$$F_2 = -0.5x_2 - x_1^3 + \cos(2\pi x_3)$$
$$F_3 = 1$$

The infinitesimal generator, from equation (4.41) is

$$U = x_2 \frac{\partial}{\partial x_1} + \left(- 0.5x_2 - x_1^3 + \cos(2\pi x_3)\right) \frac{\partial}{\partial x_2} + \frac{\partial}{\partial x_3} \qquad (4.43)$$

Operating on the variable x_1, the sequence of coefficients for the x_1 expansion is given by

$$Ux_1 = x_2$$
$$U^2x_1 = (-0.5x_2 - x_1 - x_1^3 + 50\cos 2\pi x_3)$$
$$U^3x_1 = (-0.75x_2 - 3x_2x_1^2 + 0.5x_1 + 0.5x_1^3 - 25\cos 2\pi x_3 - 314.17\sin 2\pi x_3)$$
$$U^4x_1 = (- 6x_2^2x_1 + 0.875x_2 + 3x_2x_1^2 + 0.75x_1 + 3.75x_1^3 - 2011.4\cos 2\pi x_3 + 3x_1^5 - 150x_1^2 \cos 2\pi + 157.68 \sin 2\pi x_3)$$

For the variable x_2, the sequence of coefficients is given by

$$Ux_2 = (-0.5x_2 - x_1 - x_1^3 + 50\cos 2\pi x_3)$$

$$U^2x_2 = (-0.75x_2 - 3x_2x_1^2 + 0.5x_1 + 0.5x_1^3 - 25\cos 2\pi x_3 - 314.16\sin 2\pi x_3)$$

$$U^3x_2 = (-6x_2^2x_1 + 0.875x_2 + 3x_2x_1^2 + 0.75x_1 + 3.75x_1^3 - 2011.4\cos 2\pi x_3 + 3x_1^5 - 150x_1^2\cos 2\pi x_3 + 157.1\sin 2\pi x_3)$$

$$U^4x_2 = (-6x_2^3 + 12x_2^2x_1 + 0.31x_2 + 21.75x_2x_1^2 + 27x_2x_1^4 - 1900x_2x_1\cos 2\pi x_3 - 0.875x_1 - 3.75x_1^3 + 1030.7\cos 2\pi x_3 - 3x_1^5 + 150x_1^2\cos 2\pi x_3 + 12637.9\sin 2\pi x_3 + 942.7\, x_1^2\sin 2\pi x_3)$$

The operation on the x_3 variable yields

$$Ux_3 = 1$$

and $$U^kx_3 = 0 \quad \text{for all } k > 2$$

Recalling that the forward advance is done with small time increments of h (time advance), the series solution is written in terms of the mapping

$$x_1(h) = \mathbf{G}_1(x_1, x_2, x_3, h) = x_1 + x_2h + \frac{h^2}{2}(-0.5x_2 - x_1 - x_1^3 + 50\cos 2\pi x_3) + \frac{h^3}{6}(-0.75x_2 - 3x_2x_1^2 + 0.5x_1 + 0.5x_1^3 - 25\cos 2\pi x_3 - 314.17\sin 2\pi x_3) + \frac{h^4}{24}(-6x_2^2x_1 + 0.875x_2 + 3x_2x_1^2 + 0.75x_1 + 3.75x_1^3 - 2011.4\cos 2\pi x_3 + 3x_1^5 - 150x_1^2\cos 2\pi + 157.68\sin 2\pi x_3) \tag{4.44}$$

$$\frac{\partial \mathbf{G}_1}{\partial x_1}(x_1, x_2, x_3, h) = 1.0 + \frac{h^2}{2}(-1 - 3x_1^2) + \frac{h^3}{6}(-6x_2x_1 + 0.5 + 1.5x_1^2) + \frac{h^4}{24}(-6x_2^2 + 6x_2x_1 + 0.75 + 11.25x_1^2 + 15x_1^4 - 300x_1\cos 2\pi x_3) \tag{4.45}$$

$$\frac{\partial \mathbf{G}_1}{\partial x_2}(x_1, x_2, x_3, h) = h + \frac{h^2}{2}(-0.5x_2) + \frac{h^3}{6}(-0.75 - 3x_1^2) + \frac{h^4}{24}(-12x_2x_1 + 0.875 + 3x_1^2) \tag{4.46}$$

$$x_2(h) = \mathbf{G}_2(x_1, x_2, x_3, h) = x_2 + h(-0.5x_2 - x_1 - x_1^3 + 50\cos 2\pi x_3) + \frac{h^2}{2}(-0.75x_2 - 3x_2x_1^2 + 0.5x_1 + 0.5x_1^3 - 25\cos 2\pi x_3 - 6.28\sin 2\pi x_3) + \frac{h^3}{6}(-6x_2^2x_1 + .875x_2 + 3x_2x_1^2 + .75x_1 + 3.75x_1^3 - 2011.4\cos 2\pi x_3 + 3x_1^5 - 150x_1^2 + 157.1\sin 2\pi x_3) + \frac{h^4}{24}(-6x_2^3 + 12x_2^2x_1 + .3125x_2 + 21.75x_2x_1^2 + 27x_2x_1^4 - 900x_2x_1\cos 2\pi x_3 - .875x_1 - 3.75x_1^3 + 1030.7\cos 2\pi x_3 - 3x_1^5 + 150x_1^2\cos 2\pi x_3 + 12637.9\sin 2\pi x_3 + 942.5\, x_1^2\sin 2\pi x_3) \tag{4.47}$$

$$\frac{\partial \mathbf{G}_2}{\partial x_2}(x_1, x_2, x_3, h) = 1.0 - 0.5h + \frac{h^2}{2}(-0.75 - 3x_1^2) + \frac{h^3}{6}(-12x_2x_1 + 0.875 + 3x_1^2) + \frac{h^4}{24}(-18x_2^2 + 24x_2x_1 + 0.3125 + 21.75x_1^2 + 27x_1^4 - 900x_1\cos 2\pi x_3) \tag{4.48}$$

$$\frac{\partial \mathbf{G}_2}{\partial x_1}(x_1, x_2, h) = (-1 - 3x_1^2)h + \frac{h^2}{2}(-6x_2x_1 + 0.5 + 15x_1^2) + \frac{h^3}{6}(-6x_2^2 + 6x_2x_1 + 0.75 + 112x_1^2 + 15x_1^4 - 300x_1\cos 2\pi x_3) + \frac{h^4}{24}(12x_2^2 + 43.5x_2x_1 + 108x_2x_1^3$$

$$- 900x_2\cos 2\pi x_3 - .875 - 11.625x_1^2 - 15x_1^4 + 300x_1\cos 2\pi x_3$$

$$+ 1184.9\, x_1\sin 2\pi x_3) \qquad (4.49)$$

For a 2-D system the algorithm for finding the solution is modified slightly from that used for a 1-D system, due to the presence of matrix operations. Recalling that for the system we deduce $\mathbf{G}(x)$, and then $\mathbf{P}(x) = \mathbf{G}^n(x)$, the fixed point is equal to the Poincare′ mapping of the initial condition

$$\vec{x}_0 = \mathbb{P}(\vec{x}_0) \qquad (4.50)$$

To find the initial condition that leads to a fixed point, we will need $\mathbb{P}(x)$ and $\mathbb{P}'(x)$ and use Newton-Raphson to iterate until the fixed point is located (if it exists). The calculation of $\mathbb{P}(x) = \mathbf{G}(\mathbf{G}(\ldots\mathbf{G}(x))\ldots)$, now involves matrix manipulations.

