PSX : A Program that Explores Phase Portraits of Two-dimensional Piecewise Linear Differential Equations

NISHIDA, Toyoaki; DOSHITA, Shuji

PSX: A Program that Explores Phase Portraits of Two-dimensional Piecewise Linear Differential Equations

By

Toyoaki NISHIDA and Shuji DOSHITA

(Received June 30, 1990)

Abstract

A set of all solution curves in the phase space for ordinary differential equations is called the phase portrait. Phase portraits provide global qualitative information about how dynamical systems behave under different initial conditions.

In this paper, we present a program called PSX that automatically analyzes topological structure of phase portraits for two dimensional piecewise linear differential equations. PSX has several novel features that have not been achieved before: (a) PSX possesses procedures for recognizing instances of abstract concepts defined in dynamical systems theory. PSX does not only build a memory structure for instances of abstract concepts but it also directs the search process and constructively proves that what it has found is in fact an instance of the concept. (b) Though limited to two-dimensional phase spaces, PSX is applicable to complex flows in Non-Euclidean phase spaces as well as those in Euclidean phase spaces. (c) The architecture of PSX enables qualitative analysis and quantitative analysis to interact in a cooperative manner. Qualitative analysis guides the overall analysis process. When it gets stuck due to ambiguity, qualitative analysis submits a question to the quantitative analysis and the quantitative analysis provides an answer.

PSX has been implemented using Common Lisp and tested against several examples.

1. Introduction

Analysis of nonlinear differential equations is inherently difficult. No complete analytical method is known by which an explicit form of solution is produced. Hence, a numerical method is usually called for to draw approximate information. Various tools based on sophisticated numerical algorithms have been developed and put to practical use. However, it should be noted that those numerical tools are passive since those tools do not solve any problem by themselves. In order to draw potential power of those tools,

Department of Information Science, Kyoto University Sakyo-ku, Kyoto 606, Japan

human experts have to carefully prepare plans for analysis and interpret the results.

The goal of this resecrch is to build a program which autonomously explores the behavior of nonlinear dynamical systems by simulating the intellectual process carried out by human experts. In order to plan what to analyze and interpret the result of analysis, the program has to possess abstract concepts concerning the behavior of dynamical systems such as equilibrium state or periodic solution. In addition, it is widely believed that planning and interpretation are driven by qualitative understanding of behavior.

As a first step towards the goal, we have developed a program PSX[1] which takes as input two-dimensional piecewise linear differential equations and produces as output qualitative description of all possible behaviors.

PSX has several novel features that have not been achieved before: (a) PSX posseses procedures for recognizing instances of abstract concept defined in the dynamical systems theory. PSX does not only build a memory structure for instances of abstract concepts but it also directs the search process and constructively proves that what it has found is in fact an instance of the concept. (b) Though limited to two-dimensional phase spaces, PSX is applicable to complex flows in Non-Euclidean phase spaces as well as those in Euclidean phase spaces. (c) The architecture of PSX enables qualitative analysis and quantitative analysis to interact in a cooperative manner. Qualitative analysis guides the overall analysis process. When it gets stuck due to ambiguity, qualitative analysis submits a question to the quantitative analysis and the quantitative analysis provides an answer.

2. Glimpse of PSX

The input to PSX is two-dimensional piecewise linear differential equations of form:

$$\{R_i : \langle p_i(x) \mid \dot{x}=f_i(x)\rangle\} \qquad 1\leq i\leq m,\ x\in \mathrm{R}^2 \tag{1}$$

where, $p_i(x)$ is a conjunction of linear inequalities which specifies the range of linear region R_i, and $f_i(x)$ is a linear formula of x which specifies the flow in the linear region R_i.

PSX has an inventory of basic concepts for understanding a phase portrait, a collection of all solution curves. For example, PSX 'understands' attracting limit cycles[2] in the sense that it has a procedure to recognize an instance of attracting limit cycles from the phase portrait specified by (1).

[1] PSX stands for *Phase Space eXplorer.*

[2] Informally, an attracting limit cycle is a solution curve correspoding to a closed loop in the phase spece which nearby solution curves approach as $t \to \infty$. For formal definition, see textbooks of dynamical systems theory such as [3].

Consider, for example, a Van der Pol's equation:

$$\begin{cases} \dot{x}=\dfrac{y-(x^3-x)}{c} \\ \dot{y}=-x \qquad\qquad 0<c<1 \end{cases} \tag{2}$$

By approximating the nonlinear term of the first formula by a collection of three linear terms, we obtain a piecewise linear differential equation:

$$\begin{Bmatrix} \mathrm{R}_+ : \left\langle \dfrac{1}{2}\leq x \mid \dot{x}=\dfrac{-2x+y+2}{c},\ \dot{y}=-x \right\rangle \\ \mathrm{R}_0 : \left\langle -\dfrac{1}{2}\leq x\leq \dfrac{1}{2} \mid \dot{x}=\dfrac{2x+y}{c},\ \dot{y}=-x \right\rangle \\ R_- : \left\langle x\leq -\dfrac{1}{2} \mid \dot{x}=\dfrac{-2x+y-2}{c},\ \dot{y}=-x \right\rangle \end{Bmatrix} \tag{3}$$

```
1. asymptotic destinations:
   ((ATTRACTING-BOO-3 (ENTRANCE. I-71)
        (HULL (DEPARTS I-33 LC1 LC1-MINF L-17)
              (DEPARTS I-58 LC1 L-18 L-11)
              (ARRIVES I-60 LC2 L-25 L-14)
              (DEPARTS I-63 LC2 L-25 L-14)
              (ARRIVES I-66 LC2 L-16 L-27)
              (ARRIVES I-71 LC1 L-28 L-29))))
2. local coordinates: LC1: base: (-1/2 0), orientation: (0 1)
                                   ...
3. intervals:
   (I-71 (LOCAL-COORDINATE. LC1) (FROM. L-28) (TO. L-29) (DEPTH. 5)
        (BACKWARD-MAPPERS (BOO-INTERVAL-11 I-68))
        (MAPPED-FROM (NIL I-68 (NIL I-63 (NIL I-58 (NIL I-33)))))
        (LOCKED. T) (LOCKED-DUE-TO. ENTRANCE-TO-ATTRACTING-BOO)
        (ATTRACTING-BOO. ATTRACTING-BOO-3) (SUCCESSOR-IS. I-122))
                                   ...
4. landmarks:
   (L-28 (BELONGS-TO. LC1) (LOWER-BOUND. MINF)
        (UPPER-BOUND. 0.1464466S0) (TYPE. INDEFINITE))
   (L-29 (BELONGS-TO. LC1) (LOWER-BOUND. MINF)
        (UPPER-BOUND. 0.1464466S0) (TYPE. INDEFINITE))
   (L-6 (BELONGS-TO. LC1) (LOWER-BOUND. 0.1464466S0)
       (UPPER-BOUND. 0.1464466S0) (TYPE. DEFINITE))
                                   ...
5. landmark-ordering:
   (LC1 LC1-MINF L-31 L-30 L-78 L-28 L-29 L-68 L-69 L-32 L-35 L-36 L-17 L-60
        L-61 L-4 L-6 L-7 L-9 L-20 L-21 L-19 L-62 L-63 L-18 L-37 L-38 L-39 L-70
        L-71 L-40 L-43 L-79 L-42 L-41 L-11 L-12 LC1-INF)
                                   ...
```

Figure 1: Portion of Memory Structure PSX Constructed For a Piecewise Linear Approximation (3) of Van der Pol's Equation (2)

Figure 1 illustrates what PSX has created in memory after analyzing (3). The memory structure explicitly indicates that there exists an attracting bundle of orbits in the phase portrait. The approximate location of the attracting bundle of orbits can be seen from the 'HULL' attribute which indicates the sequence of intervals {I-33, I-58, I-60, I-63, I-66, I-71} it passes transversely. Each interval, in turn, is indicated by a couple of landmarks that delimit the interval. Location of some landmarks such as L-6 is given as numerical values, while it is not so for others such as L-28 and L-29, though at least the total ordering of landmarks is given for each local coordinate. Although PSX does not always produce quantitative information about what it has found, PSX does construct a logical specification of its findings in memory, which enables a numerical method to compute numerical values from those specifications[3]. What is emphasized here is that it is qualitative analysis that guides quantitative methods on what to compute and that interprets what is computed.

3. Dynamical Systems Theory

3. 1 The Framework

PSX is based on the *Dynamical Systems Theory* (DST for short) [3, 2]. This enables, for example, to draw qualitative information about solution, even though a precise form of solution is not available.

Consider an ordinary differential equation

$$\dot{x}=f(x) \tag{4}$$

where, x is a vector of state variables $(x_1, \cdots\cdots, x_n)$ each of which gives some value in R as a function of $t\in \mathrm{R}$. An n-dimensional space spanned by $\{x_i\}$ is called the phase space. For each point $c=(c_1, \cdots\cdots, c_n)$ in the phase space, formula (4) specifies the rate and the orientation of state change: $\dot{x}|_{x=c}=\left(\frac{dx_1}{dt}, \cdots\cdots, \frac{dx_n}{dt}\right)|_{x=c}$. In other words, formula (4) defines a vector field in the phase space. A specific solution corresponding to the initial state $a=(a_1, \cdots\cdots, a_n)$ is a curve such that it passes on the point corresponding to a in the phase space and it is tangential to the vectors specified bv the vector field at each point. Such a curve is called a solution curve, a trajectory, or an orbit. Theoretically, we can think of an orbit passing on $x\in U\subset \mathrm{R}^n$ as a mapping $\phi_x(t) : \mathrm{R}\rightarrow V$ which maps x to $y\in V\subset \mathrm{R}^n$ as a function of $t\in \mathrm{R}$. We can also think of a differential equation (4) as specifying a "flow" $\Phi(x, t)=\phi_x(t) : U\times \mathrm{R}\rightarrow V$. If uniqueness of solution to differential equation holds, orbits never intersect with others nor with themselves (the non-intersection constraint). The collection of all orbits in the phase space is called the phase

[3] This feature has not been implemented.

portrait.

There is obvious correspondence between geometric properties of orbits and aspects of dynamical behavior. For example, points in the phase space at which the right-hand side of the formula (4) is *zero* are called fixed points, and they correspond to equilibrium states which will not evolve for ever. Fixed points are further classified into sinks, sources, saddles, and so on. Orbits near a sink arbitrarily approach the sink as $t \rightarrow \infty$, while orbits near a source do so when $t \rightarrow -\infty$. Orbits near saddles are first attracted to and then repelled from the saddles[4]. Closed orbits correspond to periodic behaviors.

DST is particularly interested in asymptotic behaviors of dynamical systems as $t \rightarrow \pm\infty$. In DST, it is proved that solution curves in two dimensional phase space either (1) diverge for place at infinity, (2) approach a fixed point, or (3) approach a closed orbit (called a limit cycle), as $t \rightarrow \pm\infty$. Informally, an orbit which attracts nearby orbits as $t \rightarrow \infty$ is called an attractor. Similarly, an orbit which attracts nearby orbits as $t \rightarrow -\infty$ is called a repbllor. Attractors and repellors play an important role in qualitative analysis of phase portrait. $\{y \mid \exists\, t_n(\rightarrow\infty)\,[\lim_{n\rightarrow\infty}\phi_{t_n}(x)=y]\}$, a set of points which an orbit passing a point x in the phase space approachs as $t \rightarrow \infty$, is called an ω-limit set of x. Although an ω-limit set becomes empty if the orbit diverges for a place at infinity as $t \rightarrow \infty$, we regard a place at infinity as a special kind of place and allow an ω-limit set to contain a place at infinity. An ω-limit set extended in this way is called an asymptotic destination. Similar extension to an α-limit set defined in the case $t \rightarrow \infty$ is called an asymptotic source.

3. 2 Two Dimensional Piecewise Linear Equations and their Properties

PSX takes as input a normal form of piecewise linear differential equations:

$$\{R_i : \langle p_i(x) \mid \dot{x}=f_i(x)\rangle\} \qquad 1\leq i\leq m,\ x\in \mathrm{R}^2 \tag{5}$$

Constituents delimited by angular brackets are called linear regions. Each linear region may be given a unique label R_i. $p_i(x)$ given as a logical combination of linear inequalities specifies a condition for the linear approximation R_i to be effective. $f_i(x)$ given as a linear formula of $x\in\mathrm{R}^2$ defines the local flow in linear region R_i.

If the following two conditions:

1. the collection of linear regions covers R^2 without overlap[5], and
2. the vector flows of adjacent linear regions are identical at both sides of the boundary

hold, then the solution is uniquely determined, the phase space is Euclidean (*i. e.*, R^2), and the flow is C^1 (*i. e.*, the curve itself and its derivative are continuous). In other words, if the two conditions hold, whenever an orbit leaves (enters) a linear region, the

[4] These are not definitions but properties. For definition see [3] for example.

[5] Boundaries of linear regions are allowed to overlap.

next (previous) linear region is uniquely determined and transition at the boundary of linear regions is continuous.

If the above two conditions do not hold, the behavior of differential equation may not be well-defined; the state transition may run into undefined status, cause several variables to change discontinuously, or fail to specify the next state uniquely. Our algorithm presented in this paper can handle both cases, though the algorithm becomes faster if we can assume uniqueness of the solution.

Two dimensional linear differential equations dominating each linear region are written as:

$$\dot{x}=Ax+b$$

where,

$$\dot{x}=\begin{bmatrix} x_1 \\ x_2 \end{bmatrix},\ A=\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix},\ b=\begin{bmatrix} b_1 \\ b_2 \end{bmatrix}.$$

As is obvious from the above, linear systems have only one fixed point at $-A^{-1}b \in \mathrm{R}^2$ when $det(A) \neq 0$[6]. Then, by translating the coordinate to the fixed point by $v \leftarrow x+A^{-1}b$, the above formula can be rewritten as

$$\dot{v}=Av.$$

Property of two-dimensional linear flows is simple. It can be classified into several categories by computing eigenvalues and eigenvectors of coefficient matrix A. In particular, we can grasp the asymptotic behavior of orbits from eigenvalues and eigenvectors. When eigenvalues are real, the flow is usually[7] divided into four independent subflows by two invariant manifolds[8] spanned by the eigenvectors. When eigenvalues are complex (*i. e.* $a \pm \mathrm{i} \cdot b$ where $b \neq 0$), the flow can be categorized into three subclasses: spiral sink, center, and spiral source, depending on the sign of the real part a. If a is negative, then the flow is called a spiral sink, if *zero* then a center, and if positive then a spiral source. The orbit is left-turning, if the coefficient of the imaginary part b is positive; right-turning, if it is negative.

Orbits of linear flow do not have any point of inflection. Hence, each orbit other than a fixed point is either right-turning, left-turning, or straight. The way a given orbit turns can determined by examining the sign of

$$[\dot{x} \times \ddot{x}]_z = \dot{x}_1 \cdot \ddot{x}_2 - \ddot{x}_1 \cdot \dot{x}_2$$

at some point on it. The orbit is right-turning if the sign is negative; straight if the sign

[6] Since this is almost always the case, we assume this throughout this paper.

[7] as far as the two eigenvalues are different from each other

[8] One dimensional region (*i. e.*, straight line) spanned by eigenvectors originated at the fixed point. Orbits on invariant manifold are isolated from orbits elsewhere; orbits on an invariant manifold never leave the manifold and orbits elsewhere never come in the manifold.

is *zero*; and left-turning if the sign is positive.

4. Reasoning about Phase Portraits

We need to resolve several engineering problems to make DST computational. In particular, it is crucial to represent orbits so that reasoning about the phase portrait can be made effectively. PSX is based on the integration of several novel ideas.

