ISSN 1560-3547, Regular and Chaotic Dynamics, 2017, Vol. 22, No. 3, pp. 272–297. © Pleiades Publishing, Ltd., 2017.

Computational Method for Phase Space Transport with Applications to Lobe Dynamics and Rate of Escape

Shibabrat Naik[1*], Francois Lekien[2], and Shane D. Ross[1***]**

[1] *Engineering Mechanics Program, Department of Biomedical Engineering & Mechanics Virginia Tech, Blacksburg, VA-24061, USA*

[2] *École Polytechnique, Université Libre de Bruxelles B-1050 Brussels, Belgium*

Received September 28, 2016; accepted May 10, 2017

Abstract—Lobe dynamics and escape from a potential well are general frameworks introduced to study phase space transport in chaotic dynamical systems. While the former approach studies how regions of phase space get transported by reducing the flow to a two-dimensional map, the latter approach studies the phase space structures that lead to critical events by crossing certain barriers. Lobe dynamics describes global transport in terms of lobes, parcels of phase space bounded by stable and unstable invariant manifolds associated to hyperbolic fixed points of the system. Escape from a potential well describes how the critical events occur and quantifies the rate of escape using the flux across the barriers. Both of these frameworks require computation of curves, intersection points, and the area bounded by the curves. We present a theory for classification of intersection points to compute the area bounded between the segments of the curves. This involves the partition of the intersection points into equivalence classes to apply the discrete form of Green's theorem. We present numerical implementation of the theory, and an alternate method for curves with nontransverse intersections is also presented along with a method to insert points in the curve for densification.

MSC2010 numbers: 37J35, 37M99, 65D20, 65D30, 65P99

DOI: 10.1134/S1560354717030078

Keywords: chaotic dynamical systems, numerical integration, phase space transport, lobe dynamics

1. INTRODUCTION

Chaotic transport is a nonlinear dynamical systems framework to understanding and characterizing how regions of phase space move. These movements of phase space regions manifest as different dynamical behavior, thus it is relevant for systematic analysis of chaotic phenomena; see [1] for an overview. In this approach, lobe dynamics, introduced in [2], is a geometric approach for studying the global transport in 1- and 2-degree-of-freedom (DOF) Hamiltonian systems that can be reduced to two-dimensional maps. Moreover, there have been a few attempts at extending lobe dynamics to higher-dimensional Hamiltonian systems with small perturbation where the existence of a normally hyperbolic invariant set and its associated codimension-1 stable and unstable manifolds can be guaranteed; see [3–7] for details. In this article, we present methods that are motivated by lobe dynamics in 2-DOF Hamiltonian systems that can be reduced to a two-dimensional map by using a suitable Poincaré section, or a return map, or a time-T map. The framework of chaotic transport arises in applications of escape and transition rate in classical mechanics and dynamical astronomy [8–11], loss of global integrity in ship capsize [12], reaction and dissociation rate in chemical reactions [4, 13, 14], chaotic advection in fluid mechanics [15, 16], wake generation behind a cylinder in a fluid flow [17], and transport in geophysical flow [18].

*E-mail: shiba@vt.edu

**E-mail: lekien@ulb.ac.be

***E-mail: sdross@vt.edu

Following the developments in [2, 19], lobe dynamics states that the two-dimensional phase space M of the Poincaré map f can be partitioned into certain regions with boundaries consisting of parts of the boundary of M (which may be at infinity) and/or segments of stable and unstable invariant manifolds of hyperbolic fixed points, $p_i, i = 1, \ldots, N$, as shown schematically in Fig. 1a. When the manifolds $W^u_{p_i}$ and $W^s_{p_j}$ are followed out on a global scale, they often intersect in primary intersection points or *pips*, for example, $\{q_1, q_2, q_3, q_4, q_5, q_6\}$, as in Fig. 1b, and secondary intersection points or *sips*, for example, $\{q_7, q_8, q_9, q_{10}, q_{11}, q_{12}\}$ in Fig. 1b. More precisely, an intersection point is called a pip if the curves connecting the point and the hyperbolic points intersect only at that point and is called a sip otherwise [2]. These intersections allow one to define boundaries between regions $\{R_i\}$, as illustrated in Fig. 1b. Moreover, the transport between regions of phase space can be completely described by the dynamical evolution of parcels of phase space enclosed by segments of the stable and unstable manifolds called *lobes*.

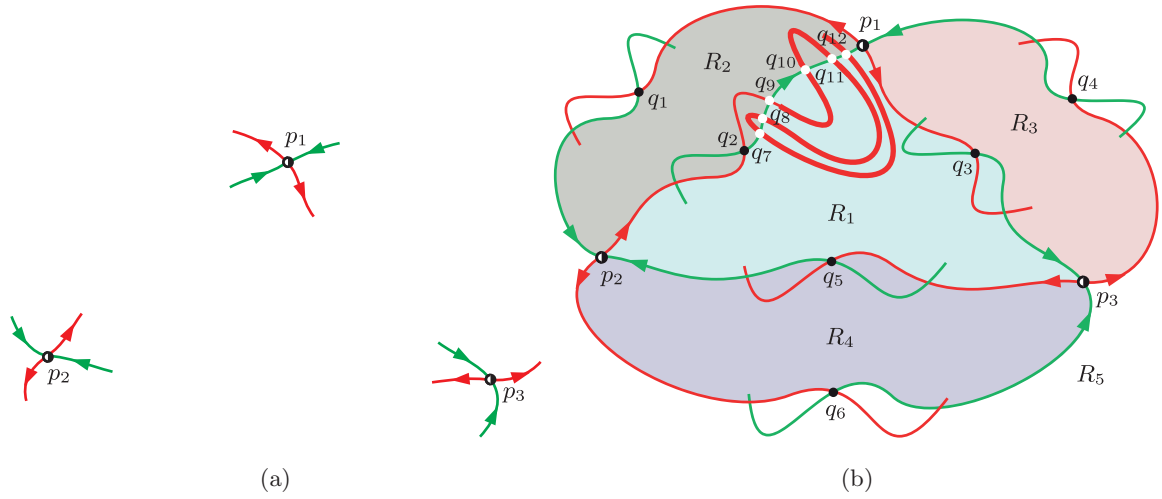


Fig. 1. (a) Pieces of the local unstable and stable manifolds, $W^u(p_i)$ (red) and $W^s(p_i)$ (green) of saddle fixed points $\{p_i\}$. (b) When the manifolds $W^u(p_i)$ and $W^s(p_i)$ are followed out on a global scale, they often intersect in primary intersection points $\{q_i\}$. These intersections allow one to define boundaries between regions $\{R_i\}$.

Classically, invariant manifolds of hyperbolic fixed points are computed for as long as possible and lobes are extracted from the two curves [2, 9]. However, it is possible to get more lobes and hence to compute transport for a much longer period of time by integrating separatrices directly, particularly, in the complex case of multiple, self-intersecting lobes [9]. In this approach, the notions of pips and sips are not suitable, thus we propose a generalization of definition of lobes for two intersecting closed curves. When applying lobe dynamics to transport problems, it is of eventual interest to quantify the volume of phase space M that crosses the boundary with higher iterates of the Poincarè map f. This is typically achieved in multiple ways:

- Distributing test points inside the phase space region of interest and computing the iterate(s) required for each to escape. This will also require taking care of the re-entrainment due to the underlying turnstile (a pair of lobes that move points across a boundary) mechanism [2, 20–22].

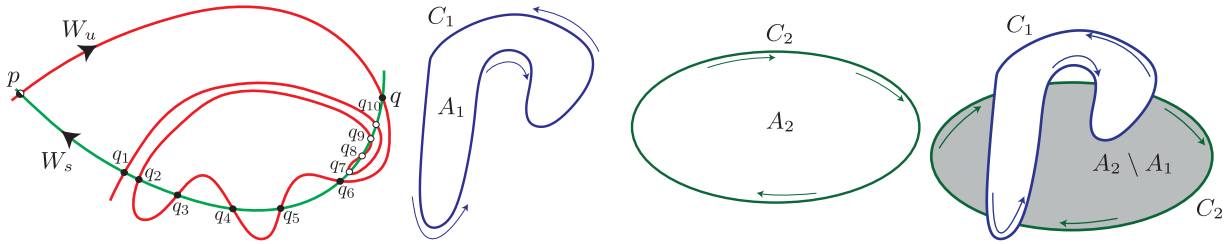
- Constructing a functional of the nonlinear system of the vector field which measures the area between the manifolds as parameterized by time. This is a semianalytical approach and remains valid near small perturbations, for example, the Melnikov method for mixing, stirring, optimal phase space flux in [23–26], and the action-integral method.