<u>Jacobian of the Composite Mapping</u>:

$$\mathbb{P}(x) = \mathbf{G}(\mathbf{G}(\ldots\mathbf{G}(x))\ldots) \qquad (4.51)$$

First, consider the composition of vector functions:

Let $y_1 = f_1(x_1,x_2,..x_m)$ $y_2 = f_2(x_1,x_2,..x_m)$ . . $y_m = f_m(x_1,x_2,..x_m)$	$\vec{y} = \vec{f}(\vec{x})$ vector function from $\mathbb{R}^m$ to $\mathbb{R}^m$

Let $\vec{g} = \vec{f}(\vec{y}) = \vec{f}\left(\vec{f}(\vec{x})\right)$ (4.52)

i.e. $g_i(x_1, x_2, \ldots, x_m) = f_i\left(f_1(\vec{x}), f_2(\vec{x}), \ldots, f_m(\vec{x})\right)$

$$\begin{array}{ccc} \| & \| & \| \\ y_1 & y_2 & y_m \end{array}$$

then for each $j = 1, 2, \ldots, m$

$$\frac{\partial g_i}{\partial x_j} = \sum_{k=1}^{m} \frac{\partial f_i}{\partial y_k} \cdot \frac{\partial y_k}{\partial x_j} = \left[\frac{\partial f_i}{\partial y_1}, \frac{\partial f_i}{\partial y_2}, \ldots, \frac{\partial f_i}{\partial y_m}\right] \cdot \begin{vmatrix} \frac{\partial y_1}{\partial x_j} \\ \frac{\partial y_2}{\partial x_j} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial f_m}{\partial x_j} \end{vmatrix} \qquad (4.53)$$

$$= \underbrace{\left[\frac{\partial f_i}{\partial y_1}, \frac{\partial f_i}{\partial y_2}, \ldots, \frac{\partial f_i}{\partial y_m}\right]}_{i^{th}\text{ row of } [J\vec{f}](\vec{y})} \cdot \underbrace{\left[\frac{\partial y_1}{\partial x_j}, \frac{\partial y_2}{\partial x_j}, \ldots, \frac{\partial y_m}{\partial x_j}\right]^T}_{j^{th}\text{ column of } [J\vec{f}](\vec{x})} \qquad (4.54)$$

Thus

$$\frac{\partial g_i}{\partial x_j} \equiv ij^{th} \text{ element of } [J\vec{g}](\vec{x})$$

$$[J\vec{g}](\vec{x}) = [J\vec{f}(\vec{y})] \cdot [Jf(\vec{x})] \qquad (4.55)$$

(↑ matrix product)

$$\Rightarrow \quad J[\vec{f}(\vec{f}(\vec{x}))] = J\vec{f}[\vec{f}(\vec{x})] \cdot Jf[\vec{x}]$$

Recalling the forward advance scheme

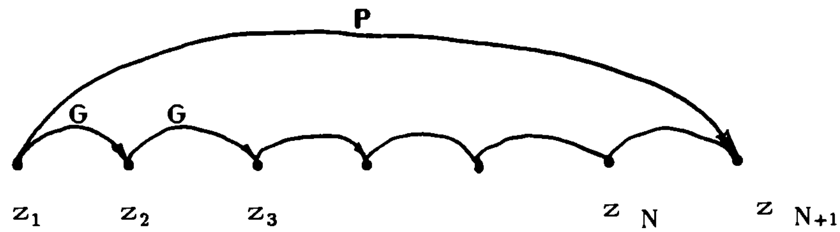


FIG 4-6.

<u>Vector Version</u>:

$$\vec{z}_{N+1} = \mathbb{P}(\vec{z}_1) = \mathbf{G}(\vec{z}_N) \tag{4.56}$$

$$= \mathbf{G}\Big(\mathbf{G}(\vec{z}_{N-1})\Big)$$

$$\Delta \qquad J\mathbb{P}[\vec{z}_1] = J\mathbf{G}[\mathbf{G}(\vec{z}_{N-1})] \cdot J\mathbf{G}(\vec{z}_{N-1}) \tag{4.57}$$

$$= J\mathbf{G}(\vec{z}_N) \cdot J\mathbf{G}((\vec{z}_{N-1}) \qquad \text{matrix product}$$

Now,

$$\left.\begin{aligned} \mathbb{P}(z_1) &= \mathbf{G}(\mathbf{G}(\mathbf{G}(\vec{z}_{N-1}))) \\ J\mathbb{P}[\vec{z}_1] &= J\mathbf{G}[z_N] \cdot J\mathbf{G}[z_{N-1}] \cdot J\mathbf{G}[z_{N-2}] \end{aligned}\right| \tag{4.58}$$

So since $\mathbb{P}(z_1) = \mathbf{G} \cdot \mathbf{G} \cdot \mathbf{G} \cdot \ldots \mathbf{G}[\vec{z}_1]$ (4.59)

↑ composition

$$J\mathbb{P}[\vec{z}_1] = J\mathbf{G}[z_N] \cdot J\mathbf{G}[z_{N-1}] \cdot J\mathbf{G}[z_{N-2}] \cdot \cdots \cdot J\mathbf{G}[\vec{z}_1]$$

↑ matrix product

At each step,

$$JG = \begin{bmatrix} \frac{\partial(xh)}{\partial x} & \frac{\partial(xh)}{\partial y} \\ \\ \frac{\partial(yh)}{\partial x} & \frac{\partial(yh)}{\partial y} \\ \\ \end{bmatrix} \qquad (4.60)$$

$$= \begin{bmatrix} dxh & dxhyh \\ \\ dyhxh & dyh \\ \\ \end{bmatrix} \qquad (4.61)$$

where the entries are partial derivatives of the forward advance map (xh,yh).

Now going back to the example, to calculate the periodic solution, we will need the equation,

$$\mathbb{DP} = J[\mathbf{G}^n]$$

$$\vec{x}_{k+1} = \vec{x}_k - [I - \mathbb{DP}(\vec{x}_k)]^{-1} [\vec{x}_k - \mathbb{P}(\vec{x}_k)] \qquad (4.62)$$

or $\vec{x}_{k+1} = \vec{x}_k - A\ [\vec{x}_k - \mathbb{P}(\vec{x}_k)]$

where $A = [I - \mathbb{DP}(\vec{x}_k)]^{-1}$

$$\text{let } B = [I - \mathbf{P}(\vec{x}_k)]$$

$$\text{and} \quad \mathbf{DP} = \begin{vmatrix} \frac{\partial \mathbf{P}_1}{\partial x} & \frac{\partial \mathbf{P}_1}{\partial y} \\ \frac{\partial \mathbf{P}_2}{\partial x} & \frac{\partial \mathbf{P}_2}{\partial y} \end{vmatrix}$$

$$\text{so } B = \begin{vmatrix} 1 - \frac{\partial \mathbf{P}_1}{\partial x} & - \frac{\partial \mathbf{P}_1}{\partial y} \\ - \frac{\partial \mathbf{P}_2}{\partial x} & 1 - \frac{\partial \mathbf{P}_2}{\partial y} \end{vmatrix}$$

In order to determine the inverse of B, let

$$B = \begin{vmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{vmatrix}$$

then in terms of the components b_{ij},

$$B^{-1} = \frac{1}{b_{11}b_{22} - b_{12}b_{21}} \begin{bmatrix} b_{22} & -b_{12} \\ -b_{21} & b_{11} \end{bmatrix}$$

So A can now be written as

$$A = \frac{1}{b_{11}b_{22} - b_{12}b_{21}} \begin{bmatrix} 1 - \frac{\partial \mathbf{P}_2}{\partial y} & \frac{\partial \mathbf{P}_1}{\partial y} \\ \frac{\partial \mathbf{P}_2}{\partial x} & 1 - \frac{\partial \mathbf{P}_1}{\partial x} \end{bmatrix}$$

The updated x_k can now be calculated from equation (4.62)

$$\vec{x}_{k+1} = \vec{x}_k - A\,[\vec{x}_k - \mathbf{P}(\vec{x}_k)]$$

The program that does this manipulation is MAIN.FOR, and is located in Appendix A. The Maple program [VECFLD3] used for calculating the forward advance map and partials is also present in Appendix A, along with the six subprograms generated by VECFLD3 (for xh, yh, dxh, dyh, dxhyh, dyhxh).