- we focus on bundle of orbits rather than single orbits, thereby deriving useful conclusions which cannot be made about single orbits.
- we abstract each bundle of orbits a sequence of mappings between locally defined $(n-1)$-dimensional hyperplanes. This enables us to make qualitatively important distinctions about bundle of orbits, without committing ourselves to the hard problem of inventing a general framework for representing gemetric objects.
- we can grasp global behavior by examining the structure of mappings corresponding to bundles of orbits.
- strong constraints of two-dimensional phase portraits enable us to generate mappings representing flows. Qualitative reasoning with a simple class of numerical computation allows us to enumerate all possible mappings. Sometimes, flow can be uniquely translated into mappings. Even if this not the case, the qualitative analysis identifies what should be computed by a more complex class of numerical computation to resolve ambiguity.

In the description below, we attempt to generalize these ideas to n-dimensional phase space as far as possible, for most of them are applicable to n-dimensional phase space.

4. 1 Reasoning about Bundle of Orbits

By examining the properties of bundles of orbits, we can derive conclusions that connot be derived for individual orbits. Suppose, for example, we have found in the given phase portrait a pattern as shown in Figure 2, in which bundle of orbits ϕ is transverse to regions I and J on the same hypersurface such that $J\subset I$. This pattern is

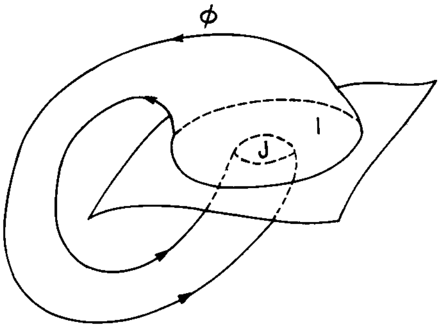


Figure 2: Contracting Recursive Bundle of Orbit Intervals

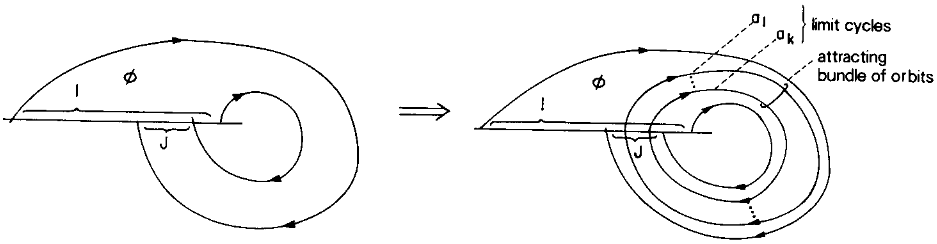


Figure 3: Circumstance in which Existence of Attracting Limit Cycle is Predicted

called a contracting recursive bundle of orbit intervals, and all orbits transverse to I are also transverse to J, and never leave the region occupied by ϕ. If the phase space is two-dimensional, region J is finitely bounded, and region I and J do not share boundary, then ϕ contains one or more limit cycle $\{a_1, \cdots\cdots, a_k\}$ as shown in Figure 3. In addition, all orbits transverse to $I-J$ approach one of those limit cycles as $t \to \infty$.

In order to make this approach computationally feasible, we start from smaller pieces called bundles of orbit intervals. A bundle of orbit intervals is defined as a set of intervals of orbits ϕ such that

1. no fixed point is involved in the interior of the region occupied by ϕ
2. for any point p in the interior of the region occupied by ϕ, there exists a hypersurface s which is transverse to the region occupied by ϕ at region a that contains p and
   (a) all intervals of orbits that belong to ϕ are transverse to s once and only once.
   (b) for any point q in a, there exists an interval of orbit involved in ϕ which is

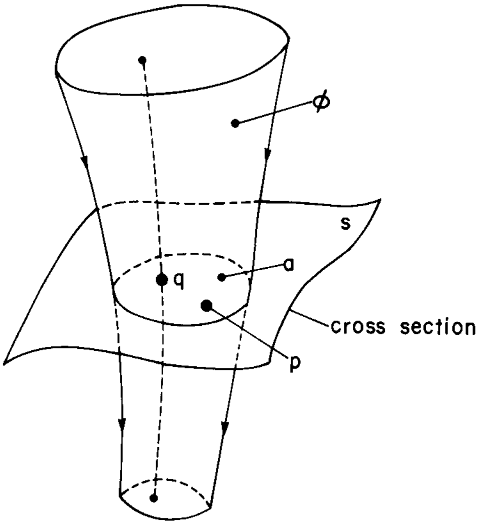


Figure 4: Bundle of Orbit Intervals and its Cross Section

transverse to s at q.

The hypersurcace s in the above definition is called a cross section of ϕ. Intuitively, the first condition and the former half of the second require the uniformness of intervals involved in ϕ, and the latter half of the second condition requests the density and continuity of orbit intervals in ϕ. See Figure 4. In local analysis, a collection of bundles of orbit intervals is generated and their property is examined individually. In global analysis, the result of local analysis is put together and global information is derived.

The remaining problem is how to find such recursive bundle of orbit intervals as above and how to represent it. We will address the problem in the following subsections.

4. 2 Representing Bundle of Orbits as Mapping between Hyperplanes

We introduce several $(n-1)$ -dimensional hyperplanes, called sampling hyperplanes, in the n-dimensional phase space to "sample" data about bundles of orbit intervals. When $n=2$, the sampling hyperplanes are straight lines, which we call sampling lines.

Consider a bundle of orbit intervals ϕ that intersects sampling hyperplanes p_1 and p_2 at r_1 and r_2, respectively (see Figure 5). An interval ϕ_0 of ϕ delimited by r_1 and r_2 continuously maps points of r_1 to r_2. We represent this as $\phi_0 : r_1 \rightarrow r_2$.

Although it is hard to find an effective representation for r_1 and r_2 in n-dimensional space, we can currently put aside this problem since we limit our concern to the two-dimensional phase space. In the two-dimensional phase space, r_1 and r_2 are line segments which can be completely specified by the location of the two end points.

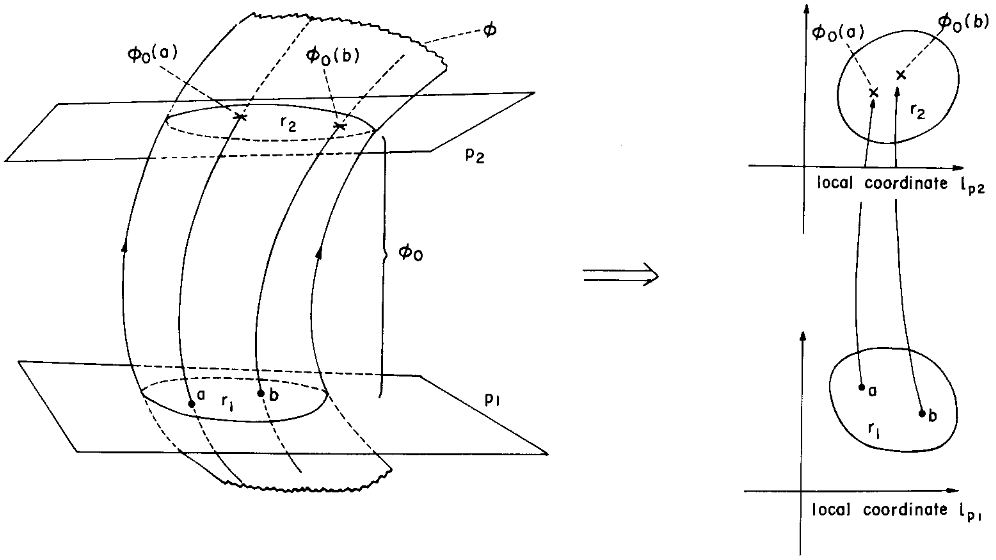


Figure 5 : Representing Bundle of Orbits as Mapping between Hyperplanes

The sampling line l_k is represented as a pair $\langle b, o\rangle$, where $b=(b_1, b_2)\in R^2$, called the base, is a point on l_k, and $o=(o_1, o_2)\in R^2$ is the orientation of l_k. By this convention, the location x of a point p on l_k can be represented as '$x=(b_1, b_3)+u\cdot(o_1, o_2)$'. We call this system of locating points local coordinate c_k associated with l_k, and we use a denotation '$loc(p)=u$ wrt c_k' which reads the location of p is u with respect to c_k. Likewise we mean by '(u, v) wrt c_k' an open interval $\overline{pq}$ bounded by two points p and q, such that $loc(p)=u$ wrt c_k and $loc(q)=v$ wrt c_k. If $loc(p)=u$ wrt c_k and $loc(q)=v$ wrt c_k and $u<v$ for two points p and q on l_k, we say that p is smaller than q with respect to c_k, and denote it as $p\prec_{c_k}q$.

The sampling line $l_k=\langle b, o\rangle$ divides the phase space into two disjoint regions : the right side defined as $\{x\mid D_{l_k}(x)>0\}$, where

$$\begin{aligned}D_{l_k}(x)&=[o\times(x-b)]_z\\&=o_1\cdot(x_2-b_2)-o_2\cdot(x_1-b_1).\end{aligned}$$

In general, points on the sampling line that delimits intervals are called landmarks. It should be noted that the exact address of landmarks is not necessary for qualitative analysis. Instead, only the total ordering of landmarks on a sampling line is needed. In two-dimensional phase spaces, mapping ϕ_0 is order-preserving, namely the order of landmarks is preserved in the mapping from r_1 to r_2.

4. 3 Grasping Global Behavior By Examining the Structure of Mappings for Bundle of Orbits

We can grasp the global characteristics of phase portraits by constructing in turn compositions of mappings representing bundle of orbit intervals. In particular, asymptotic behavior of orbits can be captured by constructing a finite number of compositions.

Formally, given a couple of bundles of orbit intervals $\phi_1: I\rightarrow J$ and $\phi_2: J\rightarrow K$, composition $\phi_2\circ\phi_1$ of ϕ_1 and ϕ_2 is defined as follows:

$$\phi_2\circ\phi_1(x)=\phi_2(\phi_1(x))=y \text{ iff } \exists z[\phi_1(x)=z,\ \phi_2(z)=y]$$

$\phi_2\circ\phi_1$ is continuous if ϕ_1 and ϕ_2 are continuous. $\phi_2\circ\phi_1$ is order-preserving if ϕ_1 and ϕ_2 are order-preserving. Mapping $\phi_m\circ\cdots\circ\phi_1$ is called a contracting recursive mapping, if its range is a subset of its domain, namely,

$$\phi_m\circ\cdots\circ\phi_1(I)\subset I.$$

Similarly, $\phi_m\circ\cdots\circ\phi_1$ is called an extending recursive mapping, if

$$\phi_m\circ\cdots\circ\phi_1(I)\supset I.$$

If a contracting recursive mapping is found, it entails the existence of an attracting bundle of orbits. Similarly, the existence of an extending recursive mapping entails the existence of a repelling bundle of orbits.

Since the algorithm underlying PSX produces only a finite number of primitive mappings for the bundle of orbit intervals, PSX can identify all possible asymptotic sources and destinations in the phase space. Furthermore, PSX classify the bundle of orbits into those comprising a region containing an asymptotic source or destination and others which comprise transient states. For transient regions, PSX can identify the asymptotic source and destination which the orbits passing there approach as $t \rightarrow -\infty$ and $t \rightarrow \infty$, respectively. In sum, PSX can draw a qualitative map of the phase portrait. All this can be done by a finite amount of computation, though it should be noted that the result may often ambiguous and an extensive numerical computation, though quite limited in frequency, is needed to resolve ambiguity.

4. 4 Generating Mappings by Local Analysis

In order for an approach based on the about idea to be computationally feasible, we should be able to develop

- ● a procedure for finding sampling hyperplanes that are useful in grasping global behavior and
- ● a procedure for abstracting flow as mapping between sampling hyperplanes.

These requirements are not easy to achieve in general, even for two dimensional flows. However, as we will show in the next section, we can in fact meet the above requirements if we further constrain the input to two-dimensional piecewise linear differential equations. Our method is

- ● to divide the phase space into convex regions called cells by invariant manifolds and boundaries between linear regions
- ● to constrain the local flow in each cell into a finite number of candidates, by examining the property of flow on the boundary of the cell. Only a finite number of computation consisting of four basic arithmetic and square root is required for this. Although a more complex class of computation is needed to resolve ambiguity, qualitative analysis uncovers what to compute.

5. Properties of Local Flow

In this section, we examine properties of local flow in cells. Formally, let us call a convex region of the phase space a cell, if it is bounded by polygon or a polyline, it has no fixed point or invariant manifold in its interior region, and it is dominated by a single linear flow. Fixed points are allowed to be only on the boundary of cells. Thus, all orbits in a cell come from somewhere at boundary and leave for somewhere at boundary either in a finite or infinite amount of time. Geometrically, a cell consists of the interior region and the boundary. Each line segment comprising the boundary is called a boundary edge. Each couple of adjacent boundary edges is delimited by a vertex. A cell is either closed or open.

5. 1 Flow at the Boundary of Cells

Let c be a cell with local flow specified by a linear differential equation $\dot{x}=Ax+c$, p

be a point on the boundary of c that is not a vertex, $l_k : \langle b, o\rangle = \langle (b_1, b_2), (o_1, o_2)\rangle$ be a sampling line passing p, and c_k be a local coordinate associated with l_k. We can see how flow in c intersects with l_k, by evaluating the z-component of the cross product $o \times \dot{x}$:

$$\begin{aligned}
[o \times \dot{x}]_z &= [o \times (Ax+c)]_z \\
&= o_1 \cdot (a_{21}(b_1+u \cdot o_1)+a_{22}(b_2+u \cdot o_2)+c_2) \\
&\quad - o_2 \cdot (a_{11}(b_1+u \cdot o_1)+a_{12}(b_2+u \cdot o_2)+c_1) \\
&= (o_1^2 \cdot a_{21}+(a_{22}-a_{11})o_1o_2-a_{12}o_2^2)u \\
&\quad + ((a_{21}b_1+a_{22}b_2+c_2)o_1-(a_{11}b_1+a_{12}b_2+c_1)o_2) \\
&\equiv pu+q \qquad (6)
\end{aligned}$$

where,

$$p = o_1^2 \cdot a_{21}+(a_{22}-a_{11})o_1o_2-a_{12}o_2^2$$
$$q = (a_{21}b_1+a_{22}b_2+c_2)o_1-(a_{11}b_1+a_{12}b_2+c_1)o_2$$

We say that the flow is transverse-right (transverse-left) to the sampling line l if the sign of $[o \times \dot{x}]_z$ is negative (positive). The flow is inward if the interior of the cell is in the right side of the sampling line and the flow is transverse-right to the sampling line or if the interior of the cell is in the left side of the sampling line and the flow is transverse-left to the sampling line. Such a point on the boundary is called an entrance to the cell. A source on the boundary is also regarded as an entrance. Maximally continuous segments of boundary at which orbits transversely enter the cell are called entrance segments. An entrance segment is a point on the boundary if it is a source; otherwise, it is an open polyline segment consisting of one or more boundary edges. Likewise we define exits and exit segments for outgoing flow. Exit segments include sinks located on the boundary as a special case.

Of particular importance are boundary sections at which flow is tangential to the

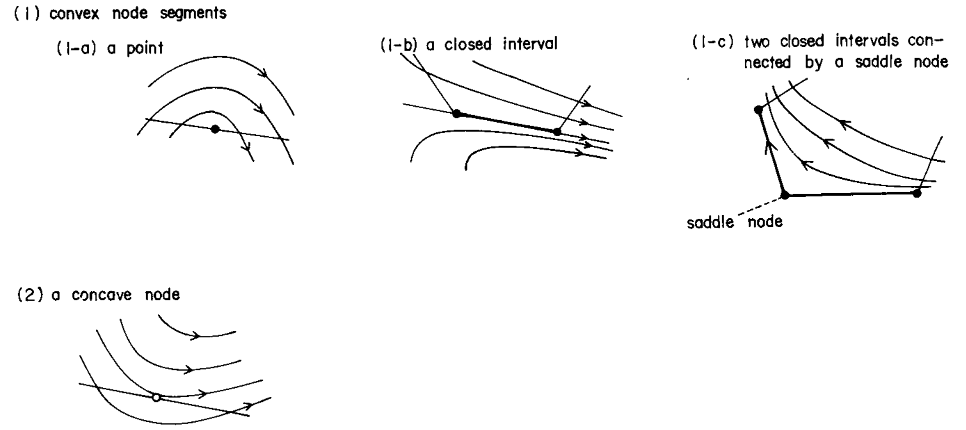


Figure 6 : Geometric Configuration of Singular Segments

boundary and no orbit leaves or enters. A point on the boundary of a cell is celled a singular node if the flow is tangential to the boundary at the point. A sigular segment is a maximal aggregation of continuous singular nodes. From the definition (6) there is only one singular point at $-\frac{q}{p}$ wrt l_k if $p \neq 0$. Singular segments are either an isolated point, a closed interval, or two closed intervals sharing a saddle node, as shown in Figure 6. In general, a singular segment is a polyline segment if the boundary edge is on an invariant manifold; otherwise it is an islated point. When a singular segment consists of a single point, we call it a singular node.