- Following the boundaries of separatrices/manifolds, as they evolve in time, and computing set operations with their preimages/images. This is a more general, but surely a difficult approach due to the stretching and folding of the curves that are involved in such computations [27–29].

Let us consider the lobes formed due to a homoclinic tangle as shown in Fig. 2a or due to a heteroclinic tangle formed in Fig. 1b. In both cases, the lobes are regions of phase space bounded by curves oriented in opposite directions since the segments bounding the lobes are stable and unstable manifolds of the hyperbolic fixed point, p. Topologically, the lobe areas can be represented by the difference of the area bounded by the individual curve. For example, let us consider two closed curves C_1 and C_2 in $\mathbb{R}^2$, oriented in counterclockwise and clockwise direction bounding the two-dimensional subsets of $\mathbb{R}^2$, A_1 and A_2, respectively, as in Fig. 2b, then the lobe area can be represented in the form of set difference of the area, that is, $A_2 \setminus A_1$ as shown in Fig. 2c. Thus, we want to determine the area $A_1 \setminus A_2$ and $A_2 \setminus A_1$ defined by these two curves and we note that the area $A_1 \setminus A_2$ can be computed once $A_2 \setminus A_1$ is known by using

$$A_1 \setminus A_2 = A_1 - (A_2 - A_2 \setminus A_1). \tag{1.1}$$

So, our approach is to develop methods that can calculate the area bounded by closed curves that are encountered in phase space transport.



(a) (b) (c)

Fig. 2. (a) shows schematically lobes, pips and sips that are formed in a homoclinic tangle. (b) shows an example of two closed curves, C_1 and C_2 in $\mathbb{R}^2$, oriented in counterclockwise and clockwise direction which bound the two-dimensional subsets of $\mathbb{R}^2$, A_1 and A_2, respectively. (c) shows schematically the area $A_2 \setminus A_1$ which represents a lobe area.

The main purpose of the paper is to expound on mathematical techniques useful for transport in systems that can be reduced to two-dimensional maps. We will present examples in the context of time-periodic and autonomous Hamiltonian systems, however, the methods are applicable in finite-time situations such as numerical simulation and experiments in which one wants to compute the amount of fluid transported from one region of phase space to another. In Sections 2–3, we derive the theory and numerical methods to separate the intersection points between the two curves into equivalence classes and derive the area of the lobes defined by the two curves. In Section 4, we present an alternative method that can be used when portions of the two curves have nontransverse intersections, which arise in the case of an iterated boundary parameterized using an intersection point. In Section 5, the numerical methods implemented as a software package *Lober* are applied to two example problems of interest in the dynamical systems literature: chaotic fluid transport in oscillating vortex pair flow and escape from a potential well in the capsize of a ship.

2. CURVE AREA

2.1. One-Dimensional Integrals for Areas

Let us define $A_1 = int(C_1)$ and $A_2 = int(C_2)$ as the area enclosed by C_1 and C_2, respectively, and denote this using the standard Lebesgue measure for μ in $\mathbb{R}^2$ as $[A] = \mu(A)$. The area of each region can be computed as

$$[A_i] = \iint_{A_i} dA = \frac{1}{2} \oint_{C_i} ydx - xdy\,, \tag{2.1}$$

by applying Green's theorem to the vector field

$$\bar{f} = \begin{pmatrix} x \\ y \end{pmatrix} \Longrightarrow \nabla \cdot \bar{f} = 2. \tag{2.2}$$

Equation (2.1) allows us to reduce the computation of the area of a complicated region to a one-dimensional integral over its boundary. Notice that the sign of the integral is to be reversed if the curves are oriented clockwise. Our hypothesis stating that the curves are oriented in a counterclockwise direction is equivalent to

$$\forall i : \oint_{C_i} ydx - xdy \geqslant 0. \tag{2.3}$$

2.2. Numerical Methods

We assume that the curves C_1 and C_2 are given in terms of a sequence of points (x_i, y_i). We want an exact evaluation of the integral in Eq. (2.1) for curves given by piecewise linear segments connecting (x_i, y_i) and (x_{i+1}, y_{i+1}). By defining

$$\begin{cases} x = x_i + t(x_{i+1} - x_i)\,, \\ y = y_i + t(y_{i+1} - y_i) \end{cases} \tag{2.4}$$

as parametric form of the segment, we have

$$\int_{x_i,y_i}^{x_{i+1},y_{i+1}} ydx - xdy = \int_0^1 (y_i x_{i+1} - x_i y_{i+1})dt = y_i x_{i+1} - x_i y_{i+1}. \tag{2.5}$$

As a result, the exact value of the contour integral for a polygon is given by

$$[A_i] = \frac{1}{2}\oint_{C_i} ydx - xdy = \frac{1}{2}\sum_{i=1}^{i=N} (y_i x_{i+1} - x_i y_{i+1})\,. \tag{2.6}$$

It is to be noted that for computing the contour integral the point $i = 1$ is repeated as $i = N + 1$, so that the curve is closed, and Eq. (2.6) represents a contour integral. This numerical method forms the core of the software package *Lober*[1)] along with the implementation of the numerical method derived in Sections 2, 3 and 4.

3. INTERSECTION POINTS AND LOBE AREA

Lobe dynamics is based on the geometry of a stable manifold, $W^s_{p_+}$, and an unstable manifold, $W^u_{p_-}$, of hyperbolic fixed points, p_+ and p_-, their intersection points, and areas enclosed by the segments of the invariant manifolds. We note that when $p_+ = p_- = p$, p is called a homoclinic point, and when $p_+ \neq p_-$, p_- and p_+ are called heteroclinic points. Following the definitions in [2, 3], a point $q_i \in W^u_p \cap W^s_p$ is called a *primary intersection point* (pip) if the segment $U[q_i, p]$ on W^u_p connecting p and q_i and the segment $S[q_i, p]$ on W^s_p connecting p and q_i intersects only at q_i, other than the point p. A point q_i is a *secondary intersection point* (sip) if there are other intersection points on the segment $U[q_i, p]$ and $S[q_i, p]$. If q_1 and q_2 are two adjacent pips, then the area enclosed by the segments $U[q_2, q_1]$ and $S[q_2, q_1]$ is called a *lobe*. This is shown in Fig. 2a, where $\{q_1, \ldots, q_6\}$ are pips, and $\{q_7, \ldots, q_{10}\}$ are sips. Furthermore, Fig. 1b shows the case when the unstable and stable manifolds are associated with two different hyperbolic fixed points. Thus, computing lobe areas requires knowing intersection points as accurately as possible and also identifying them as either pips or sips. Specifically, for closed intersecting curves C_1 and C_2 (see Fig. 2c) encountered in phase space transport problems, we separate the set of intersection points between the two curves into classes of equivalence. For two curves corresponding to the invariant manifolds of a hyperbolic fixed point, each class of equivalence corresponds exactly to the two *pips* and *sips* on the segment of the invariant manifold. Then we can compute the lobe area using the contour integral form of Eq. (2.6) which should be close to the lobe area given a well-resolved manifold.

[1)] This is available as an open-source repository in Github at `https://github.com/Shibabrat/curve_densifier` with implementation in C and additional wrapper scripts in MATLAB.