The periodic solution for the example (Duffing oscillator)

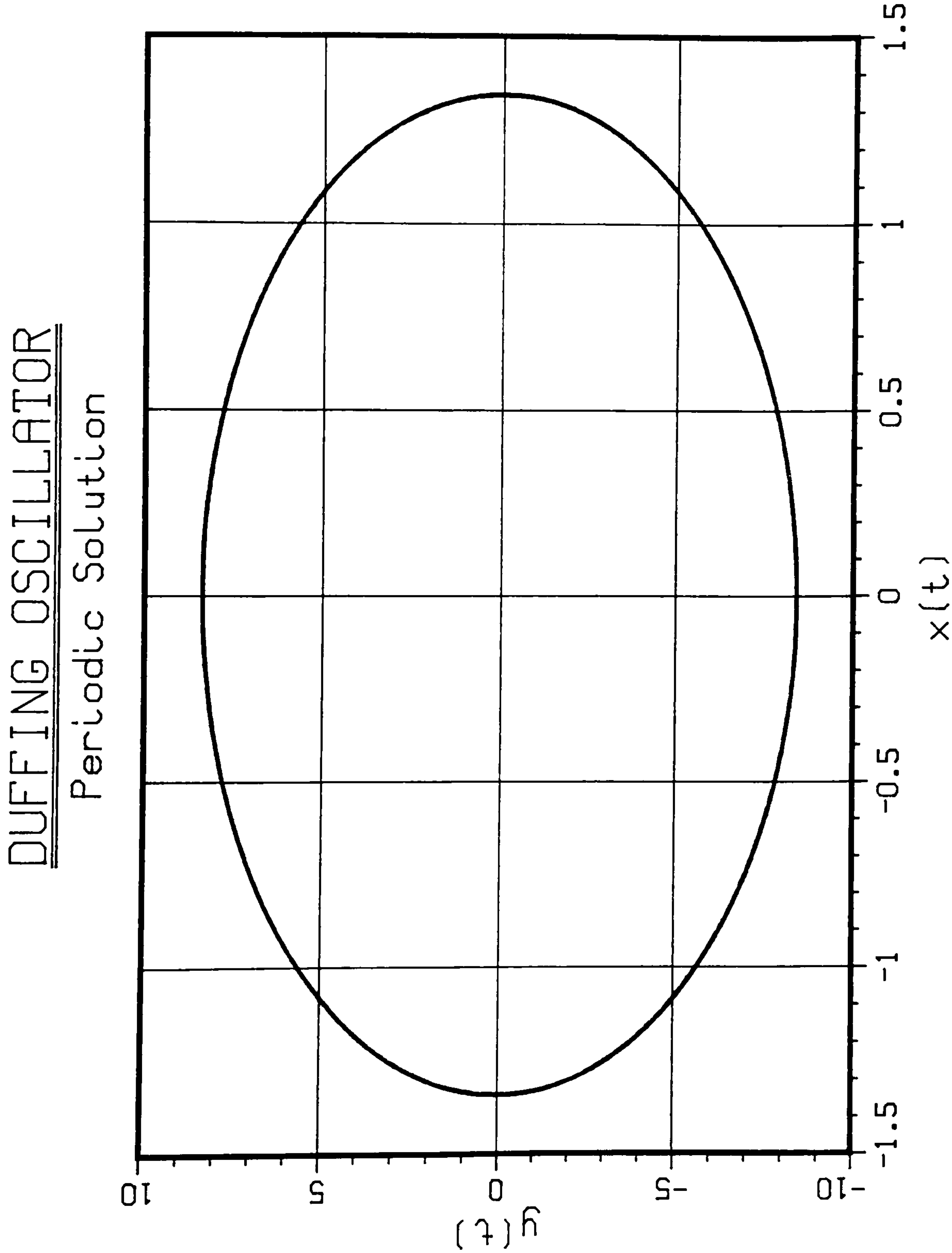
$$\ddot{x} + 0.5\dot{x} + x + x^3 = 50.0\cos(2\pi t)$$

was found to be

$$x_0 = -1.338644$$

$$\dot{x}_0 = 0.7197574$$

The following graph show the phase plot confirming the above periodic solution found. Some examples follow detailing this procedure. Higher dimension nonlinear equations are handled in a similar manner.


DUFFING OSCILLATOR
Periodic Solution
x(t)
y(t)
-1.5
-1
-0.5
0
0.5
1
1.5
10
5
0
-5
-10

CASE STUDY # 12

$$\ddot{x} - \alpha(1 - x^2)\dot{x} + x = F \sin(\Omega t) \qquad (1)$$

Objective: Find the periodic solution of the above Vanderpol equation, such that $x_0 = x(0) = x(T)$, where the period $\Omega = 2\pi$.

Solution:

Since equation (1) is nonautonomous, it can be re-expressed as an autonomous one by letting $x = x_1$, $\dot{x} = x_2$ and $t = x_3$. Equation (1) can be written as

$$\dot{x}_2 - \alpha(1 - x_1^2)x_2 + x_1 = F \sin(\Omega x_3) \qquad (2)$$

Using the following values for the parameters,

$$\alpha = 0.2, \quad \Omega = 2\pi, \quad F = 50$$

equation (2) can now be replaced by three first order differential equations

$$\dot{x}_1 = x_2 \qquad x_1(0) = x1_0$$
$$\dot{x}_2 = - x_1 - 0.2(x_1^2)x_2 + 0.2x_2 + F \sin(\Omega x_3) \qquad x_2(0) = x2_0$$
$$\dot{x}_3 = 1 \qquad x_3(0) = x3_0$$

The series solution in terms of a mapping was calculated from the program VECFLD3 with the information $F_1 = x_2$, $F_2 = \dot{x}_2$ and $F_3 = 1$ entered, and stored in the following subprograms

$x_1(h) = G_1(x_1, x_2, x_3, h)$ - subprogram [ag3xh.for]

$x_2(h) = G_2(x_1, x_2, x_3, h)$ - subprogram [ag3yh.for]

$\frac{\partial G_1}{\partial x_1}(x_1, x_2, x_3, h)$ - subprogram [ag3dxh.for]

$\frac{\partial G_2}{\partial x_2}(x_1, x_2, x_3, h)$ - subprogram [ag3dyh.for]