The orbit passing through a singular node lies in the same side of the boundary immediately before and after passing the singular segment, for orbits of linear flow do not have any point of inflection. If the orbit lies outside the boundary immediately before and after passing the singular segment, we call the singular segment a convex node segment, or a convex node if it consists of a single point. Due to the properties of linear flows, a singular segment other than convex node segments consists of a single point, which we call a concave node. By definition, an orbit passing a concave node lies inside of the cell just before and after passing it.

The type of singular segment is determined by the curvature of orbit at the singular segment, the side in which the orbit lies immediately before and after the visit, and the relationship between the orientation of the flow at the tangential point and that of the sampling line associated with the edge. Table 1 shows a set of rules for determining the type of a singular node when the orientation of the flow conforms to that of the edge at the tangential point. In what follows, entrance segments, exit segments and singular segments are generally referred to as boundary segments.

Table 1: Rules for Determining the Type of Singular Node — when the flow at the tangential point is oriented in the same direction as that of the sampling line associated with the edge

		which side of the bounday the cell exists	
		right side	left side
how the orbit turns	right-turning	concave node	convex node
	left-turning	convex node	concave node

Since liner flows have strong constaints as to the geometry of orbits, we can obtain useful clues about the flow of a cell, by a simple class of computation. In particular, a given cell has at most one concave node at the boundary, we can uniquely identify the flow pattern inside the cell. Even though it is not the case, the number of possibilities is finite and specification of a numerical computation for resolving ambiguity is synthesized from what is obtained by qualitative analysis.

Since the flow in closed cells is more regular than that in open cells, we first describe the properties of local flow in closed cells, then we extend the analysis to open cells.

5. 2 Properties of Local Flow in Closed Cells

Orbits in closed cells originate from entrance segments and tend towards exit segments. The following properties are important in identifying the flow pattern in a closed cell.

Property 1 *Let the number of convex node segments and concave nodes be n_v and n_c, respectively. Then,*

$$n_v = n_c + 2. \tag{7}$$

Property 2 *If at least one concave node is involved on the boundary of a cell, there exist a sequence of three consecutive convex node segments that have no concave node in-between. The center of such convex node sements is called the center segment.*

Property 3 *If a concave node c is involved in the boundary, then the orbit passing on c intersects each of the two sequences of the boundary segments between c and the center segment.*

Property 4 *The orbit passing on c does not intersect boundary segments adjacent to c.*

Let us call a sequence of all boundary segments ordered clockwise from the center segment a left-cyclic boundary segment list. Boundary segment *s* of a cell is said to be to the right (left) of boundary segment *p* of the same cell if *s* appears after (before) *p* in the left-cyclic boundary segment list. An orbit is said to pass through a cell from left to right if the boundary segment through which the orbit leaves the cell is to the right of the boundary segment through which the orbit enters the cell. Orbits that pass through a cell from right to left are similarly defined. The following property holds:

Property 5 *For all cells, one of the folloming conditions holds:*

- *all orbits that pass through the cell from left to right (left-right flow)*
- *all orbits that pass through the cell from right to left (right-left flow)*

It is easy to see whether a local flow is left-right or *vice versa*, For example, one might see the orientation of the flow at the boundary segment immediately to the right of the center segment.

Property 6 *It follows from the non-intersection constraint of orbits and property 3 that all orbits inside a cell are nested around the center segment.*

Now let us illustrate a couple of examples. Consider the flow in a cell shown in Figure 7. Since no concave node segment is involved in the boundary, the local flow inside this cell is uniquely characterized as inter-boundnry segment mapping : $\overline{cda} \rightarrow \overline{abc}$. In order to characterize the local flow to more detail as inter-boundary edge mapping, however, ambiguity arises unless more information is available. There are three possibi-

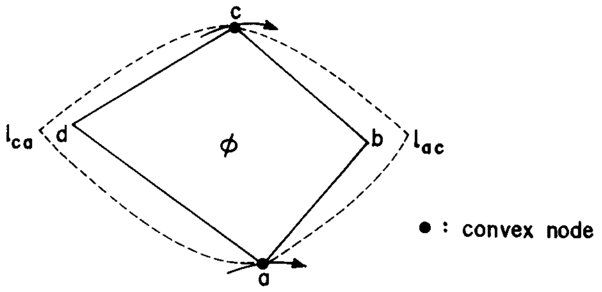


Figure 7: Example of Local Flow in a Closed Cell (1)

lities:

1. $\overline{da} \rightarrow \overline{a\phi(d)}$, $\overline{\phi^{-1}(b)d} \rightarrow \overline{\phi(d)b}$, $\overline{c\phi^{-1}(b)} \rightarrow \overline{bc}$
2. $\overline{da} \rightarrow \overline{ab}$, $\overline{cd} \rightarrow \overline{bc}$
3. $\overline{\phi^{-1}(b)a} \rightarrow \overline{ab}$, $\overline{d\phi^{-1}(b)} \rightarrow \overline{b\phi(d)}$, $\overline{cd} \rightarrow \overline{\phi(d)c}$

where, $\phi(d)$ is an image of d by ϕ, and $\phi^{-1}(b)$ is a point on the boundary that is mapped to b by ϕ. We can resolve ambiguity if we can determine, for example, which boundary entity the orbit passing on vertex d intersects on leave. In general, this requires an extensive numerical computation.

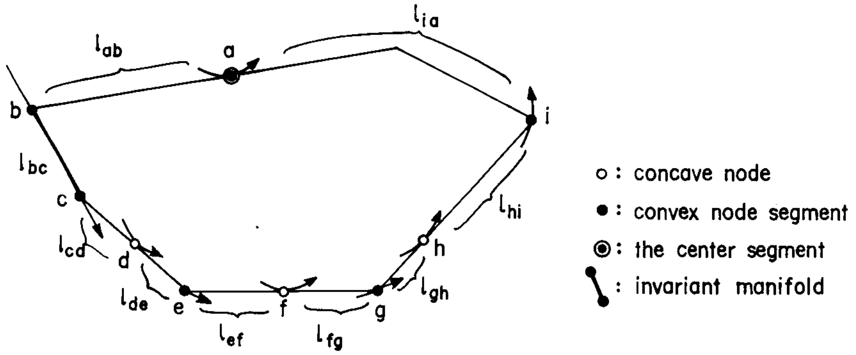


Figure 8: Example of Local Flow in a Closed Cell (2)

Consider the flow in another cell shown in Figure 8. Since this case contains three concave nodes on the boundary, we can think of eleven possible ways of qualitatively different patterns of local flow in this cell (at the inter-boundary segment level), as shown in Figure 9. In order to identify which is the case, one may call for extensive numerical computation to see which boundary segment or vertex the orbit passing the node f intersects.

How many qualitatively different patterns of flow do we have at the inter-boundary segment mapping level? Let $P_C(n)$ be the number of qualitatively different patterns of flow in a call with n concave nodes. First, let us consider the case in which no single

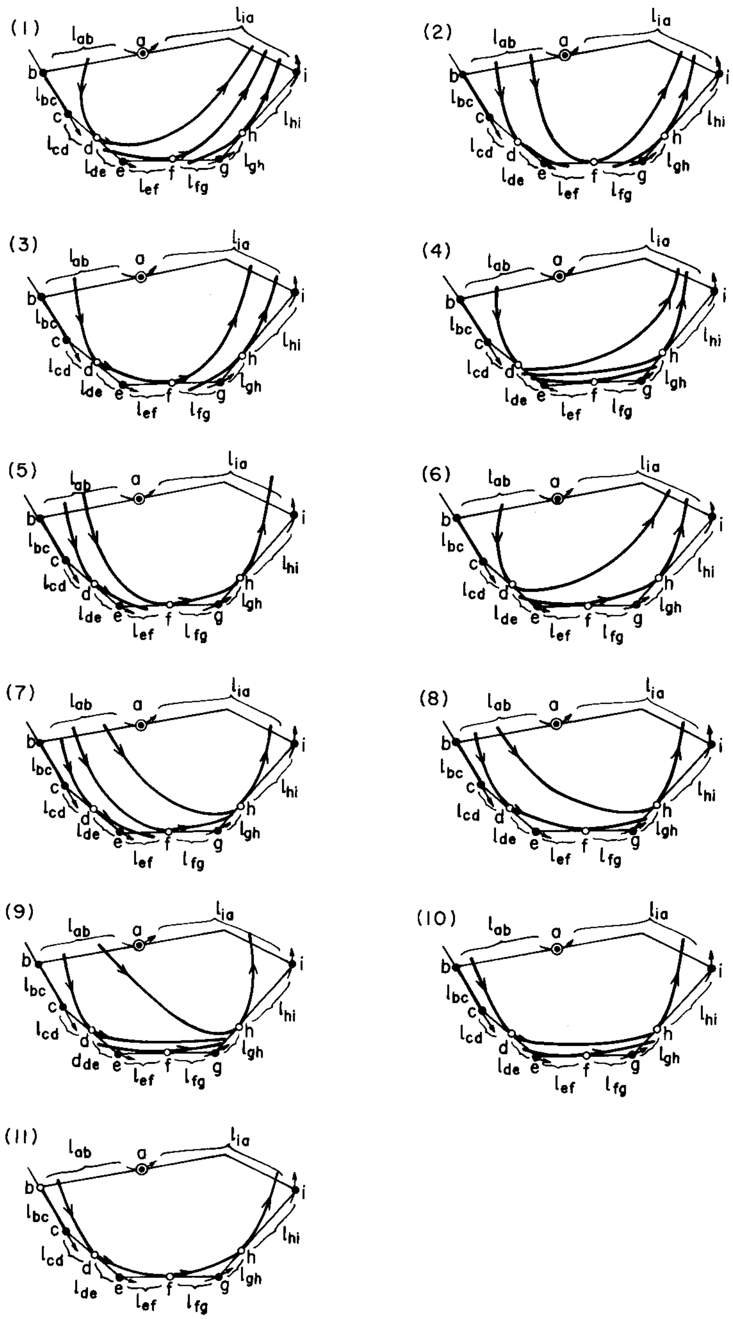


Figure 9: Eleven Possible Ways of Qualitatively Different Patterns of Local Flow for the Cell shown in Figure 8.

orbit passes more than one concave node and let the number of possibilities in that case be $P_C^N(n)$. $P_C^N(n)$ is recursively defined as follows:

$$P_C^N(0)=1$$

$$P_C^N(1)=1$$

$$P_C^N(n)=\sum_{i=1}^{n} P_C^N(i-1)P_C^N(n-i) \qquad n\geq 2$$

The values of $P_C^N(n)$ for small n are as follows:

$P_C^N(2)=2$, $P_C^N(3)=5$, $P_C^N(4)=14$, $P_C^N(5)=42$, $P_C^N(6)=132$, $P_C^N(7)=429$, $\cdots$

Using $P_C^N(n)$ as defined above, $P_C(n)$ is defined as follows:

$$P_C(0)=1$$

$$P_C(1)=1$$

$$\begin{aligned}P_C(n)=\sum_{p=1}^{n-1}\sum_{1\leq i_1<i_2<\cdots<i_p\leq n-1}&\left(\sum_{k=0}^{i_1-1}P_C(k)P_C(i_1-k-1)\right)\\&\times\left(\Pi_{j=1}^{p-1}P_C(i_{j+1}-i_j-1)\right)\\&\times P_C(n-i_p-1)\\+\sum_{i=0}^{n-1}P_C(5)P_C(n-i-1)&\qquad\text{where, } n\geq 2.\end{aligned}$$

The values of $P_C(n)$ for small n are:

$P_C(2)=3$, $P_C(3)=11$, $P_C(4)=45$, $P_C(5)=197$, $P_C(6)=903$, $P_C(7)=4279$, $\cdots$

5.3 Properties of Local Flow in Open Cells

We can reduce the problem of analyzing local flow in open cells to that of closed cells, by extending the set of boundary segments with additional virtual boundary entities such as points at infinity or an edge at infinity and giving them appropriate attributes so that the following conditions may be satisfied:

1. the number of convex segment nodes is that of concave nodes plus *two*
2. there is at least one entrance segment and exit segment in the extended set of boundary segments
3. flow at the both sides of every singular segment is converse with respect to the orientation relative to the boundary; if the flow is inward at one side, the flow should be outward at the other side.

Before going into details, let us examine several examples. First, consider an open cell shown in Figure 10. This cell has three real boundary edges. Two of them are part of invariant manifolds. The remaining one is an exit segment. Three additional virtual boundary entities, two points at infinity and an edge at infinity, are introduced. The constraint on the number of concave nodes and convex node segments is already satisfied. But as there is no entrance segment, we regard the edge at infinity as an entrance segment. This interpretation conforms to the fact that all of the orbits crossing the real exit segment diverges to the direction delimited by the two invariant manifolds

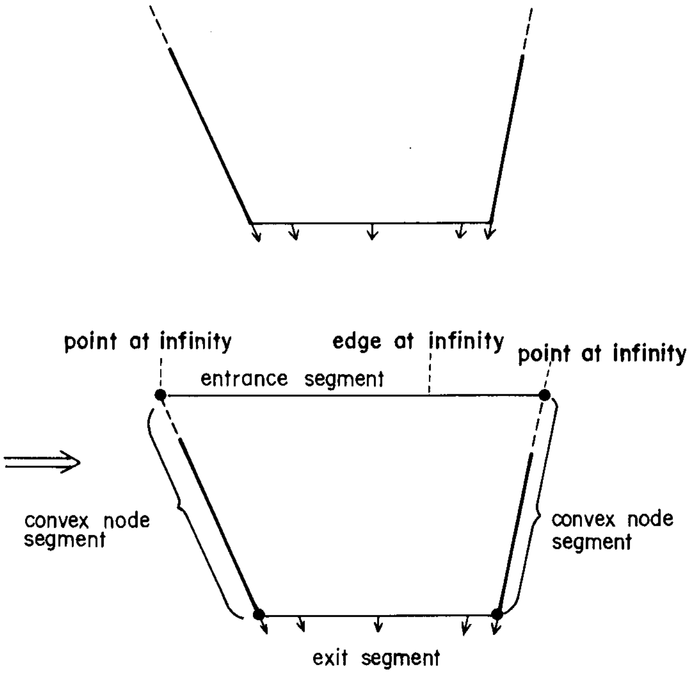


Figure 10: Example of Local Flow in Open Cell (1)

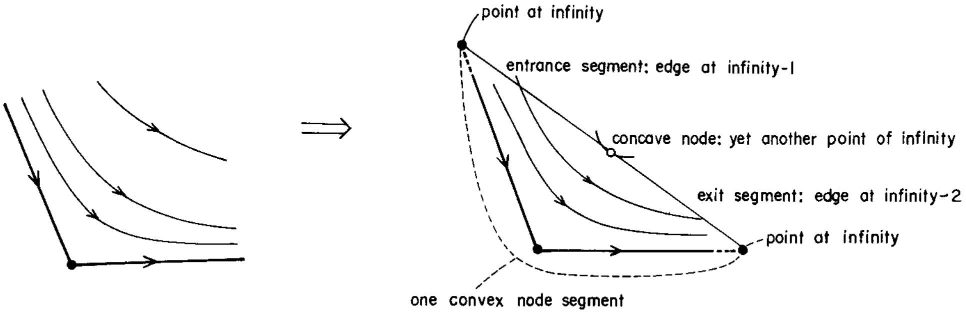

point at infinity
entrance segment: edge at infinity-1
concave node: yet another point of infinity
exit segment: edge at infinity-2
point at infinity
one convex node segment

Figure 11: Example of Local Flow in Open Cell (2)

as $t \to -\infty$.