3.1. Intersection Points

In this section, we assume that there are only transverse intersections of the curves. A numerical algorithm for efficiently computing the intersection points is presented below. The two curves are closed, so the number of intersection points must be even. We compute the $2N$ intersection points p_i between the two curves C_1 and C_2. The unit tangent vector to the curve C_j at point p_i is denoted $\mathbf{1}_j(p_i)$. For each intersection point, p_i, we define

$$\rho(p_i) = \mathrm{sgn}(\sin(\theta))\frac{||\mathbf{1}_1(p_i) \times \mathbf{1}_2(p_i)||}{\sin(\theta)} = \mathrm{sgn}(\sin(\theta))||\mathbf{1}_1(p_i)||||\mathbf{1}_2(p_i)||, \tag{3.1}$$

where θ is the angle between the tangents $\mathbf{1}_1(p_i)$ and $\mathbf{1}_2(p_i)$, as shown in Fig. 3. The quantity $\rho(p_i)$ represents the *orientation* of a given point p_i. We note that when the intersections are transverse, the denominator of Eq. (3.1) is nonzero and $\rho(p_i) \in \{-1, +1\}$. Figure 4 shows two curves, their intersection points, and the value of $\rho(p_i)$ for each point p_i. The closed curves shown in Fig. 4 can be obtained by extracting the lobes from the intersection of stable and unstable manifolds or intersection of a cylindrical manifold with a plane. It is to be noted that the orientation of the closed curves is related to the geometry of the invariant manifolds, and as such one might have a counterclockwise and a clockwise oriented curve.

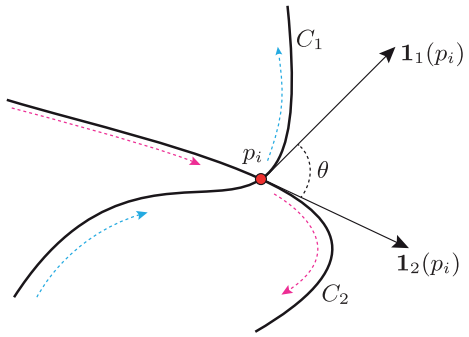


Fig. 3. Parameters involved in computing the orientation of an intersection point, p_i, when traversing along the curves in the direction shown in dashed arrows.

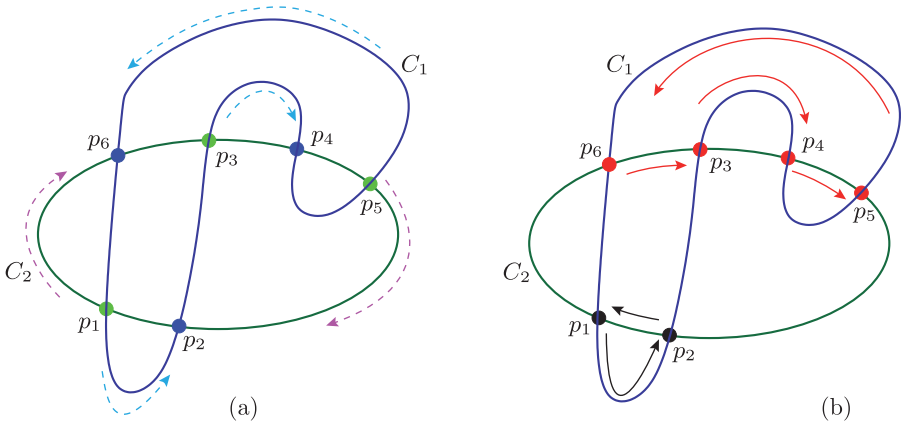


Fig. 4. Example of closed curves oriented in counterclockwise (cyan arrows) and clockwise direction (magenta arrows) with transverse intersections at $p_1, p_2, \ldots, p_6$. (a) Green intersection points have $\rho(p_i) = 1$. Blue intersection points have $\rho(p_i) = -1$. (b) Intersection points of the same color belong to the same equivalence class. The arrows represent the one-to-one and onto relationship σ between the intersection points and the points joining the same equivalence class can either be a *knob* (black arrows) or *handle* (red arrows).

3.2. Classes of Intersection Points

The segments of the curve between intersection points are most important to our computation, so we define $C_1^+[p_i, p_j]$ and $C_2^+[p_i, p_j]$ as the counterclockwise segments of, respectively, C_1 and C_2 between the points p_i and p_j. Similarly, $C_1^-[p_i, p_j]$ and $C_2^-[p_i, p_j]$ are the clockwise segments of, respectively, C_1 and C_2 between the points p_i and p_j. We define the positive and negative adjacency on C_1 by

$$A^{\pm}_{C_1}(p_i, p_j) = \begin{cases} 1 & \text{when } 1 \leqslant k \leqslant 2N : k \neq i \text{ and } k \neq j \Longrightarrow p_k \notin C_1^{\pm}[p_i, p_j], \\ 0 & \text{otherwise,} \end{cases} \tag{3.2}$$

and the adjacency on C_2 as

$$A^{\pm}_{C_2}(p_i, p_j) = \begin{cases} 1 & \text{when } 1 \leqslant k \leqslant 2N : k \neq i \text{ and } k \neq j \Longrightarrow p_k \notin C_2^{\pm}[p_i, p_j], \\ 0 & \text{otherwise.} \end{cases} \tag{3.3}$$

The geometric interpretation of adjacency is that when traversing a curve in counterclockwise or clockwise (that is, positive or negative sense) direction, the point p_j with adjacency value of 1 is the point next to the point p_i.

The objective of this section is to determine a generalized notion of a lobe. Intuitively, lobes are bounded by "segments of the curves C_1 and C_2 turning in opposite directions on each curve". We formalize this idea by defining the signed adjacency

$$\gamma(p_i, p_j) = \frac{\rho(p_i) + 1}{2} A^+_{C_1}(p_i, p_j) - \frac{\rho(p_i) - 1}{2} A^-_{C_2}(p_i, p_j). \tag{3.4}$$

which becomes $\gamma(p_i, p_j) = A^+_{C_1}(p_i, p_j)$ for $\rho(p_i) = 1$ and $\gamma(p_i, p_j) = A^-_{C_2}(p_i, p_j)$ for $\rho(p_i) = -1$. Tables 1, 2, and 3 show the value of the functions $A^+_{C_1}$, $A^-_{C_2}$, and γ, respectively, for the example of intersecting curves shown in Fig. 4.

Table 1. Function $A^+_{C_1}$ for the intersection points between the curves of Fig. 4. The lines and columns correspond, respectively, to the first and second argument of $A^+_{C_1}$.

$A^+_{C_1}(\downarrow, \rightarrow)$	p_1	p_2	p_3	p_4	p_5	p_6
p_1	0	1	0	0	0	0
p_2	0	0	1	0	0	0
p_3	0	0	0	1	0	0
p_4	0	0	0	0	1	0
p_5	0	0	0	0	0	1
p_6	1	0	0	0	0	0

In order to separate the intersection points in disjoint classes, we define an equivalence relation between the intersection points by

Definition 1. *$p_i \sim p_j$ iff there exists a sequence of K intersection points p_{α_k} such that*

$$\begin{cases} p_{\alpha_1} = p_i, \\ p_{\alpha_K} = p_j, \\ \Pi_{k=1}^{K-1} \gamma(p_k, p_{k+1}) = 1. \end{cases} \tag{3.5}$$

Table 2. Function $A^-_{C_2}$ for the intersection points between the curves of Fig. 4. The lines and columns correspond, respectively, to the first and second argument of $A^-_{C_2}$.

$A^-_{C_2}(\downarrow,\rightarrow)$	p_1	p_2	p_3	p_4	p_5	p_6
p_1	0	0	0	0	0	1
p_2	1	0	0	0	0	0
p_3	0	0	0	1	0	0
p_4	0	0	0	0	1	0
p_5	0	1	0	0	0	0
p_6	0	0	1	0	0	0

Table 3. Function γ for the intersection points between the curves of Fig. 4. The lines and columns correspond, respectively, to the first and second argument of γ.