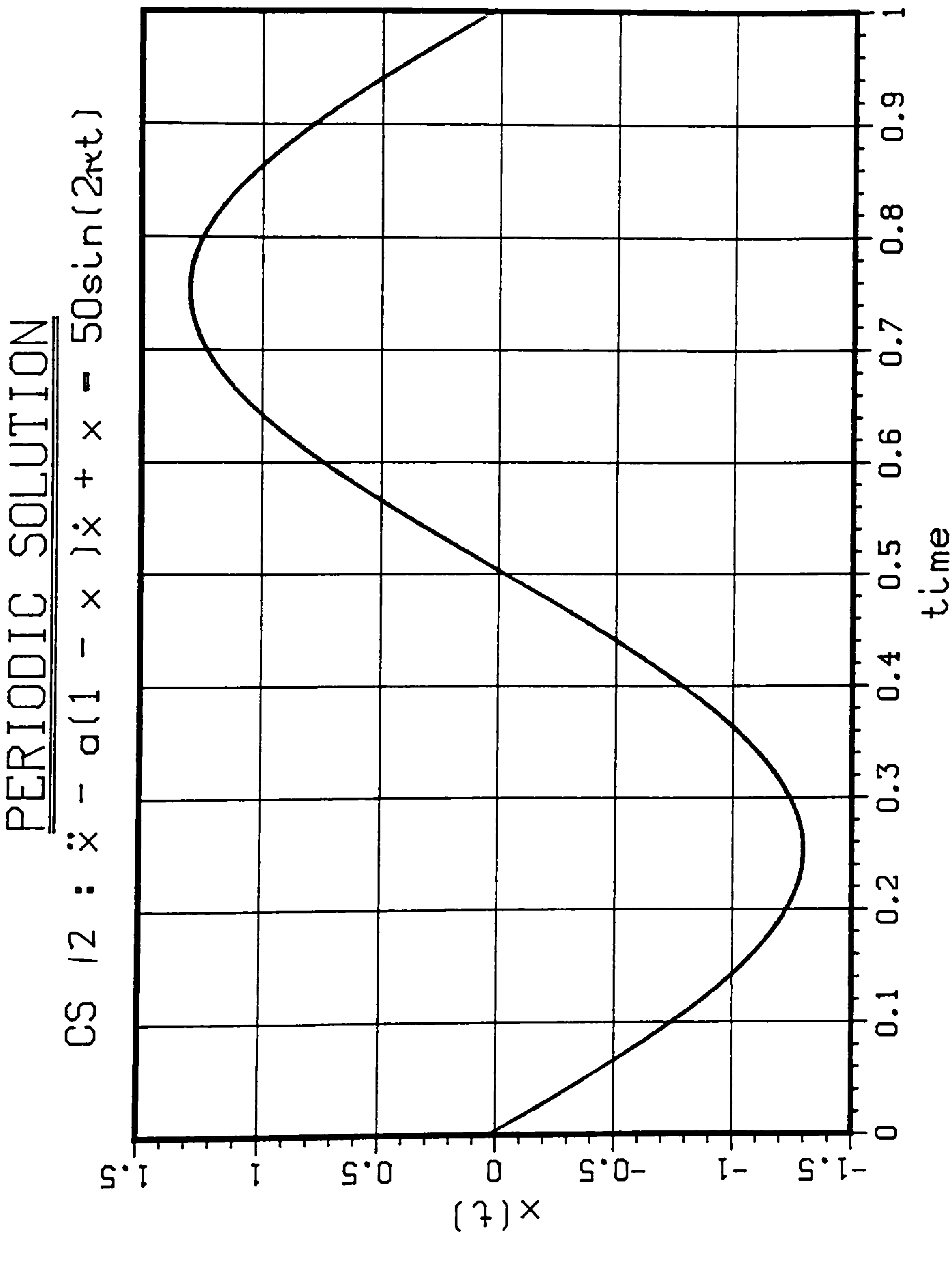
$\frac{\partial G_1}{\partial x_2}(x_1, x_2, x_3, h)$ - subprogram [ag3dxhyh.for]

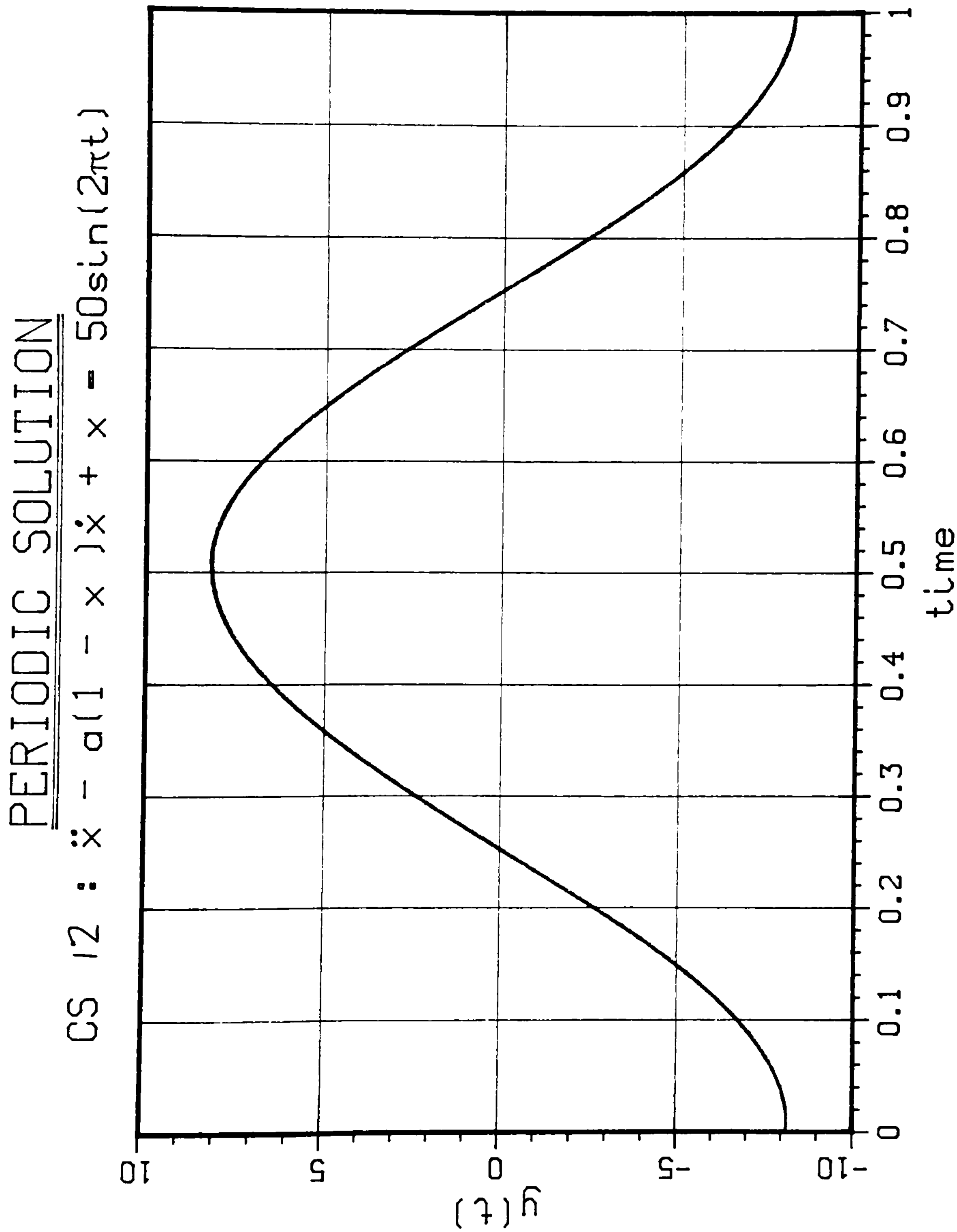
$\frac{\partial G_2}{\partial x_1}(x_1, x_2, x_3, h)$ - subprogram [ag3dyhxh.for]