Consider now the flow in an open cell shown in Figure 11. Two boundary edges comprising the boundary of the cell are both portions of invariant manifolds connected by a saddle node. All of these boundary entities are viewed as one convex node segment. Three additional virtual boundary entities are introduced as before. In this case, there is no entrance segment nor an exit segment on the boundary. In order to satisfy the constraints on boundary entities, we have to divide the edge at infinity into two and view them as an entrance segment and an exit segment.

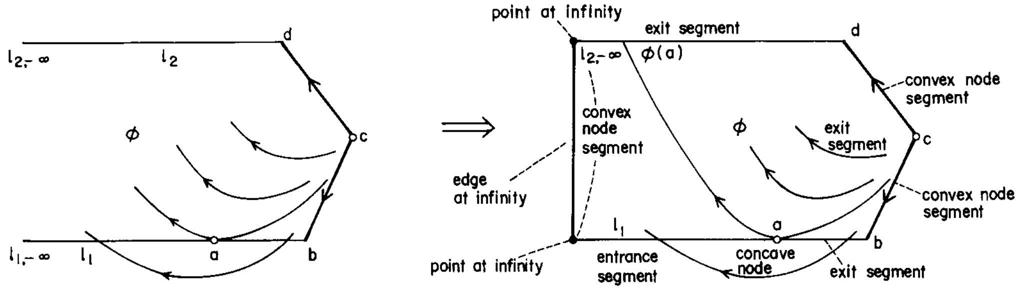


Figure 12: Example of Local Flow in Open Cell (3)

Finally consider a local flow in an open cell shown in Figure 12, which has a concave node *a* on the bounday. Although both entrance segments and exit segments exist on the boundary, the number of convex node segments is one less than required. Hence, we regard the edge at infinity a convex node segment. In this case, the local flow in this cell at the inter-boundary segment level is uniquely characterized as the following set of mappings:

$\overline{l_{1,-\infty}a} \rightarrow \overline{l_{2,-\infty}\phi(a)}$, $c \rightarrow \overline{ab}$, $c \rightarrow \overline{\phi(a)d}$

Now we are in order to describe the set of rules for assigning attributes to the edge at infinity e_∞ and two points of infinity $p_{\infty,1}$ and $p_{\infty,2}$. Let the number of "real" concave node and convex node segments be n_c and n_v, respectively.

1. if $n_v = n_c + 2$, then
   (a) if the open cell has no entrance segment, then e_∞ is an entrance segment
   (b) if the open cell has no exit segment, then e_∞ is an exit segment
   (c) otherwise, e_∞ is a convex node segment or its portion
2. if $n_v = n_c + 1$, then
   (a) if the cell has both an entrance segment and an exit segment, then e_∞ is a convex node segment
   (b) if the cell has an entrance segment but no exit segment, then e_∞ is an exit segment.
   (c) similarly, if the cell has an exit segment but no entrance segment, then e_∞ is an entrance segment.
   (d) otherwise (*i. e.*, the cell has no entrance segment nor exit segment), then divide e_∞ into two by introducing a new virtual vertex and regard the two sub-segments resulting from subdivision as an entrance segment and an exit segment, respectively. The sign of eigenvalue associated with the adjacent invariant manifold is examined to determine the type of the two boundary segments: the boundary segment adjacent to an unstable manifold (an invariant manifold associated with a positive eigenvalue) is regarded as an exit

segment, and the one adjacent to the stable manifold (an invariant manifold associate with a negative eigenvalue) is regarded as an entrance segment.

3. if $n_v = n_c$, then
   (a) if the cell has an entrance segment but no exit segment, then e_∞ is an exit segment.
   (b) similarly, if the cell has an exit segment but no entrance segment, then e_∞ is an entrance segment.
   (c) otherwise, regard $p_{\infty,1}$ and $p_{\infty,2}$ as convex nodes, and regard e_∞ as an entrance segment (exit) if the boundary segment on the other side of a point at infinity is an exit (entrance) segment[9].

The types of points of infinity p_∞, if not determined in the above, is determind based on the type of adjacent boundary segments:

1. if the type of the two adjacent boundary segments of p_∞ are the same, so is the type of p_∞
2. if one of the adjacent boundary segments of p_∞ is a singular segment, so is p_∞
3. otherwise, p_∞ is a convex node segment or its portion.

Edges and points at infinity with attributes assigned as above have the following properties:

- when the edge of infinity is regarded as an entrance segment, there are orbits which diverge to the corresponding direction as $t \rightarrow -\infty$
- when the edge of infinity is regarded as an exit segment, there are orbits which diverge to the corresponding direction as $t \rightarrow \infty$
- when the edge of infinity is regarded as a singular segment, no orbit diverges to the corresponding direction as $t \rightarrow \pm\infty$

6. Implementing the Idea

PSX has been implemented based on the ideas and observation presented above. This section surveys implementation issues about PSX, Figure 13 shows an overview of the flow of processing which PSX performs. A document on the details of implementation is in preparation [4].

6. 1 Dividing the Phase Space into Cells

We take a simple strategy for generating cells[10]. We divide the phase space by

- boundaries between linear regions
- invariant manifolds in each linear region.

[9] The result does not depend on which one of the two points of infinity is chosen.

[10] Extensive evaluation of adequacy of this strategy is left for future work.

1. divide the phase space into cells
2. local analysis
   (a) instantiate boundary segments
   (b) generate inter-boundary segment mappings
   (c) generate inter-boundary edge mappings
   (d) generate local mappings
3. global analysis
   (a) inittialize annotations to intervals
   (b) extend local mappings

Figure 13: Flow of PSX

2-dimensional regions resulting from the division are used as cells if they satisfy the specification of some linear region. Since linear regions can overlap, there may be a cell shared by several linear rigions. Sufficient number of copies may be made for such cells so that different copies are used for different linear regions.

The region manager maintains information about regions resulting from phase space division. For each linear region, a record is maintained which provides

● specification of linear flow

● the names of i-dimensional regions ($i=1, 2, 3$) involved in the region

● the names of sampling lines and associated local coordinates.

These entities are annotated with geometric or topological information. Each sampling line with local coordinate is associated with the location of base, orientation, list of landmarks (0-dimensional region), property of local flow there, and other miscellaneous information. Each i-dimensional region is given such attributes as:

x-address, y-address, delimited-intervals (for 0-dimensional regions);

associated-sampling-line, upper-limit, lower-limit, right-side, left-side (for 1-dimensional regions);

subcategorization-of-the-region, boundary-type, set-of-boundary-entities, number-of-concave-nodes, number-of-convex-nodes, number-of-entrance-segments, number-of exit-segments, the-center-segment, orientation-of-flow-in-the-cell, left-cyclic-segment-list (for 2-dimensional regions)[11].

Some of those attributes are specified at this stage, while others will be specified later. Given a new sampling line, the region manager updates the records so that the division by the new line can be correctly reflected in the data structure.

6. 2 Local Analysis

In local analysis, the properties of local flow in each cell are examined individually, and the result is produced as a set of mappings between sampling lines for each cell.

[11] Some of the labels of these attributes are slightly different from the actual implementation.

6. 2. 1 Instantiating Boundary Segments

We use a four-pass algorithm. The first two passes, pass 1 and pass 2, are for processing closed cells, and the remaining two, pass 3 and pass 4, are for open cells. Pass 1 generates annotations about flow at the boundary edges and vertices whose type can be determined by local information. Some boundary segments, such as concave nodes or sinks or sources, which consist of a single point and can be determined solely by local information, are determined at this point, while others such as convex node segments consisting of more than one boundary edge are only partially specified. Pass 2 determines the type of those remaining boundary segments based on the annotation to the adjacent boundary entities. At the same time, a sequence of boundary entities of the same type is aggregated into a single boundary segments. Record is kept on how each boundary segment is made up from boundary entities. The record will be referred to later to produce inter-boundary edge mappings from inter-boundary segment mappings. If the cell is closed, the instantiation is complete at this point; otherwise further two passes will be applied.

Pass 3 assigns the type to edges and points at infinity, if it can be determined from local information. Pass 4 is actually the same as pass 2, which determines the type of remaining (virtual) boundary entities based on that of adjacent boundary entities as well as aggregating adjacent boundary entities of the same type.

The algorithm for instantiating boundary segments is relatively straightforward. See [4] for more details.

6. 2. 2 Generating Inter-boundary Segment Mappings

We encode inter-boundary segment mapping as a set of following tuples:

$\langle l, a, r\rangle$

where, l and r are the name of boundary segments which the orbit passing a concave node a is transverse to on entrace to or on leave from the cell, and l is to the left of a which in turn is to the left of r. For example, a flow pattern (1) shown in Figure 8 can be represented as follows:

$\{\langle l_{ab}, d, l_{ia}\rangle, \langle l_{de}, f, l_{ia}\rangle, \langle l_{fg}, h, l_{ia}\rangle\}$

We call this data structure landmark orbit specification, LOS for shot.

An algorithm for generating an LOS uses the following set of sequences:

$L=\{\cdots\cdots, l_i, \cdots\cdots\}$ where, $l_i=(\cdots\cdots, l_{ij}, \cdots\cdots)$

where, l_i is a list of boundary segments, indicating that no orbit which separates boundary segments into two groups is registered yet. R is a variable in which the resulting tuples are accumulated. An algorithm for generating an LOS is this:

1. initialize L : let the initial value of L be the left-cyclic boundary segment list less all convex nodes
2. choose a concave node l_{ij} from L nonderterministically

3. find nondeterministically a couple of segment l_{ip} and l_{iq} in l_i such that l_{ip}, $l_{iq} \in l_i$, $1 \leq p \leq j-2$, and $j+2 \leq q$; divide l_i by l_{ip} and l_{iq}, and let the result be $l_i = (\alpha, l_{ip}, \beta, l_{ij}, \gamma, l_{iq}, \delta)$ where, α, β, γ, and, δ are (possibly null) sequence of boundary segments (type of l_{ip} and l_{iq} should be chosen properly to match the orientation of the flow in the cell; for example, l_{ip} must not be an exit segment if the flow in the cell is left-right.)
4. add $\langle l_{ip}, l_{ij}, l_{iq} \rangle$ to R
5. remove l_i from L, and add the following:
   - if l_{ip} is a concave node, then β; otherwise, $l_{ip} \cdot \beta$, where $x \cdot y$ means that item x is added to the head of list y. If a concave node is not involved in β, the addition is canceled.
   - if l_{iq} is a concave node, then γ; otherwise, γ with l_{iq} added in tail. If a concave node is not involved in γ, the addition is canceled.
   - a list resulting from appending α and δ. If the list does not contain any concave node, addition is canceled. When l_{ip} or l_{iq} are not concave nodes, l_{ip} or/and l_{iq} are added in advance in the tail of α or to the top of δ, respectively.

The above algorithm is non-deterministic, for steps 2 and 3 contain arbitrary choice. Discussion about nondeterminacy will made below in section 6.4.

An example. When applied to the cell shown in Figure 8, the above algorithm will produce the following LOS:

$\{\{\langle l_{ab}, d, l_{ia} \rangle, \langle l_{de}, f, l_{ia} \rangle, \langle l_{fg}, h, l_{ia} \rangle\},$
$\{\langle l_{ab}, d, l_{ef} \rangle, \langle l_{ab}, f, l_{ia} \rangle, \langle l_{fg}, g, l_{ia} \rangle\},$
$\{\langle f, d, l_{ia} \rangle, \langle l_{ia}, f, d \rangle, \langle l_{fg}, h, l_{ia} \rangle\},$
$\{\langle l_{ab}, d, l_{ia} \rangle, \langle l_{de}, f, l_{gh} \rangle, \langle l_{de}, h, l_{ia} \rangle\},$
$\dots$
$\{\langle l_{ab}, d, f \rangle, \langle d, f, h \rangle, \langle f, h, l_{ia} \rangle\}\}$

Note that the above description conforms to the eleven possible flow patterns shown in Figure 9.

Let us represent an inter-boundary segment mapping as: $\langle\langle s, e \rangle, \langle \phi(e), \phi(s) \rangle\rangle$ which means that the part of a boundary segment delimited by s and e is mapped to the part of another boundary segment delimited by $\phi(e)$ and $\phi(s)$ by a local flow inside the cell.

It is relatively easy to translate LOS into inter-boundary segment mappings. We use as working memory stack $s(\alpha)$ associated with each entrance or exit segment α. The result will be put into the variable m, whose initial value is an empty set $\{\}$. The algorithm for translating LOS into the above representation of inter-boundary segment mappings is this:

1. reorder elements of given LOS so that $\langle l_j, a_j, r_j \rangle$ may appear after $\langle l_i, a_i, r_i \rangle$ in

the resulting list if a_j is to the right of a_i according to the left-cyclic borndary segment list.

2. for each tuple $\langle l, a, r\rangle$ in the list obtained in the previous step, do the following:
   (a) if l is an entrance segment or an exit segment, create a new cut and a name standing for it; and push the name into $s(l)$
   (b) do similarly if r is an entrance segment or an exit segment, except that the new name is pushed to $s(r)$
3. for each entrance segment i, do the following:
   (a) let $s^+(i)$ be $s(i)$ with the names of two adjacent singular node segments added to the top and in the tail, respectively
   (b) for each consecutive element $l_{i,j}$ and $l_{i,j+1}$ of $s^+(i)$, compute the image $\phi(l_{i,j})$, $\phi(l_{i,j+1})$ based on the data contained in LOS ($\phi(x)$ denotes a point on the boundary, either a cut or a concave node, which the orbit passing on x goes through when it arrives or leaves the cell);
   add $\langle\langle l_{i,j}, l_{i,j+1}\rangle, \langle\phi(l_{i,j+1}), \phi(l_{i,j})\rangle\rangle$ to m

6. 2. 3 Generating Inter-boundary Edge Mappings

In order to obtain inter-boundary edge mappings the inter-boundary segment mapping

$\langle\langle s, e\rangle, \langle\phi(e), \phi(s)\rangle\rangle$

will be further subdivided into

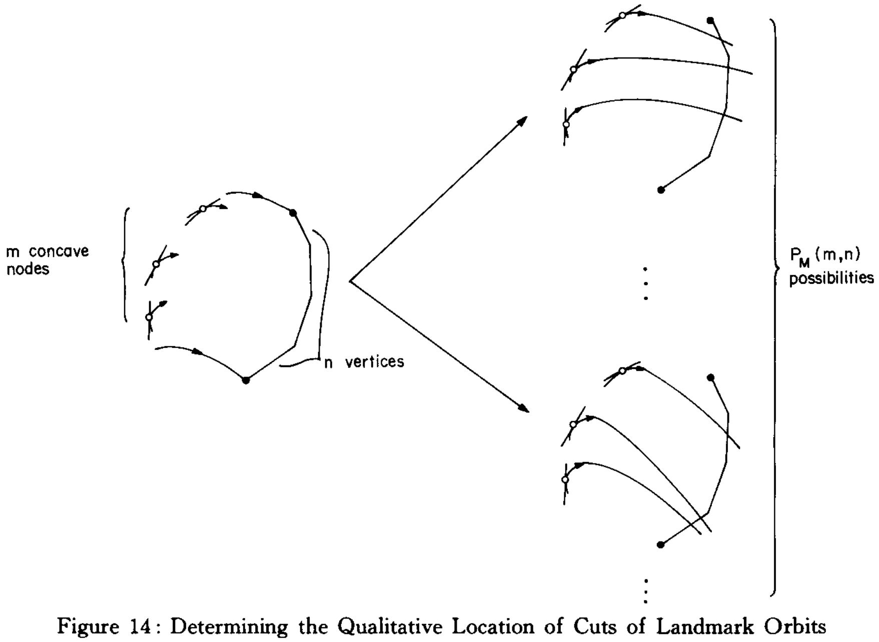


Figure 14: Determining the Qualitative Location of Cuts of Landmark Orbits

$\{\cdots\cdots\langle\langle l_{i,a}, l_{i,b}\rangle, \langle\phi(l_{i,c}), \phi(l_{i,d})\rangle\rangle\cdots\cdots\}$ where $\{a, b\}=\{c, d\}$

if a boundary segment denoted by $\langle s, e\rangle$ or another denoted by $\langle\phi(e), \phi(s)\rangle$ consists of more than one boundary edge. In the above definition, intervals denoted by $\langle l_{i,a}, l_{i,b}\rangle$ or $\langle\phi(l_{i,c}), \phi(l_{i,d})\rangle$ must be on a single sampling line. In addition, local address of $l_{i,a}$ and $\phi(l_{i,c})$ should be smaller than that of $l_{i,b}$ and $\phi(l_{i,d})$, respectively.