$\gamma\,(\downarrow,\rightarrow)$	p_1	p_2	p_3	p_4	p_5	p_6
p_1	0	1	0	0	0	0
p_2	1	0	0	0	0	0
p_3	0	0	0	1	0	0
p_4	0	0	0	0	1	0
p_5	0	0	0	0	0	1
p_6	0	0	1	0	0	0

Thus, the sequence of K intersection points bounds a domain while traversing the curve in clockwise or counterclockwise direction. We have the following properties:

Theorem 1 (Reflexivity of $\sim$).

$$\forall i : p_i \sim p_i. \tag{3.6}$$

Proof. Letting $K = 1$ and $\alpha_1 = i$ in Eq. (3.5) gives the desired result. □

Theorem 2 (Symmetry of $\sim$).

$$\forall i, j : p_i \sim p_j \Longrightarrow p_j \sim p_i. \tag{3.7}$$

Proof. We notice that for any intersection point p_i there is one and only one intersection point p_j satisfying $A^+_{C_1}(p_i, p_j) = 1$. This is a consequence of the fact that the intersections are transverse and that there are not any self-intersections. There must be at least one segment of the curve C_1 leaving p_i (based on a counterclockwise orientation of C_1). If there were more than one segment leaving this point, the curve C_1 would self-intersect. Similarly, there is also one and only one intersection point satisfying $A^-_{C_2}(p_i, p_k) = 1$. Since $\rho(p_i) \in \{-1, 1\}$, there is only one intersection point p_q ($q = j$ or k) such that $\gamma(p_i, p_q) = 1$. We define $\sigma(p_i)$ as the unique intersection point satisfying

$$\gamma(p_i, \sigma(p_i)) = 1. \tag{3.8}$$

Notice that

$$p_i \neq p_j \Longrightarrow \sigma(p_i) \neq \sigma(p_j), \tag{3.9}$$

and σ is therefore invertible. We construct the sequence (p_{α_i}) using

$$\begin{cases} p_{\alpha_1} = p_i, \\ p_{\alpha_k+1} = \sigma(p_{\alpha_k}). \end{cases} \tag{3.10}$$

Notice that the sequence (p_{α_i}) satisfies the third condition of Eq. (3.5) for any length. The number of intersection points is finite, so there must be an integer K such that p_{α_K} is identical to p_{α_k} for $k < K$. We cut the sequence at the smallest possible K, so there is only one repeated element in the sequence. The repeated element must be p_{α_1} because σ is invertible. As a result, the sequence (p_{α_i}) is the unique path from p_i to p_i that does not contain only p_i and does not contain repeated points except for the endpoints. To prove the relationship, we notice that $p_i \sim p_j \Longrightarrow$ the point p_j must be included in the unique sequence (p_{α_i}). As a result, we can take a subsequence from p_i to p_j in (p_{α_i}) and $p_j \sim p_i$. □

Table 4. Functions ρ and σ for the intersection points between the curves of Fig. 4. There are two equivalence classes for this example.

	$\rho(p)$	$\sigma(p)$	Class
p_1	$+1$	p_2	1
p_2	-1	p_1	1
p_3	$+1$	p_4	2
p_4	-1	p_5	2
p_5	$+1$	p_6	2
p_6	-1	p_3	2

Theorem 3 (Transitivity of $\sim$).

$$\forall i, j, k : \begin{cases} p_i \sim p_j \\ p_j \sim p_k \end{cases} \Longrightarrow p_i \sim p_k. \tag{3.11}$$

Proof. The hypothesis gives us one path from p_i to p_j and one path from p_j to p_k. We can combine these two paths to go from p_i to p_k and $p_i \sim p_k$. □

As a result, $\sim$ is an equivalence relation for the set P of intersection points.

3.3. Lobe Area

Thus, the lobe area can be computed by using the contour integral form of Eq. (2.6) for the intersection points that are equivalent under $\sim$. Let us define

$$S(p) = \begin{cases} \int_{C_1^+[p,\sigma(p)]} ydx - xdy & \text{if } \rho(p) = +1, \\ \int_{C_2^-[p,\sigma(p)]} ydx - xdy & \text{if } \rho(p) = -1, \end{cases} \tag{3.12}$$

where $\sigma(p)$ has been defined in the proof of Theorem 2. Then the lobe area can be computed using the following

Conjecture 1 (Lobe area).

$$[A_1 \backslash A_2] = \sum_{P_i \in P \backslash \sim} \left(\sum_{p \in P_i} S(p) \right), \tag{3.13}$$

where we define the classes of equivalence $\{P_i\} = P\backslash\sim$ as corresponding to subsets of P containing all the elements that are equivalent under $\sim$ (i.e., the quotient of P by $\sim$). We define the number of sets in $P\backslash\sim$ as n_P and we note that the equivalence classes constitute a partition of P. Thus, the P_i are disjoint and the union of their elements is P. For the example intersecting the curves in Fig. 4, there are $n_P = 2$ equivalence classes, $P_1 = \{p_1, p_2\}$ and $P_2 = \{p_3, p_4, p_5, p_6\}$.

It is to be noted that using $A^-_{C_1}$ and $A^+_{C_2}$ instead of $A^+_{C_1}$ and $A^-_{C_2}$ in the definition of γ (Eq. (3.4)) gives another equivalence relationship. The equivalence classes of the latter give the area $[A_2\backslash A_1]$ in a form identical to that used in Proposition 1. We also note that Proposition 1 could be written with a single sum over all the intersection points. However, we prefer to keep the sum of the equivalence classes because the sum over each intersection point in the same class corresponds to the area of a lobe, that is, an individual protuberance of A_1 outside A_2.

3.4. Numerical Implementation

3.4.1. Intersection points

The simplest approach of computing intersection points between two curves that are piecewise linear requires browsing every pair of linear segments (one on C_1 and one on C_2), and determining the possible intersection point between these two segments. This can become a computationally expensive operation for a manifold with million points since this approach requires solving a 2×2 linear system at each of the pair of piecewise linear segments and hence requires $O(N^2)$ time, where N is the number of line segments. Furthermore, the Bentley — Ottmann algorithm uses a line to sweep across N line segments with K intersection points in $O(N + K) \log N$ time, and randomized algorithms can solve the problem in $O(N \log N + K)$ time. However, the application of these computational geometry algorithms is much more complicated, and in the case of finite precision arithmetic leads to other challenges [30, 31]. Instead, we propose necessary and sufficient conditions to quickly check the existence of an intersection point between two piecewise linear segments.

Theorem 4. *If the segments* $[(x_1, y_1), (x_2, y_2)]$ *and* $[(x'_1, y'_1), (x'_2, y'_2)]$ *intersect, then we must have*

$$\big((y_2 - y_1)x'_1 - (x_2 - x_1)y'_1 - x_1y_2 + y_1x_2\big)\,\big((y_2 - y_1)x'_2 - (x_2 - x_1)y'_2 - x_1y_2 + y_1x_2\big) \leqslant 0. \quad (3.14)$$

Proof. The equation of the line passing through the two points (x_1, y_1) and (x_2, y_2) is

$$f(x, y) = (y_2 - y_1)x - (x_2 - x_1)y - x_1y_2 + y_1x_2 = 0. \quad (3.15)$$

The theorem states that if the segment $[(x'_1, y'_1), (x'_2, y'_2)]$ intersects the segment $[(x_1, y_1), (x_2, y_2)]$, then it must also intersect the line $f(x, y) = 0$. As a result, the endpoints of the segment $[(x'_1, y'_1), (x'_2, y'_2)]$ must be on opposite sides of the line and we have

$$f(x'_1, y'_1)f(x'_2, y'_2) \leqslant 0. \quad (3.16)$$

□

Theorem 4 gives us a necessary condition for an intersection between two segments. Notice that we can reverse the role of each segment in Theorem 4 and get another necessary condition. In addition, we have

Theorem 5. *The segments* $[(x_1, y_1), (x_2, y_2)]$ *and* $[(x'_1, y'_1), (x'_2, y'_2)]$ *intersect if and only if*

$$\big((y_2 - y_1)x'_1 - (x_2 - x_1)y'_1 - x_1y_2 + y_1x_2\big)\,\big((y_2 - y_1)x'_2 - (x_2 - x_1)y'_2 - x_1y_2 + y_1x_2\big) \leqslant 0, \quad (3.17)$$

and

$$\big((y'_2 - y'_1)x_1 - (x'_2 - x'_1)y_1 - x'_1y'_2 + y'_1x'_2\big)\,\big((y'_2 - y'_1)x_2 - (x'_2 - x'_1)y_2 - x'_1y'_2 + y'_1x'_2\big) \leqslant 0. \quad (3.18)$$

Proof. Theorem 4 directly implies one direction ($\Longrightarrow$) of the equivalence. To prove $\Longleftarrow$, notice that if the two equations are satisfied, the endpoints of the first segment are on each side of the line containing the second segment. The endpoints of the second segment are also on each side of the line containing the first segment. As a result, the two segments must intersect in at least one point. □

The two theorems above allow for a very fast and efficient algorithm to detect intersection points. Each segment on curve C_1 is checked for intersections with each segment on curve C_2. However, only the necessary condition given by Theorem 4 is checked. Only if this condition is satisfied is the second condition in Theorem 5 checked. If the necessary and sufficient condition is satisfied, then the intersection point is effectively computed.