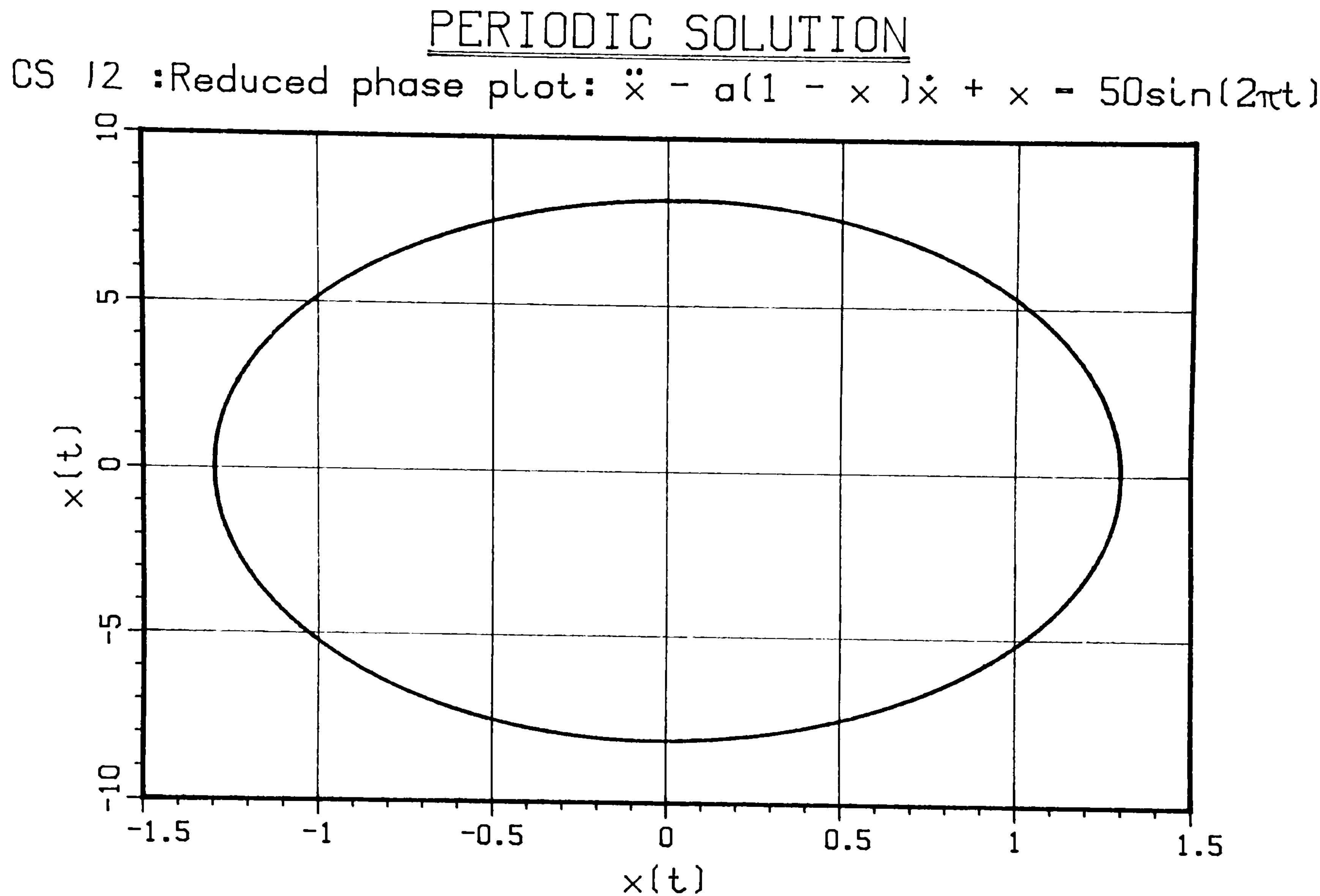
The above subprograms are called by the main Fortran program MAIN.FOR to solve for the periodic solution point with period $\Omega = 2\pi$. The periodic solution was found to be

$$x_0 = 0.02755$$
$$\dot{x}_0 = -8.15775$$

Plots are displayed in the figures that follow to verify the periodic solution found. Copies of all the programs used for calculation are in Appendix A.


PERIODIC SOLUTION
CS 12 : ẍ - a(1 - x)ẋ + x = 50sin(2πt)
time
x(t)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
1.5
1
0.5
0
-0.5
-1
-1.5


PERIODIC SOLUTION
CS 12 : ẍ - a(1 - x)ẋ + x = 50sin(2πt)
y(t)
10
5
0
-5
-10
time
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


PERIODIC SOLUTION
CS 12 :Reduced phase plot: ẍ - a(1 - x)ẋ + x = 50sin(2πt)
10
5
0
-5
-10
x(t)
-1.5
-1
-0.5
0
0.5
1
1.5
x(t)

CONCLUSIONS and RECOMMENDATIONS

The technique developed in this investigation can be used to locate periodic solutions for forced linear and nonlinear systems. The technique is semi-automated by the use of MAPLE (symbolic mathematics program). Symbolic computation allows for efficient manipulation of differential equations and initial conditions, leading to periodic solutions. For linear systems, as well as nonlinear systems, a MAPLE program algorithm was developed to generate Infinitesimal Generator series expansions of the solutions. These solutions were easily converted to Fortran code (as subroutines) by MAPLE and easily called by the main Fortran program for the determination of initial conditions which give rise to periodic solutions. Although attention has been given to the solution of first and second order equations, the technique can easily be extended to higher-order equations, as pointed out in the respective "development sections" for linear and nonlinear systems.

Steady state solutions were analyzed by utilizing the associated Poincare' Mapping of flow. It was shown that periodic solutions of dynamical systems correspond to fixed points of the time-advance mapping. For linear systems, the Poincare' Map can in principle be explicitly determined. This is based on the fact that forced solutions can be developed from the fundamental solutions of the system. For first-order systems, this turns out to be an elementary exercise. For higher order systems, the explicit calculations can be formidable. The proposed method is efficient to use and does not compromise any accuracy for stable linear systems.

The Poincare' map for a nonlinear system cannot in general be analytically derived. This would require solution of the governing equations. Instead, series approximations were developed, based on Lie Series expansions. These solutions were only approximate, but the ability to symbolically compute the series expansions out to an arbitrary number of terms allows for highly accurate analytical expressions for the Poincare' map. Even the brute-force method is only an approximation. When the algorithm converges, it does offer a highly efficient way of finding periodic solutions that would otherwise be based on trial and error.

For nonlinear systems, the one drawback of the proposed methodology is that the initial guess required to start the Newton-Raphson iteration must be in the basin of attraction of the periodic solution (if it exists). Otherwise, the algorithm will not converge due to instabilities or the presence competing basins of attraction.

Recommendations for the extension of this work include modification of the programs to allow reverse mapping (that is, backward time stepping). Such a procedure could be used to locate periodic solutions that are unstable under a positive time advance. Perhaps a global search method could be developed which would seek out all fixed points of the Poincare' map in a specified region of state space. Indeed, some nonlinear systems support multiple or even an infinite number of periodic solutions. As a final recommendation, the computer codes could be modified to enable computation of the eigenvalues of the Poincare' map. These associated eigenvalues measure the orbital stability characteristics of periodic solutions.

REFERENCES

1. P. Hagedorn, Nonlinear Oscillations, Clarendon Press, Oxford (1981).

2. L. Meirovitch, Analytical Methods in Vibrations, Macmillan, Toronto (1967).

3. T.J. Aprille, Jr and T.N. Trick, "A Computer Algorithm to Determine the Steady-State Response of Nonlinear Oscillators", IEEE Transactions on Circuit Theory, Vol CT-19, No. 4, pp. 354-360 (July 1972).

4. T.S. Parker and L.O. Chua, "The Double Scroll Equation", IEEE Transactions on Circuits and Systems, Vol CAS-34, pp. 1059-1073 (September 1987).

5. T.J. Aprille, Jr and T.N. Trick, "Steady State Analysis of Nonlinear Circuits with Periodic Inputs", Proceedings of the IEEE, Vol 60, No. 1 (January 1972).

6. N. Minorsky, Nonlinear Oscillations, Nostrand, Princeton New Jersey (1962).

7. E.R. Rainville and P.E. Bedient, Elementary Differential Equations, Macmillan, New York.

8. J.S. Torok and S.H. Advani, "Continuous Transformation Groups and Series Solution of Initial Value Problems", Int. J. Non-Linear Mech., Vol 20,No. 4, pp.283-289 (1985).

9. Maple - Symbolic Computation Program

10. J.R. Ward and R.D. Strum, State Variable Analysis, Prentice-Hall, Englewood Cliffs, New Jersey (1970).

11. J.S. Torok, "Symbolic Computation of Lie Series for the Solution of Ordinary Differential Equations", Proceedings, 1991 ASEE Annual Conference, New Orleans.