Generation of inter-boundary edge mapping is performed in two stages. First, location of cuts of landmark orbits that pass concave nodes is qualitatively determined (that is, a boundary edge of vertex on which the cut is located is determined). Then, the location of cuts of orbits passing on vertices is qualitatively determined.

Determining the qualitative location of cuts of landmark orbits. Constraints on locations of cuts are:

- the ordering of cuts should be preserved
- arbitrary number of cuts can be located on a boundary edge
- at most one cut can be located on a vertex.

See Figure 14.

Let the number of vertices and the number of cuts involved in the entrance segment or the exit segment s be m and n, respectively. And let the number of possibilities for n cuts to be on m vertices and $m+1$ bundary edges be $P_M(m, n)$.

Then,

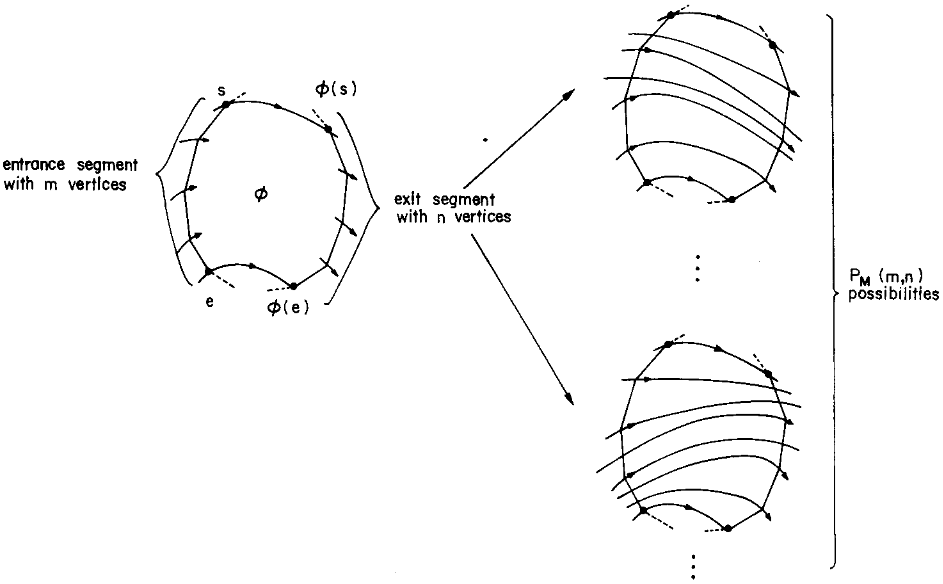


Figure 15: Determining the Qualitative Location of Cuts of Orbits Passing on Vertices

$$P_M(m,\ n)=P_M^N(m,\ n)$$
$$+\sum_{p=1,\dots,\min(m,n)}\ \sum_{1\leq i_1<\dots<i_p\leq n}\ \sum_{1\leq j_1<\dots<j_p\leq m} T(i_1,\ \cdots\cdots,\ i_p\ ;\ j_1\cdots\cdots,\ j_p) \tag{8}$$

where,

$$P_M^N(m,\ n)={}_{m+n}C_n \tag{9}$$

$$T(i_1,\ \cdots\cdots,\ i_p\ ;\ j_1,\ \cdots\cdots,\ j_p)=P_M^N(i_1-1,\ j_1-1) \tag{10}$$
$$\times(\Pi_{\alpha=1}^{p-1}P_M^N(i_{\alpha+1}-i_\alpha-1,\ j_{\alpha+1}-j_\alpha-1)$$
$$\times P_M^N(m-i_p,\ n-j_p)$$

Determining the location of cuts of orbits passing on vertices. We can use a simple nondeterministic list merge algorithm to enumerate possible qualitative locations of cuts of orbits that pass on vertices involved in entrance or exit segments. As shown in Figure 15, there are again, $P_M(m,\ n)$ possibilities arising for an inter-boundary edge mapping $\langle\langle s,\ e\rangle,\ \langle\phi(e),\ \phi(s)\rangle\rangle$, if there are m vertices in $\langle s,\ e\rangle$ and n vertices in $\langle\phi(e),\ \phi(s)\rangle$.

6. 2. 4 Generating Local Mappings

Now it is trivial to produce local mappings

$$\{f_k : I_k \rightarrow J_k\} \tag{11}$$

from inter-boundary-segment mappings. Besides this, information which PSX actually produces involves:

- polarity of local mapping; whether a sequence of landmarks is mapped in the same order (+) or in the reverse order (−)
- list of asymptotic sources corresponding to sources at boundary
- list of asymptotic sinks corresponding to sinks at boundary

Since local analysis is performed independently for each cell, landmarks and intervals on the physically same sampling line may be produced without being co-related with each other. For example, consider a situation illustrated in Figure 16, where cell a has interval $\overline{a_1a_4}$ as part of the boundary, cell b has interval $\overline{b_1b_5}$ as part of the boundary, and $\overline{a_1a_4}$ and $\overline{b_1b_5}$ are defined on the common sampling line l_k with an associated local

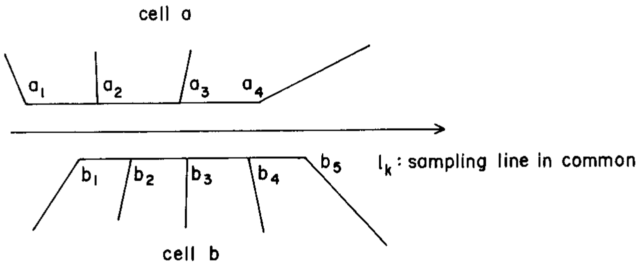


Figure 16: Two Overlapping Sequences of Landmarks and Intervals Defined on the Same Sampling Line

coordinate c_k. Suppose that we only know $a_1 \prec_{ck} b_1 \prec_{ck} a_4 \prec_{ck} b_5$, $a_1 \prec_{ck} a_2 \prec_{ck} a_3 \prec_{ck} a_4$ and $b_1 \prec_{ck} b_2 \prec_{ck} b_3 \prec_{ck} b_4 \prec_{ck} b_5$ about those landmarks. Since global analysis requires landmarks on a sampling line to be totally ordered, the two sequences $\{a_i\}$ and $\{b_j\}$ of landmarks are merged so that the above constraints may be satisfied and such a sequence as $(a_1, b_1, a_2, b_2, a_3, b_3, b_4, a_4, b_5)$ may be constructed. This can be done by a simple nondeterministic list merge algorithm.

6. 3 Global Analysis

The purpose of global analysis is to examine the structure of compositions of local mappings obtained in local analysis.

6. 3. 1 Problems and Requirements to Global Analysis

The complexity of global analysis significantly differs, depending on whether uniqueness of solution is the case. If uniqueness of the solution holds, it is relatively easy to implement an effective algorithm for global analysis.

Unfortunately, uniqueness of the solution does not hold in general for our subject. If some linear regions overlap as shown in Figure 17, orbits may branch or merge at the boundary. We take each branching as representing nondeterministic evolution of state.

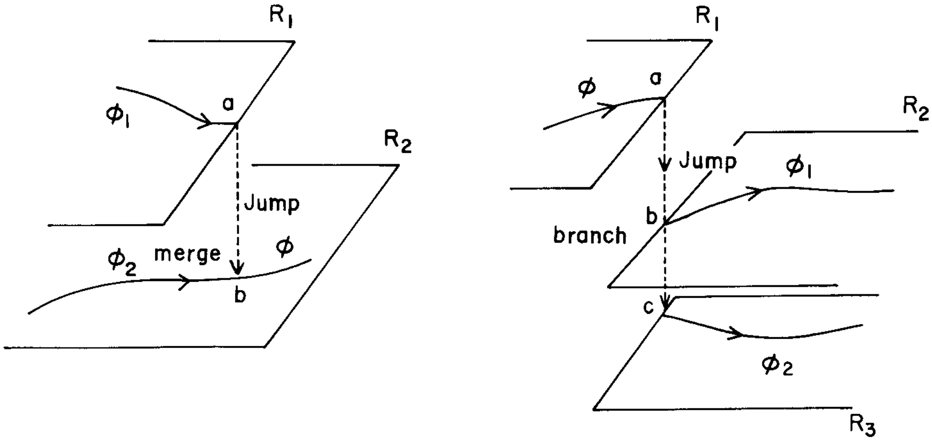


Figure 17 : Merge and Branch of Orbits in Complex Phase Space

When uniqueness of solution does not hold, the analysis of phase portrait becomes harder, for orbits can tangle in an arbitrarily complex manner. See orbits in Figure 18 for illustration of such a situation. There, bundle of orbit intervals ϕ_1 and ϕ_2 merge together to connect to ϕ_3, part of which in turn merges with ϕ_8 to connect to ϕ_4. There is a loop structure consisting of ϕ_4, ϕ_6, and part of ϕ_8. This is not a contracting recursive bundle of orbits in a strict sense, for it contains a nondeterministic branch to ϕ_7. But it is useful to recognize these loops as kinds of recursive mappings and use them as a clue for finding recursive mappings in a strict sense.

We have developed two algorithms for global analysis: one that assumes the uniqueness of solution and another that does not assume that. Since the latter is more general, we will only present the latter. Before that, let us describe basic issues underlying the algorithm.

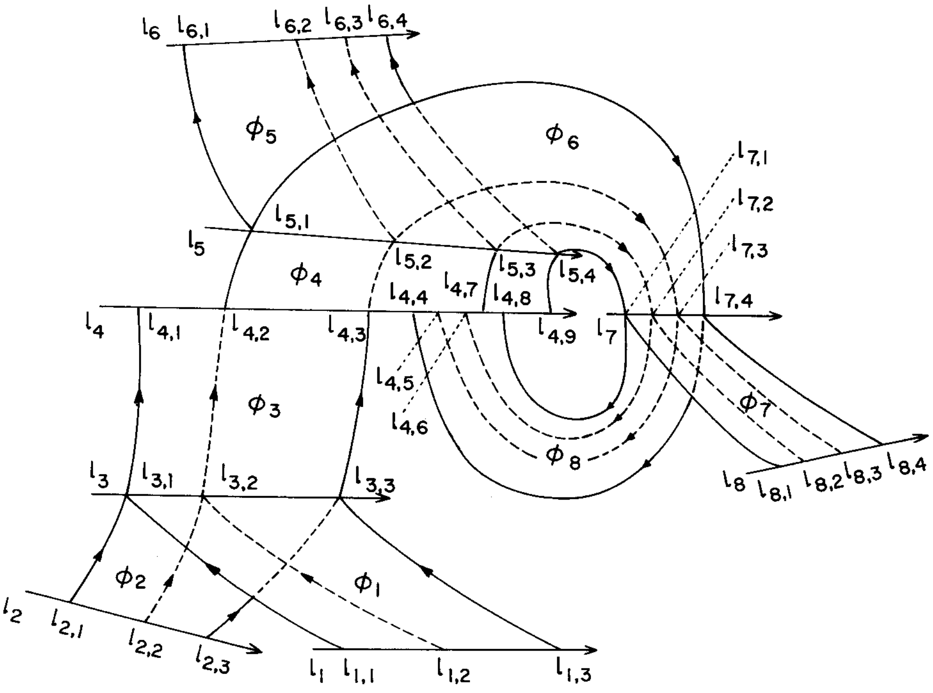


Figure 18: Collection of Bundle of Orbit Intervals with Complex Connectivity

6. 3. 2 Propagation of Split

Suppose $\phi : I \to J$, $\psi : K \to L$, and $J \cap K \neq \{\}$, but $J \neq K$. In this case, although we cannot construct $\psi \circ \phi$, we can split J or/and K and construct a composition $\psi \circ \phi : \phi^{-1}(J \cap K) \to \psi(J \cap K)$. This may cause intervals I and K to be split. This kind of split will be propagated in turn to the rest of the intervals. This is a potential cause of preventing the global analysis from termination. This problem is avoided by bringing in a breadth first feature to global analysis.

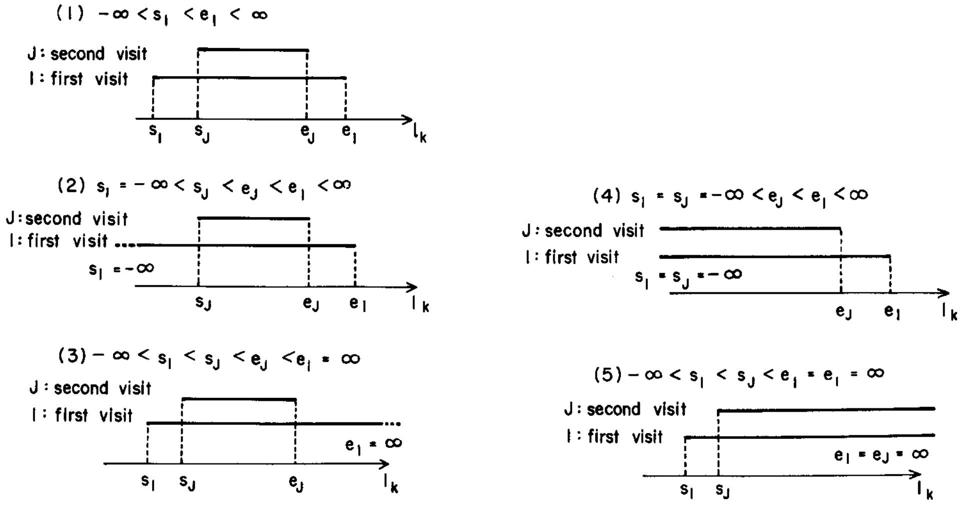
6. 3. 3 Coincidence Patterns and their Interpretation

In the process of constructing composition of mappings, we say that coincidence occurs if a mapping is constructed such that $\phi_m \circ \cdots\cdots \circ \phi_1 : I \to J$ and $I \cap J \neq \{\}$. Let c_k be a local coordinate associated with sampling line l_k, and (s_I, e_I) and (s_J, e_J) be intervals corresponding to I and J, respectively. We can classify pattens of coincidence as in Table 2.

Table 2 : Coincidence Patterns

name	definition
left	$e_J < s_I$
left-touching	$e_J = s_I$
right	$e_I < s_J$
right-touching	$e_I = s_J$
absorbed	$s_I < s_J < e_J < e_I$
absorbed-left-touching	$s_I = s_J < e_J < e_I$
absorbed-right-touching	$s_I < s_J < e_J = e_I$
identical	$s_I = s_J < e_J = e_I$
absorbing	$s_J < s_I < e_I < e_J$
absorbing-left-touching	$s_J = s_I < e_I < e_J$
absorbing-right-touching	$s_J < s_I < e_I = e_J$
overlapping-left	$s_J < s_I < e_J < e_I$
overlapping-right	$s_I < s_J < e_I < e_J$

Although 'left', 'left-touching', 'right', 'right-touching' are not coincidental, they are included in the table for completeness of classification.