3.4.2. Area bounded by a closed curve

We assume that the curves C_1 and C_2 are given in terms of a sequence of points (x_i, y_i). We want an exact evaluation of the integral in Eq. (3.12) for piecewise linear curves. By using Eq. (2.6) in the section above, Eq. (3.12) becomes

$$S(p) = \sum_{(x_i,y_i)\in\mathcal{C}^{\rho(p)}[p,\sigma(p)]} (y_i x_{i+1} - x_i y_{i+1}), \tag{3.19}$$

where $\mathcal{C} = C_1/C_2$ and in a form suitable for numerical implementation.

4. NONTRANSVERSE INTERSECTIONS

In this section, in contrast with Section 3, we *relax* the assumption that the two curves can only have transverse intersections. Thus, we allow nontransverse intersections of the curves, and the input curves can have common segments as shown in Fig. 5. This scenario arises in applying lobe dynamics on the Poincaré surface of section in 2 or more degrees of freedom, time-periodic 1-DOF system, 2-dimensional time-periodic fluid flow, and when multilobe, self-intersecting turnstiles are formed. This is shown in Fig. 6 and discussed in detail in [1, 9, 32]. In this case of multilobe turnstiles, transverse intersection cannot be guaranteed. In the case of transverse intersections, we used an algorithm based on primary intersection points and secondary intersection points to identify the boundary of each lobe, which was then used to compute the area enclosed by the boundary. The near-tangency (when the determinant of the linear system resulting from the pair of linear segments is close to 0) of intersection of the two curves creates both a confusion in the ordering of the intersection points and the position of the lobes as well as an enormous computational difficulty to extract the actual position of the intersection. Thus, to get around the nontransverse intersection, we resort to computing a function that is amenable to the tangle geometry of manifolds and lobes.

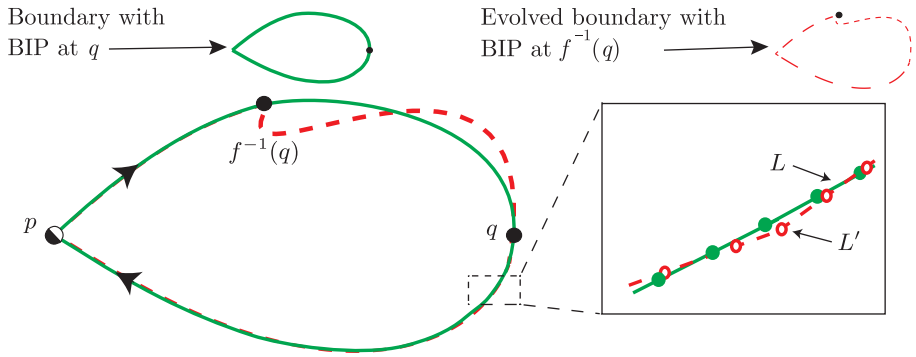


Fig. 5. Schematic showing the boundary (green solid curve) and its preimage (red dashed curve) with a boundary intersection point (BIP) at q and $f^{-1}(q)$, respectively. When the boundaries have near-tangent intersections, computing intersection points requires expanding/shrinking the points on the curve to avoid false lobes formed by the segments of curve L and L'.

4.1. Interior Function

Let us define the complex function

$$f_{z_0}(z) = \frac{1}{x - x_0 + i(y - y_0)}, \quad \text{where} \quad z = x + iy. \tag{4.1}$$

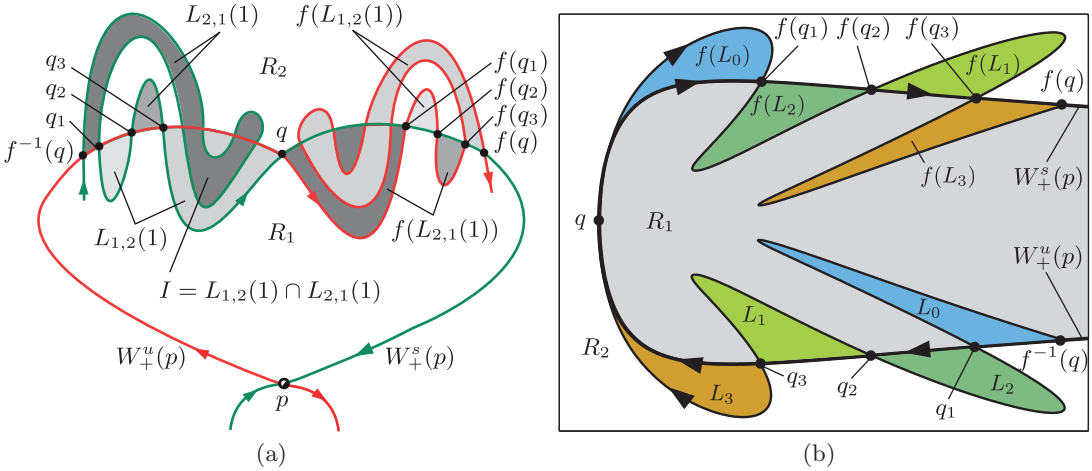


Fig. 6. (a) Schematic showing the geometry of multilobe, self-intersecting lobes. (b) An example of a multilobe turnstile that arises when considering the motion of a small body (for example, ejecta) in the field of a rotating asteroid; see [32] for details.

The integral of $f(z)$ over a closed curve is equal to $2i\pi$ times the number of turns that the closed curve makes around the point $x_0 + iy_0$ (to prove that, use the fact that $f(z)$ is analytic everywhere in the complex plane except at $x_0 + iy_0$ and use the residue theorem). We define

$$J_i(x_0, y_0) = Im\left\{\int_{C_i} \frac{dx + idy}{x - x_0 + i(y - y_0)}\right\}, \tag{4.2}$$

and

$$I_i(x_0, y_0) = \begin{cases} 1 & \text{if } J_i(x_0, y_0) < \pi, \\ -1 & \text{if } J_i(x_0, y_0) \geqslant \pi. \end{cases} \tag{4.3}$$

We note that $I_i(x_0, y_0)$ is negative when the point (x_0, y_0) is contained in the simple closed curve A_i, and is positive otherwise.