12. A. Cohen, An Introduction to the Lie Theory of One-Parameter Groups, D.C. Heath, Boston (1911).

13. P. Tondeur, Introduction to Lie Groups and Transformation Groups, Springer-Valley, Berlin (1969).

14. G.M. Swisher, Introduction to Linear Systems Analysis, Matrix Publishers, Inc., Oregon (1976).

APPENDIX A

LIST OF PROGRAMS

```
*********************************************************************
*                                                                    *
*     Program:    TEST.FOR                                           *
*                                                                    *
*   Objective:    For solving 1-D nonlinear system equations         *
*                 for their periodic solution                        *
*                                                                    *
**********************************************************************
*
      Dimension z(5000)
C   Define forward-advance map:
*
      NRS=100
      period = 1.
      type*,' '
*
C   Input initial guess and number of steps per period
*
      type*, 'Starting points xp, and number of steps?  '
      accept*, xp,nsteps
      hh = period/float(nsteps)
*
      do 50 jj = 1,NRS
*
      z(1) = xp
*
*
      do 10 k = 1,nsteps
*
        z(k+1) = gnnxh(z(k),float(k-1)*hh,hh)
*        print *,'Z =',z(k+1),'jj =',jj
*
*
   10 continue
*
      P = z(nsteps+1)
      DP = dgnnxh(z(1),0.,hh)
*
*
      do 20 k = 2,nsteps
        DP = dgnnxh(z(k),float(k-1)*hh,hh)*DP
*
   20 continue
*
      xp = xp - (xp - P )/(1. - DP )
*
*
   50 continue
*
      Print *,'The Periodic Solution is  = ',xp
      stop
      end
*
      include 'gnnxh.for'
      include 'dgnnxh.for'
```

```
*********************************************************************
*      PROGRAM:  MAIN.FOR                                           *
*                                                                   *
*   OBJECTIVE:  THIS PROGRAM SOLVES 2-D NONLINEAR EQUATIONS FOR     *
*               PERIODIC SOLUTIONS.                                 *
*                                                                   *
*********************************************************************
*
      Parameter NP=2
      Dimension x(5000),y(5000),A(np,np),D(np,np),B(np,np)
*
C  Define forward-advance map:
*
      period = 1.
      type*,' '
C  Input initial guess and number of steps per period
C
      type*, 'Starting points xp, yp and number of steps, and NRS?
      accept*, xp,yp,nsteps,NRS
      hh = period/float(nsteps)
*
      do 500 jj = 1,NRS

      x(1) = xp
      y(1) = yp
*
      do 5 k = 1,nsteps
         x(k+1) = aG3xh(x(k),y(k),float(k-1)*hh,hh)
         y(k+1) = aG3yh(x(k),y(k),float(k-1)*hh,hh)
C         print*,x(k+1),y(k+1)

    5 continue


      DP1X = ag3dxh(x(1),y(1),0.0,hh)
      DP1Y = ag3dxhyh(x(1),y(1),0.0,hh)
      DP2X = ag3dyhxh(x(1),y(1),0.0,hh)
      DP2Y = ag3dyh(x(1),y(1),0.0,hh)

      Do 10 k=2,nsteps
*
      DNP1X = ag3dxh(x(k),y(k),float(k-1)*hh,hh)
      DNP1Y = ag3dxhyh(x(k),y(k),float(k-1)*hh,hh)
      DNP2X = ag3dyhxh(x(k),y(k),float(k-1)*hh,hh)
      DNP2Y = ag3dyh(x(k),y(k),float(k-1)*hh,hh)

      P1X = DNP1X*DP1X + DNP1Y*DP2X
      P1Y = DNP1X*DP1Y + DNP1Y*DP2Y
      P2X = DNP2X*DP1X + DNP2Y*DP2X
      P2Y = DNP2X*DP1Y + DNP2Y*DP2Y

      DP1X = P1X
      DP1Y = P1Y
      DP2X = P2X
      DP2Y = P2Y

   10 Continue
*
C  Set up Matrix B
```

```
      B(1,1) = 1.0-DP1X
      B(1,2) = -DP1Y
      B(2,1) = -DP2X
      B(2,2) = 1.0-DP2Y

*   Compute B inverse

      B1=1.0/(B(1,1)*B(2,2) - B(1,2)*B(2,1))

      D(1,1) = B(2,2)
      D(1,2) = -B(1,2)
      D(2,1) = -B(2,1)
      D(2,2) = B(1,1)

      Do 11 I=1,np
         Do 12 J=1,np

         A(I,J) =B1*D(I,J)

   12    Continue
   11 Continue
C
C       Print *,((A(i,j), j=1,np),i=1,np)
c

c


c
      diffx = xp - x(nsteps+1)
      diffy = yp - y(nsteps+1)
*
      xp = xp - A(1,1)*diffx-A(1,2)*diffy
      yp = yp - A(2,1)*diffx-A(2,2)*diffy

c


*
  500 continue
*
      Print *,'The Periodic Solution is  = ',xp, yp
      stop
      end
*    Subprograms called
      function ag3xh(x1,x2,x3,h)
      include 'ag3xh.for'
      return
      end

      function ag3dxh(x1,x2,x3,h)
      include 'ag3dxh.for'
      return
      end

      function ag3yh(x1,x2,x3,h)
      include 'ag3yh.for'
      return
      end

      function ag3dyh(x1,x2,x3,h)
```

```
include 'ag3dyh.for'
return
end

function ag3dxhyh(x1,x2,x3,h)
include 'ag3dxhyh.for'
return
end

function ag3dyhxh(x1,x2,x3,h)
include 'ag3dyhxh.for'
return
end
```

```
#                   Program VECFLD2
#          This code computes the Nth order series approximation of the
#          solution to a system of 2 first order differential equations.
#          The time advance variable is denoted by h.
#
with( linalg ):
#          Define the right hand side of the equation.
#
F1 :=-0.1*x1^3+sin(x2);
F2 := 1;
#
vars := [x1,x2]:
#
vec := array([F1,F2]):
w := array([x1,x2]):
#
xh := x1:
yh := x2:
#
for k from 1 to N do
          J := jacobian(w,vars):
#
          Uw := evalm(J&*vec):
#
          xh := xh + Uw[1]*(h^k)/factorial(k):
          yh := yh + Uw[2]*(h^k)/factorial(k):
          w := Uw:
od:
#          Print the Nth order series approximation
#
xh;
yh;
F:=proc(a,b)
          subs(x1=a,x2=b,xh):
end:
G:=proc(a,b)
          subs(x1=a,x2=b,yh):
end:
#
dxh:=diff(xh,x1):
readlib(fortran):
xh1:=evalf(xh):
dxh1:=evalf(dxh):
fortran(xh1,filename='gxh.for',optimized):
fortran(dxh1,filename='dgxh.for',optimized):
```

```
#           Program VECFLD3
#     This code computes the Nth order series approximation of
#     the solution to a system of 3 first order differential
#     equations. The time advance variable is denoted by h.
#
with (linalg):
#         Define the right hand side of the equation.
#
F1 := x2;
F2 := -1.0*x1+.2*x2-.2*x1*x2+50*sin(6.28315*x3);
F3 := 1;
#
vars :=[x1,x2,x3]:
#
vec := array([F1,F2,F3]):
w := array([x1,x2,x3]):
#
xh := x1:
yh := x2:
zh := x3:
#
for k from 1 to N do
        J := jacobian(w,vars):
#
        Uw := evalm(J&*vec):
#
        xh := xh + Uw[1]*(h^k)/factorial(k):
        yh := yh + Uw[2]*(h^k)/factorial(k):
        zh := zh + Uw[3]*(h^k)/factorial(k):
od:
#       Print the Nth order series approximation
#
xh;
yh;
zh;
#
dxh:=diff(xh,x1):
dyh:=diff(yh,x2):
dxhyh:=diff(xh,x2):
dyhxh:=diff(yh,x1):
readlib(fortran):
dxhyh1:=evalf(dxhyh):
dyhxh1:=evalf(dyhxh):
xh1:=evalf(xh):
dxh1:=evalf(dxh):
yh1:=evalf(yh):
dyh1:=evalf(dyh):
fortran(xh1,filename=`ag3xh.for`,optimized):
fortran(dxh1,filename=`ag3dxh.for`,optimized):
fortran(yh1,filename=`ag3yh.for`,optimized):
fortran(dyh1,filename=`ag3dyh.for`,optimized):
fortran(dxhyh1,filename=`ag3dxhyh.for`,optimized):
fortran(dyhxh1,filename=`ag3dyhxh.for`,optimized):
#
```

```
                    ********************************************************
*
*    PROGRAM:   RUNG4.FOR
*
*    FUNCTION:  R-K routine for a system of 2.5 diff. equ.s
*
*    VARIABLES:     phase plot for010, x(t) for015, y(t) for020

*************************************************************************
C
C DEFINE SYSTEM HERE
C
      F(t,x,y) = x + t*t
      G(t,x,y) = 1.0
C
C  Parameters
C
      t0 = 0.
      M = 2
      N = 1000
C
C   INPUTS
C
      type*,' Input  xic, yic, and Tfinal'
      accept*,xic,yic,tf
C
      h = (tf-t0)/float(N)
      t = t0
C
C  Print initial values
C
      type 101 , t,xic,yic
  101 format(//,5x,' Initial values ',f15.7,x,f15.7,x,f15.7)
      write(10,555) xic,yic

      w1 = xic
      w2 = yic

C  ITERATION

      DO 666, II = 1,N

         rk11 = h*F(t,w1,w2)
         rk12 = h*G(t,w1,w2)

         rk21 = h*F(t+.5*h,w1+.5*rk11,w2+.5*rk12)
         rk22 = h*G(t+.5*h,w1+.5*rk11,w2+.5*rk12)

         rk31 = h*F(t+.5*h,w1+.5*rk21,w2+.5*rk22)
         rk32 = h*G(t+.5*h,w1+.5*rk21,w2+.5*rk22)

         rk41 = h*F(t+h,w1+rk31,w2+rk32)
         rk42 = h*G(t+h,w1+rk31,w2+rk32)

         w1   = w1+(rk11+2.*rk21+2.*rk31+rk41)/6.
         w2   = w2+(rk12+2.*rk22+2.*rk32+rk42)/6.

         t = t + h

         write(10,555) w1,w2
```

```
        write(15,555) t,w1
        write(20,555) t,w2
555     Format(5x,f15.7,5x,f15.7)
666 Continue

    STOP
    END
```