(1) $-\infty < s_I < e_I < \infty$

(2) $s_I = -\infty < s_J < e_J < e_I < \infty$

(3) $-\infty < s_I < s_J < e_J < e_I = \infty$

(4) $s_I = s_J = -\infty < e_J < e_I < \infty$

(5) $-\infty < s_I < s_J < e_I = e_I = \infty$

Figure 19 : Subcategorization of Coincidence Patten 'absorbed'

Below we show typical cases and their interpretation.

'absorbed'

First, let us assume that the uniqueness of solution holds. Type 'absorbed' can be further classified into five subclasses as shown in Figure 19.

If $-\infty < s_J$, $e_J < \infty$ (which corresponds to type (1)-(3) in Figure 19), we can make the following conclusions:

1. there exists one or more limit cycles $\{a_1, \cdots\cdots, a_k\}$ which pass interval J of local coodinate c_k; set $\{a_1, \cdots\cdots, a_k\}$ is called an attracting bundle of orbits
2. all orbits transverse to J asymptotically approach one of $\{a_1, \cdots\cdots, a_k\}$; interval J is called an entrance to the attracting bundle of orbits.

Although it is hard to see how many limit cycles are there by a qualitative or quantitative method, it may not matter in application as far as the range of its existence (s_J, e_J) is precise enough. We simply regard $\{a_1, \cdots\cdots, a_k\}$ as a single destination.

If $s_I = s_J = -\infty$, $e_I = e_J = \infty$ (which corresponds to types (4) and (5) in Figure 19), we cannot predict the existence of an attracting bundle of orbits, since there is a possibility that orbits diverge to infinity. PSX simply suggests the possibility of existence of an attracting bundle of orbits.

When a uniqueness of solution is not confirmed, we should be more careful about prediction. For example, we cannot predict the existence of an attracting bundle of orbits from the facts $J \subset I$ and $-\infty < s_J$, $e_J < \infty$, for one of nondeterministic branch of orbit may have a branch which breaks the recurrence. Such nondeterministic cycle of bundle of orbit intervals is called an attracting bundle of orbits in a weaker sense and is distinguished from real attracting bundle of orbits (or attracting bundle of orbits in a stronger sense). We cannot make the prediction unless we have checked for all possible branches of orbits.

Interpretation of coincidence patterns 'absorbed-left-touching' and 'absorbed-right-touching' is similar, except that the predicted attracting bundle of orbits may be a point as well as a limit cycle. The interpretation of coincidence patterns 'absorbing', 'absorbing-left-touching' and 'absorbing-right-touching' is similar, too, except replacing t by $-t$ (going backward to the past) and "attracting" by "repelling". Pattern 'equivalent' can be viewed as a special case of 'absorbed' and 'absorbing'.

'overlapping-left'

First let us assume that the solution is unique. Two qualitatively distinct interpretations are possible as shown in Figure 20.

interpretation-1 there is a limit cycle c between s_I and e_J, and all orbits transverse to interval (e_J, e_I) of c_k asymptotically approach c as $t \to \infty$, and all orbits transverse to interval (s_J, s_I) asymptotically approach c as $t \to -\infty$.

interpretation-2 unlike interpretation-1, there is no limit cycle and all orbits trans-

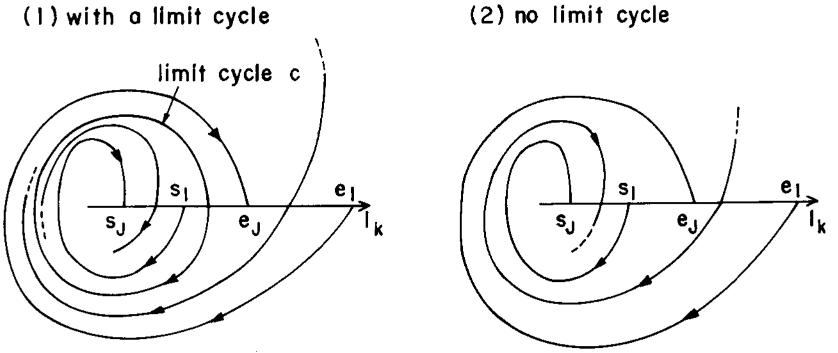


Figure 20: Two Possible Interpretations of Coincidence Pattern 'overlapping-left'

verse to interval (e_J, e_I) of c_k arrive at somewhere in (s_J, s_I) in a finite amount of time.

It is difficult to distinguish the two even by a quantitative method. Suppose we trace down the orbit passing on e_J. If the trace reaches (s_J, s_I), we can say that **interpretation-2** is the case. Otherwise, however, we cannot distinguish between the two interpretations. When the ambiguity is not resolved, PSX exploits each possibility separately.

The interpretation is made more deliberately when the uniqueness of the solution is not known.

6. 3. 4 Algorithm for Global Analysis

The basic idea behind the algorithm is this:

1. annotate each interval with historical information specifying the known source which is mapped to the interval and known destination to which the interval is mapped
2. repeat extending historical information in an incremental manner; asymptotic properties of the whole phase portrait is found in this cycle. Intervals may be subdivided in this phase.

In order to cope with complex branching and merging of orbits, we distinguish two sides of an interval; one side at which orbits arrive, and another side from which orbits depart. We call the former the arriving side and the latter the departing side. Annotation of each interval consists of one to the departing side and another to the arriving side. Annotation to the departing side is given as a tree structure each pass of which represents a known possible sequence of intervals the orbits are transverse to. Figure 21 shows an example. Each branch of the tree conforms to branching of orbits. Annotation to the arriving side is similarly made except that each sequence represents a passage back to the past, rather than to the future. In this case, each branch of the tree conforms to merging of orbits.

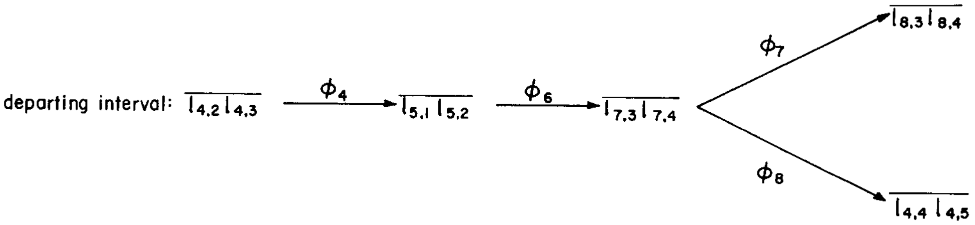


Figure 21 : Tree Structure For Representing History of Interval

At the beginning of global analysis, annotations to the departing and arriving sides of each interval are initialized to trees of depth one which reflect information obtained in local analysis. These trees of historical information are grown in an incremental fashion in the succeeding stages of global analysis.

Global analysis is implemented as a concurrent process using a prioritized task queue called an agenda. The structure of the agenda is

(······(*priority*$_i$ *process*$_i$)···)

where items of the agenda are sorted in a descending order of *priority*$_i$. The name and function of several major agenda processes are given below.

- connect-forward [*I*] attempts to construct a composition of a mapping ϕ with I as range and a mapping ψ with interval J such that $I \cap J \neq \{\}$ as domain.

  If there is another mapping ϕ' whose range is also I, composition is not made. Instead, aganda process merge-arriving-intervals is created which will merge ϕ and ϕ'. After the merge, connect-forwad will be put on agenda again.

  If there is another mapping ϕ'' whose range is not equal to I but is overlapping with I, composition is not made either. Agenda process split-arriving-intervals is created which will divide $I \cup I'$ into $I \cap I'$, $I - I \cap I'$, $I' - I \cap I'$. Composition will be tried later for mapping resulting from the above division. See Figure 22 for

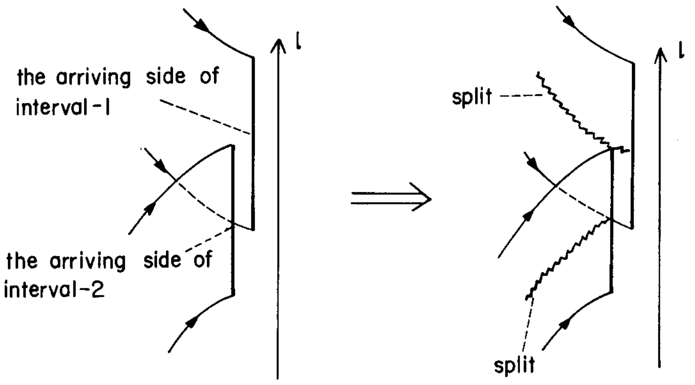

the arriving side of interval-1
the arriving side of interval-2
l
⟹
split
split
l

Figure 22 : Dividing Overlapping Arriving Sides of an Interval

the illustration of this process.

Similarly, if there is mapping $\phi' : J \rightarrow K$ such that $I \neq J$, $I \cap J \neq \{\}$, composition is not made. Agenda process split-departing-interval will be created which will divide $I \cup J$ into $I \cap J$, $I - I \cap J$, $J - I \cap J$.

Otherwise, composition is possible and agenda process connect-consecutive-intervals will be created which will actually construct the composition.

Agenda process connect-backward performs the similar processing.

- split-arriving-interval [J, L] divides the arriving side of J by a sequence of landmarks L. When this is done, agenda process map-intervals-backward will be created which will merge resulting landmarks with those that already exist in I such that $\phi : I \rightarrow J$.

  Process split-departing-interval [I, L] similarly divides the departing side of I by L.

- map-intervals-backward [J, D, I] propagates division D of the arriving side of interval J back to the departing side of interval I, and merges the resulting landmarks with those that are already involved in I. As a side effect of dividing interval I, interval J may again be divided.

  If I and J contain n_I and n_J landmarks respectively, then, there are $P_M(n_I, n_J)$ different ways[12] of possibilities arising here (see Figure 23). PSX either proceeds nondeterministically or calls for numerical computation to resolve ambiguity, as we will describe in section 6. 4.

  map-landmarks-forward [I, D, J] similarly propagates division D of I such that $\phi : I \rightarrow J$ forward to J.

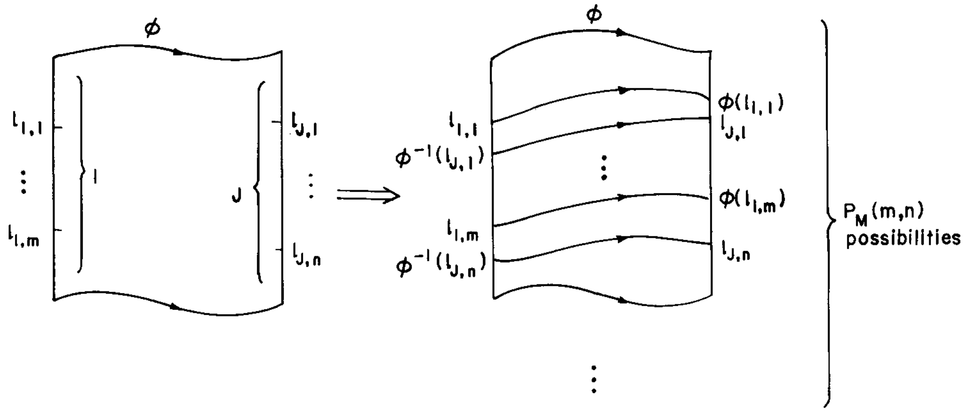


Figure 23: Backward Propagation of Division of the Arriving Side of Interval by a Sequence of Landmarks

[12] See (8) for definition of $P_M(x, y)$.

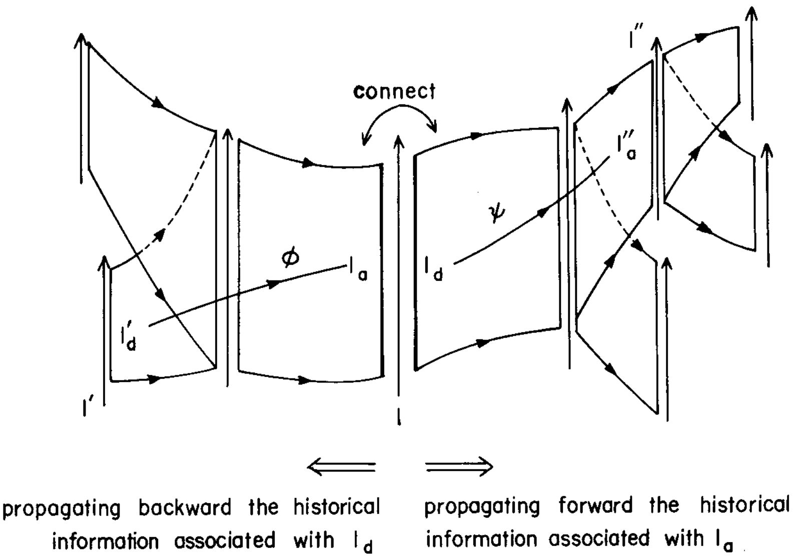

connect
ϕ
I_a
I_d
ψ
I'_d
I'
I
I''
I''_a
propagating backward the historical information associated with I_d
propagating forward the historical information associated with I_a

Figure 24 : Forward and Backward Propagation of Historical Information

- merge-arriving-intervals $[a_1, a_2]$ merges arriving sides a_1 and a_2. Similarly, merge-departing-intervals $[d_1, d_2]$ merges departing sides d_1 and d_2.
- connect-consecutive-intervals $[I_a, I_d]$ merges historical information associated with the arriving side I_a and the departing side I_d of interval I, and constructs an annotation reflecting more global information obtained from the compositions of mappings with I_a as range and mappings with I_d as domain. In addition, it updates annotations to intervals related to the composition. See Figure 24 for illustration.

  For example, historical information associated with the departing side I_d is propagated backward and is added historical information associated with the departing side I'_d of each interval I' which is mapped to I_a by mapping ϕ. (backward propagation of historical information of the departing side I_d). This operation corresponds to synthesizing historical information about $\psi \circ \phi$, where ϕ is a mapping from I'_d to I_a and ψ is a mapping with I_d as domain.

  Similar processing is performed to the arriving side I''_a of each interval I'' to which the departing side I_d is mapped (forward propagation of historical information of the arriving side I_a).

  After this processing is complete, agenda process check-for-coincidence $[I_a, I_d]$ will be created which will check for coincidence as a result of composition.
- check-for-coincidence $[I_a, I_d]$ uses historical information associated with the arriving side I_a and the departing side I_d of interval I to seek for a pair $\phi^{-1}(I)$ and

$\phi(I)$ such that $\phi^{-1}(I) \cap \phi(I) \neq \{\}$. Existence of such a pair entails the existence of coincidence. If this is the case, the coincidence pattern is examined and an appropriate process for handling coincidence will be created.

6. 4 About Nondeterminacy

As we have seen so far, nondeterminacy arises at various points of the algorithm. This is a trade-off resulting from computing with incomplete information using a simpler class of numerical calculus. We have two versions of PSX, one enumerates all possibilities (as far as computing resource is available) and the other calls for extensive numerical calculus to resolve ambiguity. Note that both algorithms work completely the same way as far as computation with a limited class of numerical calculus can uniquely determine the qualitative properties of flow.

7. Example

7. 1 Analysis of Van der Pol's Equation

Consider again the piecewise linear approximation

$$\left\{\begin{array}{l} \mathrm{R}_+ : \left\langle \frac{1}{2} \leq x \mid \dot{x} = \frac{-2x+y+2}{c},\ \dot{y} = -x \right\rangle \\ \mathrm{R}_0 : \left\langle -\frac{1}{2} \leq x \leq \frac{1}{2} \mid \dot{x} = \frac{2x+y}{c},\ \dot{y} = -x \right\rangle \\ \mathrm{R}_- : \left\langle x \leq -\frac{1}{2} \mid \dot{x} = \frac{-2x+y-2}{c},\ \dot{y} = -x \right\rangle \end{array}\right\} \tag{12}$$

of Van der Pol's equation that we cited in the beginning of this paper. Now we describe the way PSX analyzes this form in more detail[13].