4.2. Lobe Area

In order to extract $[A_1 \backslash A_2]$ and $[A_2 \backslash A_1]$ from the shape of the curves, we define the following quantities:

$$Q_1 = \int_{C_1} I_2(x, y)\,(ydx - xdy) + \int_{C_2} I_1(x, y)\,(ydx - xdy)\,, \tag{4.4}$$

$$Q_2 = \int_{C_1} \frac{I_2(x, y) + 1}{2}\,(ydx - xdy) + \int_{C_2} \frac{I_1(x, y) + 1}{2}\,(ydx - xdy)\,, \tag{4.5}$$

$$Q_3 = \int_{C_1} \frac{I_2(x, y) - 1}{-2}\,(ydx - xdy) + \int_{C_2} \frac{I_1(x, y) - 1}{-2}\,(ydx - xdy)\,. \tag{4.6}$$

A quick look at the different paths in Fig. 7 reveals that

$$Q_1 = [A_1 \cup A_2] - [A_1 \cap A_2] = [A_1 \backslash A_2] + [A_2 \backslash A_1] \tag{4.7}$$

$$Q_2 = [A_1 \cup A_2], \tag{4.8}$$

$$Q_3 = [A_1 \cap A_2]. \tag{4.9}$$

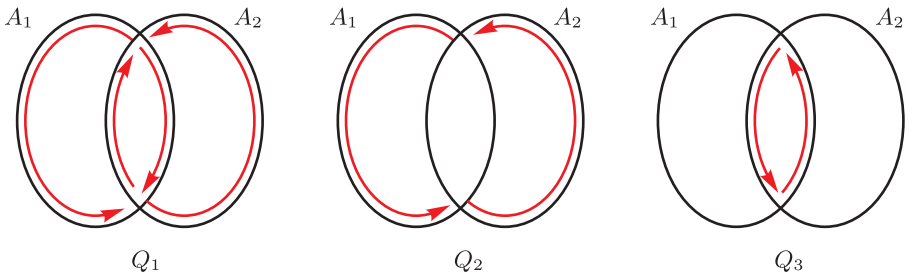


Fig. 7. Schematic view of the paths involved in the definition of Q_1, Q_2 and Q_3.

Notice that the three equations above are not linearly independent and *any* of them can give us the expected results. However, we compute Q_1, Q_2 and Q_3 and use the redundancy to minimize the error on the integrals and provide an approximation of the computational error. Since

$$[A_1 \backslash A_2] = [A_1 \cup A_2] - [A_2] = [A_1] - [A_1 \cap A_2], \tag{4.10}$$

and

$$[A_2 \backslash A_1] = [A_1 \cup A_2] - [A_1] = [A_2] - [A_1 \cap A_2], \tag{4.11}$$

we have

$$[A_1 \backslash A_2] = \frac{1}{2}(Q_2 - Q_3) + \frac{1}{2}([A_1] - [A_2]), \tag{4.12}$$

and

$$[A_2 \backslash A_1] = \frac{1}{2}(Q_2 - Q_3) + \frac{1}{2}([A_2] - [A_1]). \tag{4.13}$$

And using

$$[A_1] + [A_2] = [A_1 \cup A_2] + [A_1 \cap A_2], \tag{4.14}$$

we find

$$[A_1 \backslash A_2] = \frac{1}{2}Q_1 + \frac{1}{2}([A_1] - [A_2]), \tag{4.15}$$

and

$$[A_2 \backslash A_1] = \frac{1}{2}Q_2 + \frac{1}{2}([A_2] - [A_1]). \tag{4.16}$$

Our algorithm combines these two results and provides the following final answer:

$$[A_1 \backslash A_2] = \frac{1}{2}(A_1 - A_2) + \frac{1}{4}(Q_1 + Q_2 - Q_3), \tag{4.17}$$

$$[A_2 \backslash A_1] = \frac{1}{2}(A_2 - A_1) + \frac{1}{4}(Q_1 + Q_2 - Q_3), \tag{4.18}$$

$$\delta[A_1 \backslash A_2] = \delta[A_2 \backslash A_1] = \frac{1}{4}\left|Q_2 - Q_3 - Q_1\right|, \tag{4.19}$$

where the last equation gives the approximate error on the computed area.

4.3. Numerical Methods

The only technical difficulty in this algorithm is the numerical computation of the functions $J_i(x_0, y_0)$ in Eq. (4.2), and we present an algorithm for piecewise linear boundaries in Section 4.3.1. The computation of the integrals in Eqs. (4.4), (4.5), and (4.6) is similar to the numerical method used for the integral in Section 2.2. In addition, we present an algorithm to increase the number of points on the input curve close to the intersection points; we refer to the implementation of this algorithm in *Lober* as *densifier*, and its use is presented in Section 4.3.2. The algorithm in Section 4.3.2 has proved to increase the accuracy of this method without compromising the computational cost too much.

4.3.1. Computation of $J_i(x, y)$

In this section, we derive the exact value of the function J_i defined in Eq. (4.2) when the boundary of C_i is given as a sequence of piecewise linear segments. For each linear segment $[(x_1, y_1), (x_2, y_2)]$, we have

$$\begin{cases} x = x_1 + t(x_2 - x_1), \\ y = y_1 + t(y_2 - y_1). \end{cases} \tag{4.20}$$

Using $dx = (x_2 - x_1)\,dt$ and $dy = (y_2 - y_1)\,dt$, we obtain

$$\operatorname{Im}\left\{\frac{dx + i dy}{x - x_0 + i(y - y_0)}\right\} = \frac{(x_1 - x_0)(y_2 - y_1)\,dt - (y_1 - y_0)(x_2 - x_1)\,dt}{(x_1 - x_0 + t(x_2 - x_1))^2 + (y_1 - y_0 + t(y_2 - y_1))^2} \tag{4.21}$$

$$= \bar{\mathbf{1}}_z \frac{(\bar{x}_1 - \bar{x}_0) \times (\bar{x}_2 - \bar{x}_1)\,dt}{\|\bar{x}_2 - \bar{x}_1\|^2 t^2 + 2(\bar{x}_2 - \bar{x}_1) \cdot (\bar{x_1} - \bar{x}_0)t + \|\bar{x}_1 - \bar{x}_0\|^2}, \tag{4.22}$$

where $\bar{\mathbf{1}}_z$ is the unit normal vector in the z-direction and $\bar{x}_0 = (x_0, y_0)$, $\bar{x}_1 = (x_1, y_1)$, $\bar{x}_2 = (x_2, y_2)$.

$$\operatorname{Im}\left\{\frac{dx + i dy}{x - x_0 + i(y - y_0)}\right\} = \frac{\|\bar{u}\|^2 \|\bar{v}\|^2 \sin(\bar{u}, \bar{v})\,dt}{\|\bar{v}\|^2 t^2 + 2\|\bar{u}\|\|\bar{v}\|\cos(\bar{u}, \bar{v})\,t + \|\bar{u}\|^2}, \tag{4.23}$$

where

$$\begin{cases} \bar{u} = \bar{x}_1 - \bar{x}_0, \\ \bar{v} = \bar{x}_2 - \bar{x}_1. \end{cases} \tag{4.24}$$

We note that the discriminant of the denominator of Eq. (4.23) is

$$\rho = -4\|\bar{u}\|^2 \|\bar{v}\|^2 \sin^2(\bar{u}, \bar{v}) \leqslant 0. \tag{4.25}$$

As a result, we have

$$\operatorname{Im}\left\{\int_{[\bar{x}_1, \bar{x}_2]} \frac{dx + i dy}{x - x_0 + i(y - y_0)}\right\} = \int_0^1 \frac{\|\bar{u}\|^2 \|\bar{v}\|^2 \sin(\bar{u}, \bar{v})\,dt}{\|\bar{v}\|^2 t^2 + \|\bar{u}\|\|\bar{v}\|\cos(\bar{u}, \bar{v}) + \|\bar{u}\|^2} \tag{4.26}$$

$$= \frac{\sin(\bar{u}, \bar{v})}{|\sin(\bar{u}, \bar{v})|} \left[\tan^{-1}\left(\frac{\|v\|^2 t + \bar{u} \cdot \bar{v}}{\|\bar{u}\|\|\bar{v}\||\sin(\bar{u}, \bar{v})|}\right)\right]_0^1 \tag{4.27}$$

$$= \tan^{-1}\left(\frac{\|\bar{v}\|^2 + \bar{u} \cdot \bar{v}}{\bar{\mathbf{1}}_z \cdot (\bar{u} \times \bar{v})}\right) - \tan^{-1}\left(\frac{\bar{u} \cdot \bar{v}}{\bar{\mathbf{1}}_z \cdot (\bar{u} \times \bar{v})}\right). \tag{4.28}$$

We note that for $\sin(\bar{u}, \bar{v}) = 0$, the increment to the integral is zero. This is consistent with the equation above where the right-hand term is continuous for $\bar{u} \times \bar{v} = 0$ and vanishes.