Programs generated by VECFLD3 and used for the Duffing example calculation

g3xh.for	$[x_1(h)]$
g3yh.for	$[x_2(h)]$
g3dxh.for	$[\frac{\partial \mathbf{G}_1}{\partial x_1}(x_1, x_2, x_3, h)]$
g3dyh.for	$[\frac{\partial \mathbf{G}_2}{\partial x_2}(x_1, x_2, x_3, h)]$
g3dxhyh.for	$[\frac{\partial \mathbf{G}_1}{\partial x_2}(x_1, x_2, x_3, h)]$
g3dyhxh.for	$[\frac{\partial \mathbf{G}_2}{\partial x_1}(x_1, x_2, x_3, h)]$

g3xh.for

```
 t4 = x1**2
 t5 = t4*x1
 t7 = 0.628315E1*x3
 t8 = cos(t7)
 t11 = h**2
 t15 = x2*t4
 t20 = sin(t7)
 t26 = x2**2
 t34 = t4**2
 t41 = t11**2
 t44 = x1+x2*h+0.5E0*(-0.5E0*x2-x1-t5+50.0*t8)*t11+0.1666667E0*(-0.
+75E0*x2-3.0*t15+0.5E0*x1+0.5E0*t5-0.25E2*t8-0.3141575E3*t20)*t11*h
++0.4166667E-1*(-6.0*t26*x1+0.875E0*x2+0.3E1*t15+0.75E0*x1+0.375E1*
+t5-0.2011399E4*t8+3.0*t34*x1-150.0*t4*t8+0.1570788E3*t20)*t41
 g3xh = t44
```

g3yh.for

```
t3 = x1**2
t4 = t3*x1
t6 = 0.628315E1*x3
t7 = cos(t6)
t12 = x2*t3
t17 = sin(t6)
t20 = h**2
t23 = x2**2
t24 = t23*x1
t31 = t3**2
t32 = t31*x1
t34 = t3*t7
t62 = t20**2
t65 = x2+(-0.5E0*x2-x1-t4+50.0*t7)*h+0.5E0*(-0.75E0*x2-3.0*t12+0.5
+E0*x1+0.5E0*t4-0.25E2*t7-0.3141575E3*t17)*t20+0.1666667E0*(-6.0*t2
+4+0.875E0*x2+0.3E1*t12+0.75E0*x1+0.375E1*t4-0.2011399E4*t7+3.0*t32
+-150.0*t34+0.1570788E3*t17)*t20*h+0.4166667E-1*(-6.0*t23*x2+0.12E2
+*t24+0.3125E0*x2+0.2175E2*t12+27.0*x2*t31-900.0*x2*x1*t7-0.875E0*x
+1-0.3875E1*t4+0.1030699E4*t7-0.3E1*t32+0.15E3*t34+0.1263792E5*t17+
+0.9424725E3*t3*t17)*t62
g3yh = t65
```

g3dxh.for

```
 t1 = x1**2
 t4 = h**2
 t7 = x2*x1
 t14 = x2**2
 t18 = t1**2
 t25 = t4**2
 t28 = 1.0+0.5E0*(-1.0-3.0*t1)*t4+0.1666667E0*(-6.0*t7+0.5E0+0.15E1
+*t1)*t4*h+0.4166667E-1*(-6.0*t14+0.6E1*t7+0.75E0+0.1125E2*t1+15.0*
+t18-300.0*x1*cos(0.628315E1*x3))*t25
 g3dxh = t28
```

g3dyh.for

```
 t2 = x1**2
 t5 = h**2
 t8 = x2*x1
 t15 = x2**2
 t19 = t2**2
 t26 = t5**2
 t29 = 1.0-0.5E0*h+0.5E0*(-0.75E0-3.0*t2)*t5+0.1666667E0*(-12.0*t8+
+0.875E0+0.3E1*t2)*t5*h+0.4166667E-1*(-18.0*t15+0.24E2*t8+0.3125E0+
+0.2175E2*t2+27.0*t19-900.0*x1*cos(0.628315E1*x3))*t26
 g3dyh = t29
```

g3dxhyh.for

```
 t1 = h**2
 t3 = x1**2
 t13 = t1**2
 t16 = h-0.25E0*t1+0.1666667E0*(-0.75E0-3.0*t3)*t1*h+0.4166667E-1*(
+-12.0*x2*x1+0.875E0+0.3E1*t3)*t13
 g3dxhyh = t16
```

g3dyhxh.for

```
 t1 = x1**2
 t5 = x2*x1
 t9 = h**2
 t12 = x2**2
 t16 = t1**2
 t18 = 0.628315E1*x3
 t19 = cos(t18)
 t20 = x1*t19
 t40 = t9**2
 t43 = (-1.0-3.0*t1)*h+0.5E0*(-6.0*t5+0.5E0+0.15E1*t1)*t9+0.1666667
+E0*(-6.0*t12+0.6E1*t5+0.75E0+0.1125E2*t1+15.0*t16-300.0*t20)*t9*h+
+0.4166667E-1*(0.12E2*t12+0.435E2*t5+108.0*x2*t1*x1-900.0*x2*t19-0.
+875E0-0.11625E2*t1-0.15E2*t16+0.3E3*t20+0.1884945E4*x1*sin(t18))*t
+40
 g3dyhxh = t43
```