Dividing the Phase Space into Cells

PSX first divides R^2 by straight lines $x = -1/2$ and $x = 1/2$ which are boundaries between the three linear regions. PSX then divides each linear region by invariant manifolds if any.

The linear region R_+ has a fixed point (sink) at (0, −2). The coefficient matrix of linear flow in this region has as eigenvalues λ_1, λ_2 and eigenvectors v_1, v_2

$$\lambda_1 = \frac{-1+\sqrt{1-c}}{c} \qquad v_1 = \begin{bmatrix} \dfrac{1-\sqrt{1-c}}{c} \\ 1 \end{bmatrix}$$

$$\lambda_2 = \frac{-1-\sqrt{1-c}}{c} \qquad v_2 = \begin{bmatrix} \dfrac{1+\sqrt{1-c}}{c} \\ 1 \end{bmatrix}$$

[13] Although parameter *c* should be replaced by some specific real number, we will leave *c* as a symbolic parameter to demonstrate the process of computation.

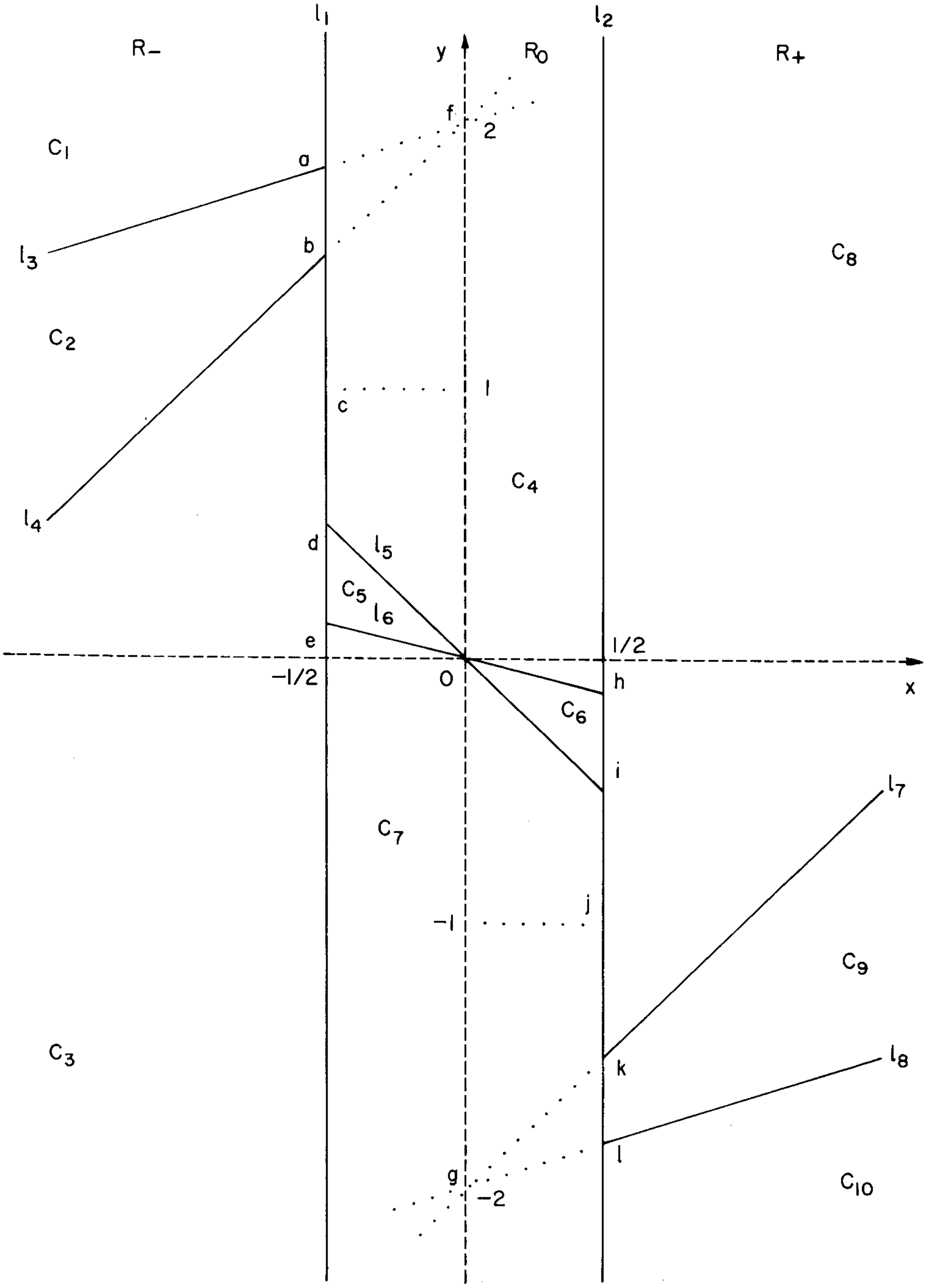


Figure 25 : Dividing the Phase Spase for Piecewise Linear Differential Approximation (12) of Van der Pol's Equation

$$
\begin{aligned}
&\mathrm{C}_1 : \phi_1 : s_1 \rightarrow \overline{a l_{1,\infty}}\\
&\mathrm{C}_2 : \phi_2 : s_2 \rightarrow \overline{ba}\\
&\mathrm{C}_3 : \phi_3 : \overline{l_{1,-\infty} c} \rightarrow \overline{cb}\\
&\mathrm{C}_4 : \phi_{4,1} : s_3 \rightarrow \overline{dc}\\
&\qquad \phi_{4,2} : \overline{c l_{1,\infty}} \rightarrow \overline{\phi_4(c)\, l_{2,\infty}}\\
&\qquad \phi_{4,3} : s_3 \rightarrow \overline{h \phi_4(c)}\\
&\mathrm{C}_5 : \phi_5 : s_3 \rightarrow \overline{ed}\\
&C_6 : \phi_6 : s_3 \rightarrow \overline{ih}\\
&\mathrm{C}_7 : \phi_{7,1} : s_3 \rightarrow \overline{ji}\\
&\qquad \phi_{7,2} : \overline{l_{2,-\infty} j} \rightarrow \overline{l_{1,-\infty} \phi_7(j)}\\
&\qquad \phi_{7,3} : s_3 \rightarrow \overline{\phi_7(j)\, e}\\
&C_8 : \phi_8 : \overline{j l_{2,\infty}} \rightarrow \overline{kj}\\
&\mathrm{C}_9 : \phi_9 : s_4 \rightarrow \overline{lk}\\
&\mathrm{C}_{10} : \phi_{10} : s_5 \rightarrow \overline{l_{2,-\infty} l}
\end{aligned}
$$

where,

s_1 : source located in the region delimited by sampling lines l_1, l_3.
s_2 : source located in the region delimited by sampling lines l_3, l_4.
s_3 : source located at the origin of the phase space
s_4 : source located in the region delimited by sampling lines l_7, l_8.
s_5 : source located in the region delizmited by sampling lines l_8, l_2.

and

$$
\begin{aligned}
&h \prec_{c_2} \phi_4(c) \prec_{c_2} l_{2,\infty}\\
&l_{1,-\infty} \prec_{c_1} \phi_7(j) \prec_{c_1} e
\end{aligned}
$$

Figure 26 : Local Analysis of Piecewise Linear Approximation (12) of Van der Pol's equation

Thus, this region has two invariant manifolds, both stable, corresponding to two eigenspaces. So PSX divides R_+ by two straight lines corresponding to these invariant manifolds that pass on $(0, -2)$ and are oriented to v_1 and v_2, respectively. PSX divides the other two linear regions in a similar fashion, resulting in *ten* cells as shown in Figure 25.

Let us denote the local coordinate associated with sampling line l_i as c_i, and local flow in cell C_j as ϕ_j. The result of local analysis is a set of *fourteen* local mappings as shown in Figure 26. Figure 27 visualizes it.

In global analysis, PSX extends in turn annotations to each interval, attempting to grasp global and asymptotic behavior. During this process, PSX extends the annotation to the departing side of interval $\overline{l_{2,-\infty} j}$. This interval is mapped to interval $\overline{l_{1,-\infty}\phi_7(j)}$ by flow ϕ_7 in cell C_7. This mapping cannot be extended immediately, for there is no interval whose domain is equal to $\overline{l_{1,-\infty}\phi_7(j)}$. PSX finds that a mapping ϕ_3 has a domain which overlaps $\overline{l_{1,-\infty}\phi_7(j)}$ and divides $\overline{l_{1,-\infty}\phi_7(j)}$ into two intervals $\overline{l_{1,-\infty}\phi_7(j)}$ and $\overline{\phi_7(j) c}$

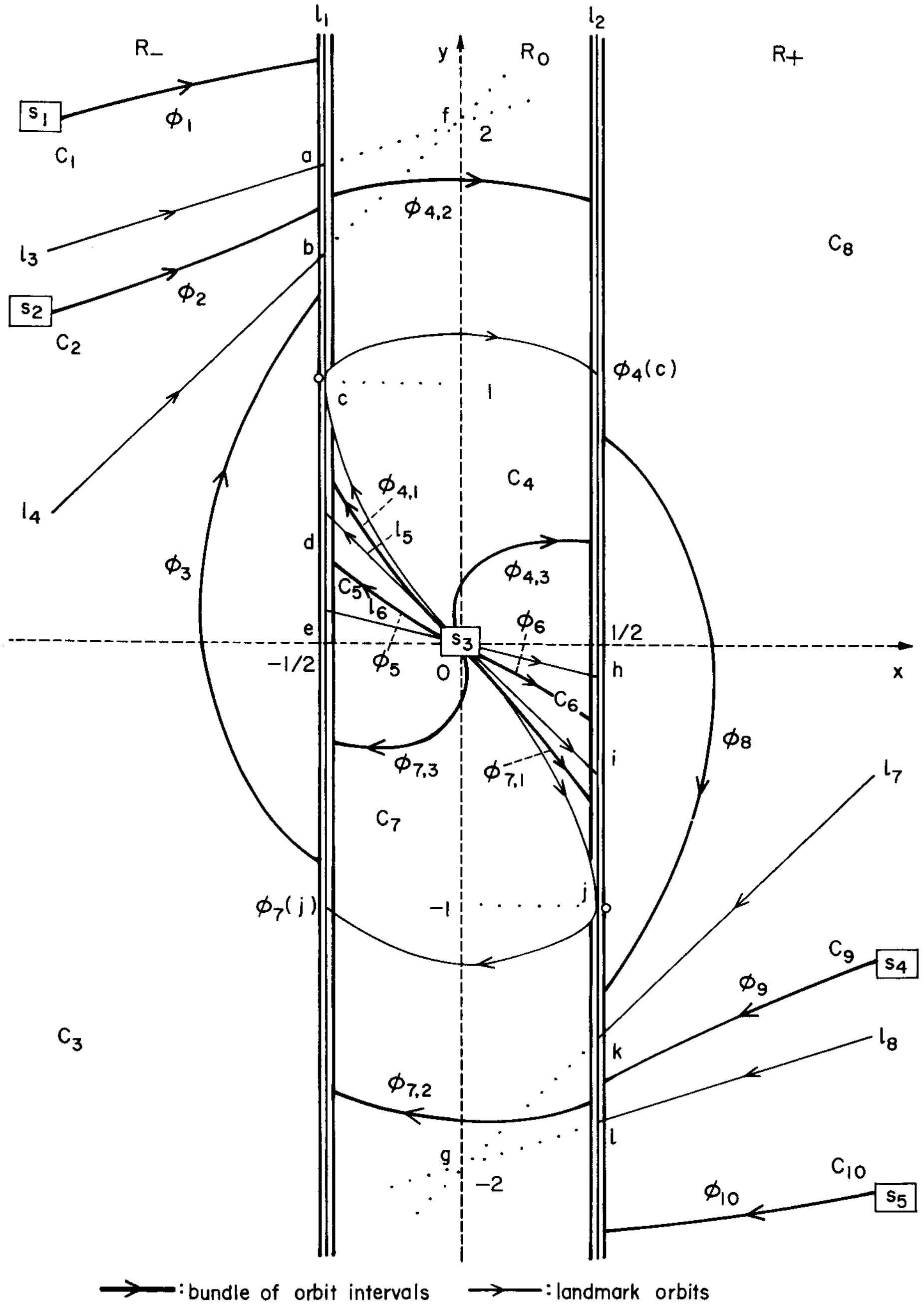


Figure 27 : Visualizing the Result of Local Analysis shown in Figure 26 for Piecewise Linear Approximation (12) of Van der Pol's Equation

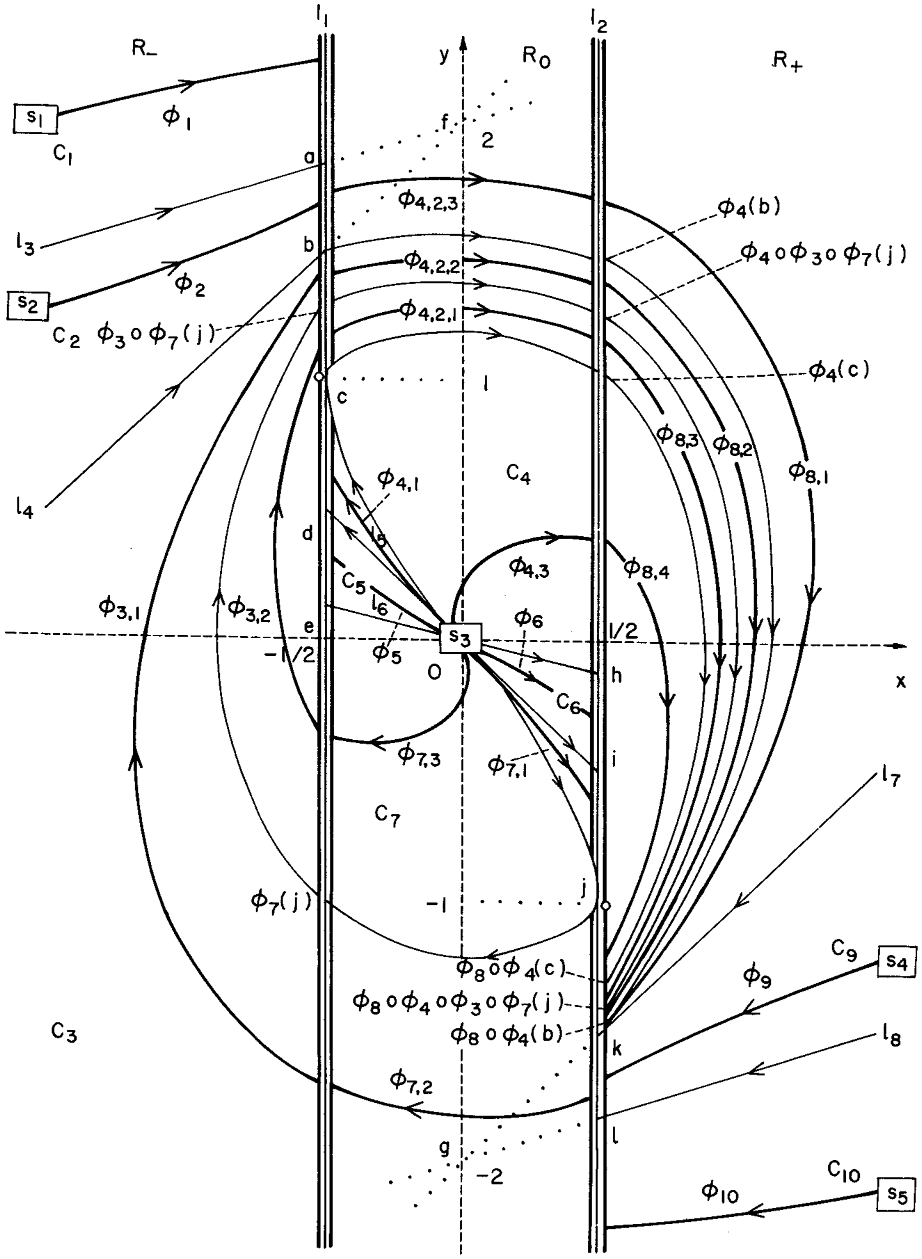


Figure 28: Global Analysis of Piecewise Linear Approximation (12) of Van der Pol's Equation

with $\phi_7(j)$ as a landmark. This also causes decomposition of ϕ_3 into

$$\phi_{3,1} : \overline{l_{1,-\infty}\phi_7(j)} \rightarrow \overline{\phi_3 \circ \phi_7(j)\, b},\ \phi_{3,2} : \overline{\phi_7(j)\, c} \rightarrow \overline{c\phi_3 \circ \phi_7(j)}.$$