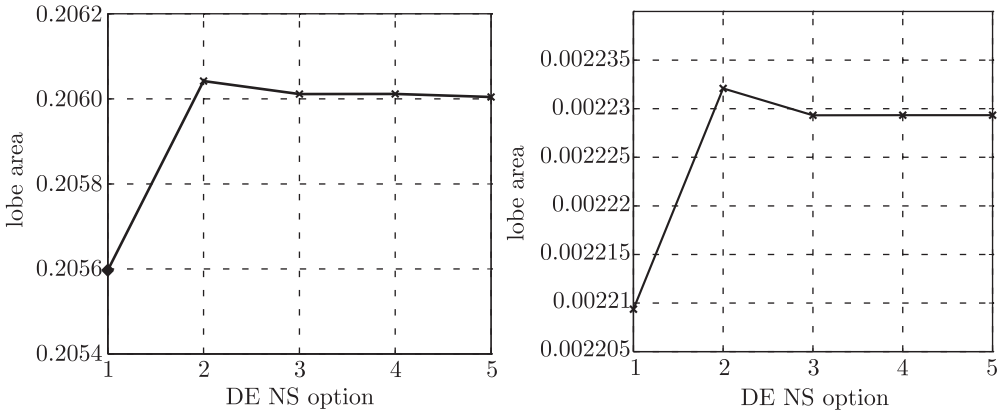
The approach in this section is similar to the point-in-polygon problem that is tackled in computational geometry (see [31]) by using the ray sweep method to check if the point is inside or outside a polygon. Although our method is mathematically satisfying, it may show higher computational cost as compared to the more efficient method of ray passing. This is due to the integral and trigonometric function evaluations that are involved, but our approach is to adopt a method amenable to high iterates of manifolds, which are curves with infinite folds and twists, at the expense of computational performance.

4.3.2. Curve densifier

There is a built-in *densifier* module in the light version of the package *Lober*, which adds points on the curves close to the intersection points. The densifier can be activated by adding the parameters `-DENS <nPass> <nDens>` at the command line. The arguments `<nPass>` and `<nDens>` give the number of passes to be performed and the number of points to add near each intersection at each pass, respectively.

The extra precision is always $i_r = {n_{dens}}^{n_{pass}}$, where n_{dens} = `<nDens>` and n_{pass} = `<nPass>`. In other words, the precision with $(n_{pass} = 1, n_{dens} = 1000)$ is the same as with $(n_{pass} = 3, n_{dens} = 10)$. For a constant i_r, the value of the two parameters n_{dens} and n_{pass} should minimize the computational time. Small n_{pass} means that fewer steps are necessary to densify the curve and can reduce the computational time. However, small n_{pass} usually implies large n_{dens} to maintain a constant i_r. Since the extra length of the curve is $n_{dens}n_{pass}$, the number of points increases rapidly if n_{pass} is too small and lengthens the computation. So there is an optimal n_{pass} that minimizes computation time, and this optimal n_{pass} parameter depends on the local curvature of the manifold under consideration.

We present a comparison of the *densifier* module in Fig. 8 by using this to compute the lobe area for two application problems. As demonstrated by the results, the accuracy in the lobe area does not improve dramatically beyond addition of 10^3 points, and is comparable to the default densifier in the light version of *Lober*.



(a) OVP flow (b) PCR3BP

Fig. 8. The numerically computed lobe area that used the *densifier* option. (a) oscillating vortex pair (OVP) flow as discussed in [2], and (b) planar circular restricted three-body problem (PCR3BP) as discussed in [33]. Along the x-axis, the indices denote the following *densifier* options: 1. `-DENS 0`, 2. `-DENS 1 10`, 3. `-DENS 3 10`, 4. `-DENS 5 100`, 5. without -DENS option which triggers the default densifier. The y-axis is the average of the entraining and detraining lobes for the parameters used in the references.

5. APPLICATION TO LOBE DYNAMICS AND RATE OF ESCAPE

In this section, we present applications of the computational approach to problems of transport in phase space that uses lobe dynamics for chaotic transport in fluid flow and rate of escape from a potential well in a ship capsize model. In essence, using the numerical methods presented in Sections 3 and 4, we demonstrate the qualitative and quantitative results that are relevant for these geometric methods of phase space transport.

5.1. Chaotic Transport in Fluid Flow

In this section, we apply the numerical methods to quantify phase space transport in the oscillating vortex pair (OVP) flow that was introduced in [2]. In this example from fluid dynamics, it is of interest to calculate the transport rate, in terms of the iterates of a two-dimensional map, of species S_1 from the inner core R_1 to the outside region R_2 shown in Fig. 9b. More generally and referring to Fig. 1b, this transport in phase space can be discussed using the language of lobe dynamics and transport in two-dimensional maps developed in [2, 21]. We will briefly summarize the OVP flow and the transport quantities that are relevant for the numerical experiments in our study.

In a two-dimensional, incompressible, inviscid fluid flow, the stream-function is analogous to the Hamiltonian governing particle motion and the domain of the fluid flow is identified as the

phase space. The OVP flow is generated by two counterclockwise rotating vortices under sinusoidal perturbations which can generate chaotic particle trajectories and cause regions of phase space to move across transport barriers. That is, the time-dependent velocity field, which is referred to as unsteady flow in fluid dynamics, can generate particle motion between the region with closed streamlines (shown as magenta in Fig. 9a) and the region outside the heteroclinic connection (shown in black in Fig. 9a). When the flow is time-dependent and has periodic forcing, a suitable approach is to construct a two-dimensional map of the flow by defining the Poincaré section at a certain phase of the flow and given by

$$\begin{aligned} &f : U^0 \to U^0, \\ \text{where} \qquad &U^0 = \{(x, y, \theta) \in \mathbb{R}^2 \times \mathbb{S}^1 : \theta = 0\}. \end{aligned} \tag{5.1}$$

Thus, when the fluid flow is time-dependent and has a sinusoidal perturbation, the hyperbolic fixed points of the two-dimensional map and the associated invariant manifolds can be used to define a boundary parameterized by an intersection point, which acts as a *partial barrier* between regions of the phase space. We note here that partial barriers are analogous to *separatrices* which are complete barriers of transport in the unperturbed fluid flow. When perturbation is applied to fluid flow, the partial barrier, however, acts as a boundary, shown as a black line connecting p_-, q, p_+ in Fig. 9b, across which transport occurs via specific regions of phase space enclosed by the invariant manifolds and lobes.

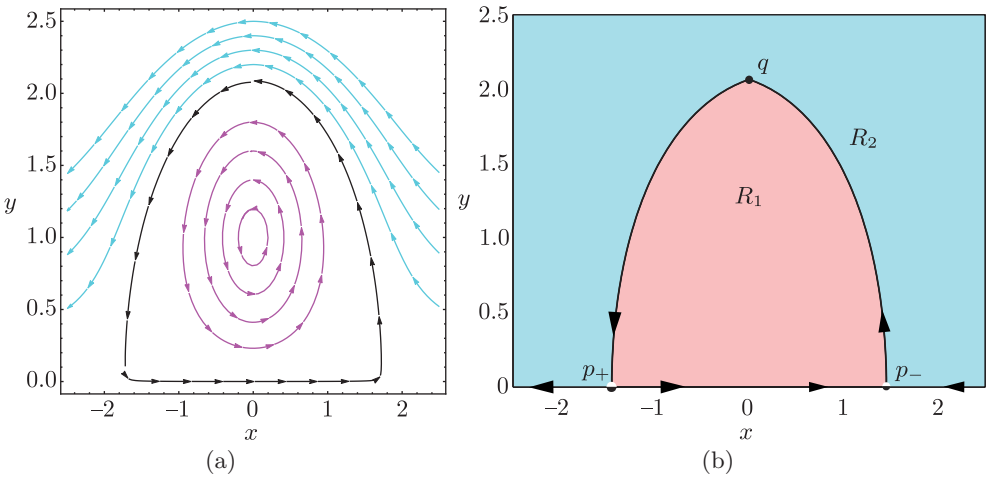


Fig. 9. (a) The closed streamlines and the free flow region for the unperturbed vortex pair flow. Even a simple sinusoidal perturbation changes the picture dramatically as the transport between the regions R_1 and R_2 becomes feasible due to chaotic particle trajectories. (b) The regions of interest R_1 and R_2 for studying the transport in the OVP flow. The transport between the regions R_1 and R_2 can be explained and quantified in terms of the turnstile mechanism.