Programs generated by VECFLD3 and used for Case Study # 12 calculation

ag3xh.for	$[x_1(h)]$
ag3yh.for	$[x_2(h)]$
ag3dxh.for	$[\frac{\partial \mathbf{G}_1}{\partial x_1}(x_1, x_2, x_3, h)]$
ag3dyh.for	$[\frac{\partial \mathbf{G}_2}{\partial x_2}(x_1, x_2, x_3, h)]$
ag3dxhyh.for	$[\frac{\partial \mathbf{G}_1}{\partial x_2}(x_1, x_2, x_3, h)]$
ag3dyhxh.for	$[\frac{\partial \mathbf{G}_2}{\partial x_1}(x_1, x_2, x_3, h)]$

ag3xh.for

```
 t4 = x1**2
 t5 = t4*x2
 t7 = 0.628315E1*x3
 t8 = sin(t7)
 t11 = h**2
 t15 = x2**2
 t16 = x1*t15
 t21 = t4*x1
 t23 = t4**2
 t24 = t23*x2
 t26 = t4*t8
 t28 = cos(t7)
 t62 = t11**2
 t65 = x1+x2*h+0.5E0*(-x1+0.2E0*x2-0.2E0*t5+50.0*t8)*t11+0.1666667E
+0*(-0.96E0*x2-0.4E0*t16-0.2E0*x1-0.8E-1*t5+0.1E2*t8+0.2E0*t21+0.4E
+-1*t24-0.1E2*t26+0.3141575E3*t28)*t11*h+0.4166667E-1*(-0.4E0*t15*x
+2-0.392E0*x2-0.32E0*t16+0.1576E1*t5+0.32E0*t21*t15-0.6E2*x2*x1*t8+
+0.96E0*x1-0.2021899E4*t8+0.8E-1*t21+0.24E-1*t24-0.4E1*t26-0.4E-1*t
+23*x1-0.8E-2*t23*t4*x2+0.2E1*t23*t8+0.628315E2*t28-0.628315E2*t4*t
+28)*t62
 ag3xh = t65
```

ag3yh.for

```
  t3 = x1**2
  t4 = t3*x2
  t6 = 0.628315E1*x3
  t7 = sin(t6)
  t12 = x2**2
  t13 = x1*t12
  t18 = t3*x1
  t20 = t3**2
  t21 = t20*x2
  t23 = t3*t7
  t25 = cos(t6)
  t28 = h**2
  t31 = t12*x2
  t36 = t18*t12
  t39 = x2*x1*t7
  t47 = t20*x1
  t49 = t20*t3
  t50 = t49*x2
  t52 = t20*t7
  t55 = t3*t25
  t96 = t20**2
  t101 = t7**2
  t109 = t28**2
  t112 = x2+(-x1+0.2E0*x2-0.2E0*t4+50.0*t7)*h+0.5E0*(-0.96E0*x2-0.4E
+0*t13-0.2E0*x1-0.8E-1*t4+0.1E2*t7+0.2E0*t18+0.4E-1*t21-0.1E2*t23+0
+.3141575E3*t25)*t28+0.1666667E0*(-0.4E0*t31-0.392E0*x2-0.32E0*t13+
+0.1576E1*t4+0.32E0*t36-0.6E2*t39+0.96E0*x1-0.2021899E4*t7+0.8E-1*t
+18+0.24E-1*t21-0.4E1*t23-0.4E-1*t47-0.8E-2*t50+0.2E1*t52+0.628315E
+2*t25-0.628315E2*t55)*t28*h+0.4166667E-1*(-0.502652E3*x2*x1*t25+0.
+392E0*x1+0.8816E0*x2-0.176E0*t47*t12-0.52E2*t39+0.4224E1*t13+0.352
+E0*t36+0.12E1*t3*t31+0.12736E1*t4-0.4143797E3*t7-0.56E0*t31-0.1576
+E1*t18+0.52E2*x2*t18*t7-0.11504E1*t21-0.1270389E5*t25+0.5335797E3*
+t23+0.8E-2*t20*t18-0.24E-1*t47-0.64E-2*t50+0.12E1*t52-0.251326E2*t
+55-0.4E0*t49*t7+0.125663E2*t20*t25+0.16E-2*t96*x2-0.12E3*t12*t7-0.
+3E4*x1*t101)*t109
 ag3yh = t112
```

ag3dxh.for

```
 t1 = x1*x2
 t4 = h**2
 t7 = x2**2
 t10 = x1**2
 t12 = t10*x1
 t13 = t12*x2
 t15 = 0.628315E1*x3
 t16 = sin(t15)
 t17 = x1*t16
 t33 = t10**2
 t45 = t4**2
 t48 = 1.0+0.5E0*(-0.1E1-0.4E0*t1)*t4+0.1666667E0*(-0.4E0*t7-0.2E0-
+0.16E0*t1+0.6E0*t10+0.16E0*t13-0.2E2*t17)*t4*h+0.4166667E-1*(-0.32
+E0*t7+0.3152E1*t1+0.96E0*t10*t7-0.6E2*x2*t16+0.96E0+0.24E0*t10+0.9
+6E-1*t13-0.8E1*t17-0.2E0*t33-0.48E-1*t33*x1*x2+0.8E1*t12*t16-0.125
+663E3*x1*cos(t15))*t45
 ag3dxh = t48
```

ag3dyh.for

```
 t1 = x1**2
 t5 = x1*x2
 t8 = t1**2
 t11 = h**2
 t14 = x2**2
 t18 = t1*x1
 t19 = t18*x2
 t21 = 0.628315E1*x3
 t22 = sin(t21)
 t23 = x1*t22
 t26 = t8*t1
 t50 = t8**2
 t56 = t11**2
 t59 = 1.0+(0.2E0-0.2E0*t1)*h+0.5E0*(-0.96E0-0.8E0*t5-0.8E-1*t1+0.4
+E-1*t8)*t11+0.1666667E0*(-0.12E1*t14-0.392E0-0.64E0*t5+0.1576E1*t1
++0.64E0*t19-0.6E2*t23+0.24E-1*t8-0.8E-2*t26)*t11*h+0.4166667E-1*(-
+0.52E2*t23+0.52E2*t18*t22+0.12736E1*t1+0.8816E0+0.8448E1*t5-0.1150
+4E1*t8+0.704E0*t19-0.64E-2*t26-0.168E1*t14-0.502652E3*x1*cos(t21)+
+0.36E1*t1*t14-0.352E0*t8*x1*x2+0.16E-2*t50-0.24E3*x2*t22)*t56
 ag3dyh = t59
```

ag3dxhyh.for

```
 t1 = x1**2
 t4 = h**2
 t7 = x1*x2
 t10 = t1**2
 t16 = x2**2
 t31 = t4**2
 t34 = h+0.5E0*(0.2E0-0.2E0*t1)*t4+0.1666667E0*(-0.96E0-0.8E0*t7-0.
+8E-1*t1+0.4E-1*t10)*t4*h+0.4166667E-1*(-0.12E1*t16-0.392E0-0.64E0*
+t7+0.1576E1*t1+0.64E0*t1*x1*x2-0.6E2*x1*sin(0.628315E1*x3)+0.24E-1
+*t10-0.8E-2*t10*t1)*t31
 ag3dxhyh = t34
```

ag3dyhxh.for

```
      t1 = x1*x2
      t5 = x2**2
      t8 = x1**2
      t10 = t8*x1
      t11 = t10*x2
      t13 = 0.628315E1*x3
      t14 = sin(t13)
      t15 = x1*t14
      t18 = h**2
      t23 = t8*t5
      t25 = x2*t14
      t31 = t8**2
      t33 = t31*x1
      t34 = t33*x2
      t36 = t10*t14
      t38 = cos(t13)
      t39 = x1*t38
      t76 = t14**2
      t83 = t18**2
      t86 = (-0.1E1-0.4E0*t1)*h+0.5E0*(-0.4E0*t5-0.2E0-0.16E0*t1+0.6E0*t
     +8+0.16E0*t11-0.2E2*t15)*t18+0.1666667E0*(-0.32E0*t5+0.3152E1*t1+0.
     +96E0*t23-0.6E2*t25+0.96E0+0.24E0*t8+0.96E-1*t11-0.8E1*t15-0.2E0*t3
     +1-0.48E-1*t34+0.8E1*t36-0.125663E3*t39)*t18*h+0.4166667E-1*(-0.52E
     +2*t25-0.502652E2*t39+0.156E3*x2*t8*t14+0.502652E2*t10*t38+0.25472E
     +1*t1+0.392E0+0.1067159E4*t15+0.4224E1*t5-0.46016E1*t11-0.4728E1*t8
     ++0.1056E1*t23-0.384E-1*t34+0.48E1*t36-0.12E0*t31-0.502652E3*x2*t38
     ++0.56E-1*t31*t8+0.24E1*x1*t5*x2-0.88E0*t31*t5+0.128E-1*t31*t10*x2-
     +0.3E4*t76-0.24E1*t33*t14)*t83
      ag3dyhxh = t86
```