Now comprsition $\phi_{3,1} \circ \phi_{7,2}$ is defined and PSX updates the annotation to the departing side of interval $\overline{l_{2,-\infty} j}$ to

$$\xrightarrow{\phi_{7,2}} \overline{l_{1,-\infty}\phi_7(j)} \xrightarrow{\phi_{3,1}} \overline{\phi_3 \circ \phi_7(j)\, b}.$$

By repeating extension of annotations this way, PSX eventually finds out a contracting recursive mapping, which entails the existence of an attracting limit cycle. The annotation to the departing side of interval $\overline{l_{2,-\infty} j}$ is :

$$\xrightarrow{\phi_{7,2}} \overline{l_{2,-\infty}\phi_7(j)} \xrightarrow{\phi_{3,1}} \overline{\phi_3 \circ \phi_7(j)\, b} \xrightarrow{\phi_{4,2,2}} \overline{\phi_4 \circ \phi_3 \circ \phi_7(j)\, \phi_4(b)} \xrightarrow{\phi_{8,2}} \overline{\phi_8 \circ \phi_4(b)\, \phi_8 \circ \phi_4 \circ \phi_3 \circ \phi_7(j)}$$

where,

$$l_{2,-\infty} \prec_{c_2} \cdots \prec_{c_2} k \prec_{c_2} \phi_8 \circ \phi_4(b) \prec_{c_2} \phi_8 \circ \phi_4 \circ \phi_3 \circ \phi_7(j) \prec_{c_2} \phi_8 \circ \phi_4(c) \prec_{c_2} j \prec_{c_2} \cdots \prec_{c_2} l_{2,\infty}$$

and,

$$l_{1,-\infty} \prec_{c} \phi_7(j) \prec_{c_1} e \prec_{c_1} \cdots \prec_{c_1} l_{1,\infty}$$

PSX goes further, ending up in finding that all orbits except the origin eventually pass over interval $\overline{\phi_8 \circ \phi_4(b)\, \phi_8 \circ \phi_4 \circ \phi_3 \circ \phi_7(j)}$ in a finite amount of time, and "absorbed" into the attracting limit cycle found above. Figure 28 illustrates this.

7. 2 Analysis of Behavior of Unstable Multivibrator

Consider an unstable multivibrator shown in Figure 29. We obtain the circuit equation :

$$\left\{\begin{array}{l} \mathrm{R}_{1,0} : \left\langle x \leq v_\sigma \,\middle|\, \dot{x} = \dfrac{v_{CC} - x}{C_1 R_2},\ \dot{y} = \dfrac{-v_{CC} + v_\sigma - y}{C_2 R_4} \right\rangle, \\ \mathrm{R}_{0,1} : \left\langle y \leq v_\sigma \,\middle|\, \dot{x} = \dfrac{-v_{CC} + v_\sigma - x}{C_1 R_1},\ \dot{y} = \dfrac{v_{CC} - y}{C_2 R_3} \right\rangle, \\ \quad \cdots \end{array}\right\} \tag{13}$$

if we model the two transistors as ideal switching elements, where variables x and y denote voltage across capacitors C_1 and C_2, respectively. As for other parameters,

$v_\sigma = 0.7$, $v_{CC} = 5$, $\{C_1, C_2, R_1, R_2, R_3, R_4\} > 0$.

For simplicity, let us assume that we have only two linear regions : $\mathrm{R}_{1,0}$ and $\mathrm{R}_{0,1}$. $\mathrm{R}_{1,0}$

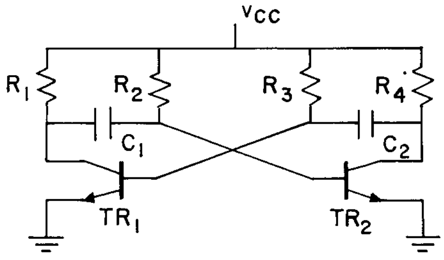


Figure 29 : Unstable Multivibrator

stands for a state in which TR_1 is ON and TR_2 is OFF. $R_{0,1}$ stands for a state in which the activation pattern of the two transistors are exactly the converse of that in $R_{1,0}$. Unlike the piecewise linear approximation of Van der Pol's equation in the previous example, these two linear regions do overlap and hence the uniqueness of the solution does not bold.

PSX first divides linear regions $R_{1,0}$ and $R_{0,1}$ into cells, as shown in Figure 30. Local analysis identifies local flow in each cell, as shown in Figure 31. During global analysis, PSX finds the following contracting recursive mapping

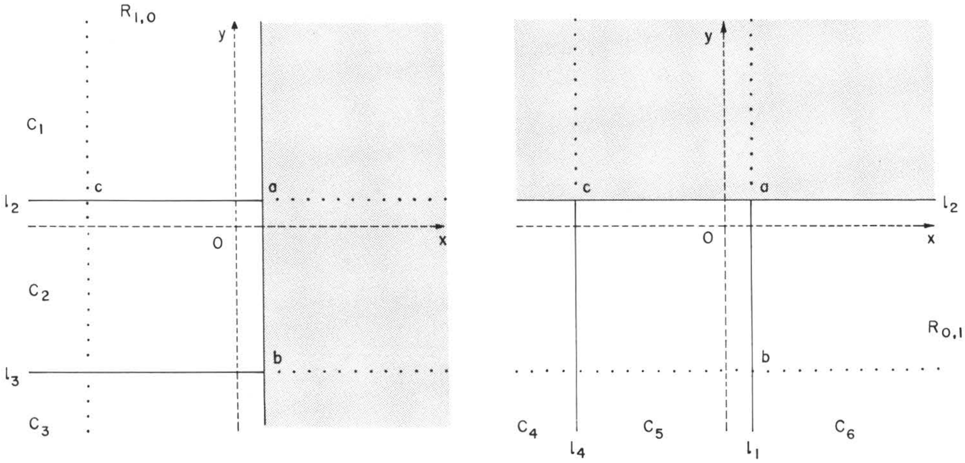


Figure 30 : Dividing the Phase Space for Piecewise Linear Differential Equation (13) for an Unstable Multivibrator, into Cells

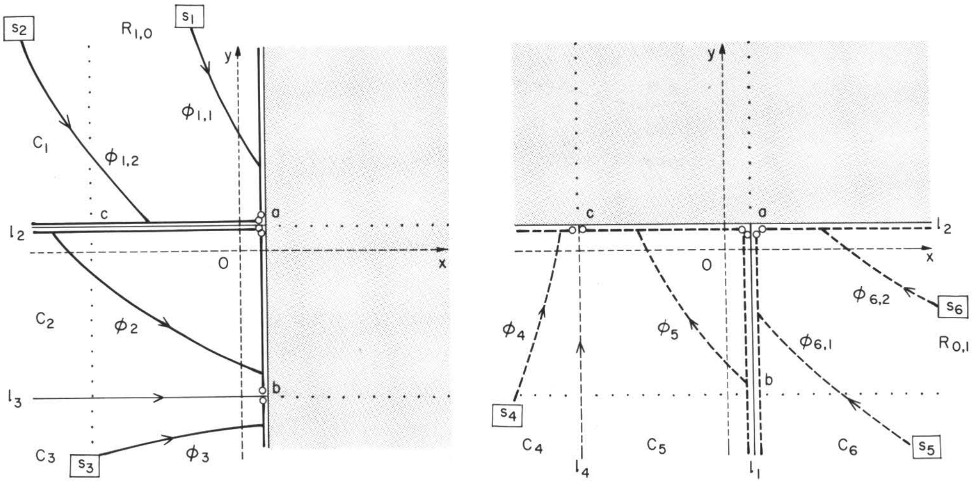


Figure 31 : Result of Local Analysis of Piecewise Linear Differential Equation (13) for an Unstable Multivibrator

$\phi_{5,2}\circ\phi_2 : \overline{l_{2,-\infty}a} \rightarrow \overline{\phi_5(b)a} \subset \overline{l_{2,-\infty}a}.$

This convinces PSX of the existence of an attracting bundle of orbits (in weaker sense) that is transverse to $\overline{\phi_5(b)a}$ where,

$l_{2,-\infty} \prec_{c2} c \prec_{c2} \phi_5(b) \prec_{c2} a \prec_{c2} l_{2,\infty}.$

Furthermore, PSX finds out that all orbits except those originated from sources S_1 and S_6 that are located in place at infinity will eventually be absorbed into the attracting bundle of orbits found above. Figure 32 illustrates global analysis by PSX. The analysis correctly captures the qualitative behavior of the unstable multivibrator.

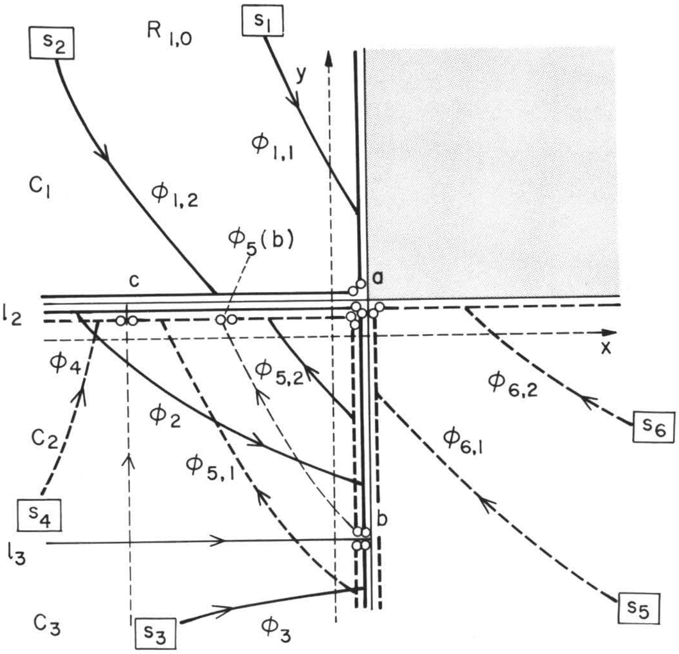


Figure 32: Global Analysis of (13) by PSX

8. Related Work

In general, there would be two extremes in the spectrum of approaches to qualitative phase space analysis. One extreme is to heavily rely on numerical computation: running a numerical simulator and interpreting the result in an intelligent way. Knowledge about dynamical systems is used to guide the search. Techniques from computer vision and computational geometry are used to interpret phase portraits partly drawn by a numerical simulator. Several authors took this approach. Yip [7] took this approach to analyze discrete systems and Sacks [6] presented a method for analyzing continuous systems. Abelson [1] provides a survey on activities at MIT. The advantage of this approach is that it is applicable to a wide variety of nonlinear systems. On the other hand, this approach has two disadvantages. First, controlling the search process is not easy because it is hard to derive clues about orbits from unrestricted nonlinear systems. Second, it is likey that numerical error is accumulated.

The other extreme is to rely more on analytical methods. The disadvantage is that we have to limit the class of problems since no universal method is known about the unirestricted class of nonlinear systems. In return, however, we can strongly guide the search process and the result is less likely to be affected by numerical errors.

The approach we have taken in this paper is closer to the second extreme. Work by Sacks [5] is also based on the same spirit. The general procedure employed in the two is similar : proceed from local analysis to global analysis. Our method of dividing phase space into linear region is more general that Sacks' : we allowed division by any polyline while Sacks allowed division only by straight lines that are parallel to x- or y-axis.

A more essential difference is the internal representation used for analysis. Sacks used transition graphs as an internal representation. With transition graphs alone, it is

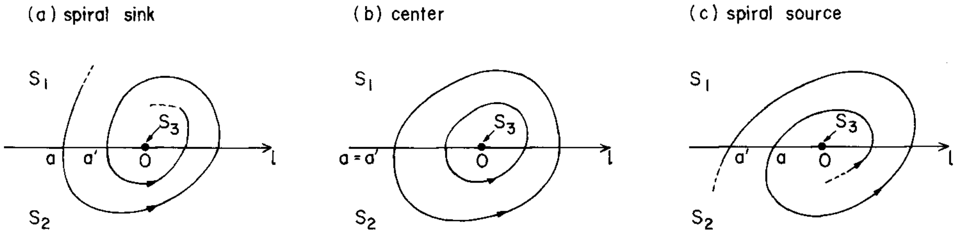


Figure 33 : Spiral Sink, Spiral Source, and Center

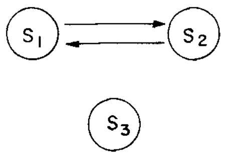


Figure 34 : Transition Graph Representation of Behaviors Shown in Figure 33

(a) f such that $f(\overline{Oa}) \subset \overline{Oa}$

(b) f such that $f(\overline{Oa}) = \overline{Oa}$

(c) f such that $f(\overline{Oa}) \supset \overline{Oa}$

Figure 35 : Representing the Behavior in Figure 33 as a Collection of Mappings

impossible to make an important distinction among spiral sinks, spiral sources, and centers, as shown in Figure 33. All of them reduce to a single transition graph as shown in Figure 34. In contrast, it is possible to distinguish the three by the mapping representation employed in this paper, as shown in Figure 35 (a) through (c). The final advantage of the representation used here is that it guides numerical computation. It is straightforward to derive information for further quantitative computation from our representation. It is not so in transition graphs.

9. Concluding Remark

In this paper, we have presented a program PSX that explores phase portraits of two-dimensional piecewise linear differential equations. The program is based on four simple ideas : focusing on bundle of orbits rather than single orbits, abstracting orbits as mapping between local coordinates, inferring the local flow in a given cell by only looking at the type and location of singular segments on the boundary, and extending mappings defined for local flow to derive information about global behavior. We have described details of algorithms in which qualitative and quantitative analysis interact in a cooperative fashion. We have estimated the number of ambiguities arising in qualitative analysis.

Although the application area of PSX is currently limited, we believe the presented method provides insights about more complex phase portraits.

Acknowledgment

We would like to thank Mr. Yoichi Shimada for his discussions. We also thank Mr. Kenji Satoh for writing a program which counts the number of ambiguities based on the formula given in this paper.

References

1) Harold Abelson, Michael Eisenberg, Matthew Halfant, Jacob Katzenelson, Elisha Sacks, Gerald J. Sussman, Jack Wisdom, and Kenneth Yip. Intelligence in scientific computing. *Communications of the ACM*, 32 : 546-562, 1989.

2) John Guckenheimer and Philip Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields.* Springer-Verlag, 1983.

3) Morris W. Hirsch and Stephen Smale. *Differential Equations, Dynamical Systems, and Linear Algebra.* Academic Press, 1974.

4) Toyoaki Nishida. Phase spase explorer PSX reference manual. (forthcoming, in Japanese), 1990.

5) Elisha Sacks. Automatic qualitative analysis of ordinary differential equations using piecewise

linear approximations. Technical Report 1031, MIT Artificial Intelligence Laboratory, 1988.

6) Elisha Sacks. Qualitative analysis of continuous dynamic systems by intelligent numeric experimentation. presented at Third Qualitative Physics Workshop, Stanford, August 9-11, 1989.

7) Kenneth Man-kam Yip. Generating global behaviors using deep knowledge of local dynamics. In *proceedings AAAI–88*, pages 280-285. American Association for Artificial Intelligence, 1988.