We can cast the equation of motion of a passive particle in the OVP flow in the form of Hamilton's equations using the stream function as the Hamiltonian function and performing perturbation expansion of the velocity field (see [2] for details). Thus, the system can be expressed as

$$\begin{aligned} \frac{dx}{dt} &= f_1(x, y) + \epsilon g_1(x, y, t/\gamma; \gamma) + \mathcal{O}(\epsilon^2), \\ \frac{dy}{dt} &= f_2(x, y) - \epsilon g_2(x, y, t/\gamma; \gamma) + \mathcal{O}(\epsilon^2), \end{aligned} \tag{5.2}$$

where

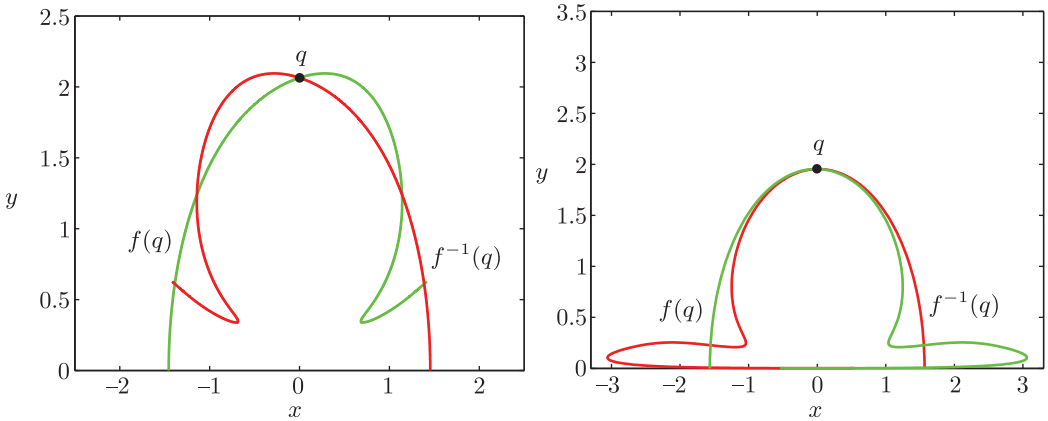
$$\begin{aligned} f_1 &= -\frac{y-1}{I_-} + \frac{y+1}{I_+} - 0.5, \\ f_2 &= \frac{x}{I_-} - \frac{x}{I_+} \end{aligned} \tag{5.3}$$

and

$$
\begin{aligned}
g_1 &= [\cos(t/\gamma) - 1]\left\{\frac{1}{I_-} + \frac{1}{I_+} - \frac{2(y-1)^2}{I_-^2} - \frac{2(y+1)^2}{I_+^2}\right\} \\
&\quad + (x/\gamma)\sin(t/\gamma)\left\{\gamma^2\left[\frac{y-1}{I_-^2} - \frac{y+1}{I_+^2}\right] + 1\right\} - 0.5, \\
g_2 &= 2x[\cos(t/\gamma) - 1]\left\{\frac{y-1}{I_-^2} + \frac{y+1}{I_+^2}\right\} \\
&\quad + (1/\gamma)\sin(t/\gamma)\left\{\frac{\gamma^2}{2}\left[\frac{1}{I_-} - \frac{1}{I_+}\right] - x^2\gamma^2\left[\frac{1}{I_-^2} - \frac{1}{I_+^2}\right] - y\right\},
\end{aligned}
\tag{5.4}
$$

and $I_\pm = x^2 + (y \pm 1)^2$.

The OVP flow has two nondimensional parameters which denote the circulation strength of the vortices, γ, and perturbation amplitude, ϵ. For a complete analysis of transport in such a flow, a combination of these two parameters is studied in [2], but we will present two cases as an application of the computational method implemented in *Lober*. Due to the complicated nature of the velocity field in Eq. (5.2), we numerically compute the invariant manifolds using the *globalization* technique (see [8, 33, 34]), and based on the benchmark algorithm in [35]. The two cases considered here show different manifold geometry, although topologically similar, as shown in Fig. 10. We call the difference in the geometry of *turnstile*, the pair of lobes formed by the segments of manifolds between q and $f^{-1}(q)$, the *near-orthogonal* (shown in Fig. 10a) and *near-tangent intersections* (shown in Fig. 10b), which are generated in these two cases ($\gamma = 0.5, 1.81$). Using the computed manifolds as inputs to *Lober*, we identify the intersection points and intersection areas, that is, pips and lobes, respectively, as shown in Fig. 11. We note that this approach is based on computing the area of the turnstile and its iterates to compute the transport rate.



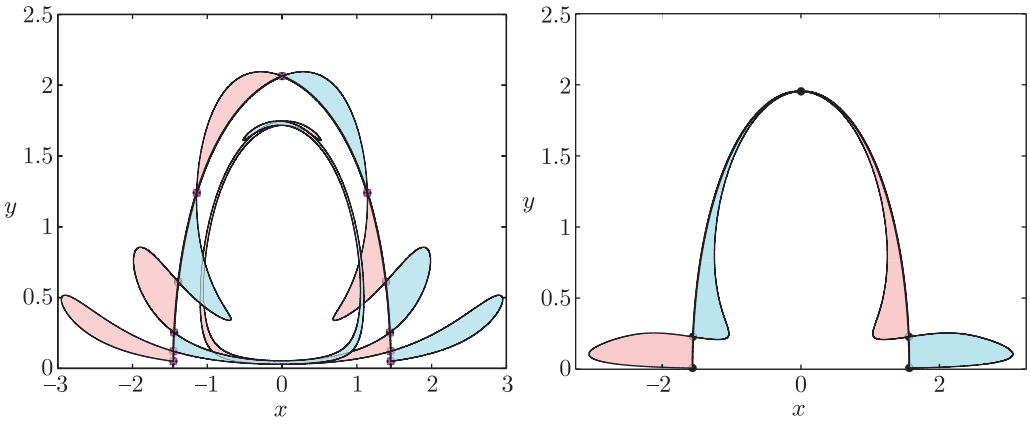
(a) $\gamma = 0.5$ (b) $\gamma = 1.81$

Fig. 10. Computed unstable (red) and stable (green) manifolds of the hyperbolic fixed points for the perturbation amplitude of $\epsilon = 0.1$, (a) $\gamma = 0.5$ and (b) $\gamma = 1.81$. The manifolds are shown for a qualitative comparison and a change in orientation for the γ values as predicted by Melnikov theory in [2].

Another approach in phase space transport is to consider the iterates of the boundary parameterized using an intersection point, say q as in Fig. 9b, and the segments of stable/unstable manifolds connected with the hyperbolic fixed points, say p_+ and p_- as in Fig. 9b. Thus, in the case of a time-dependent fluid flow, the boundary can be defined by

$$B(q) = S[q, p_+] \cup U[q, p_-], \tag{5.5}$$

where the point q is the *boundary intersection point* or *bip* which is the first intersection of the manifolds computed for a given initial phase of the map. The bip parameterizes the boundary which



(a) $\gamma = 0.5$ (b) $\gamma = 1.81$

Fig. 11. Output of *Lober* identifying the intersection points, that is, pips shown as magenta cross on filled black circles, intersection areas, that is, lobes shown as cyan and magenta filled area, and the boundary connecting the hyperbolic fixed points and the intersection point. The lobes colored as magenta map from region R_1 into R_2, and the ones that are colored cyan map from region R_2 to R_1 in successive iterates of the Poincaré map. For both cases, $\epsilon = 0.5$.

acts as a partial barrier when perturbation is added to the flow. Furthermore, the turnstile can be defined in terms of the bip and its preimage as

$$L_{1,2}(1) \bigcup L_{2,1}(1) = S[f^{-1}(q), q] \bigcup U[f^{-1}(q), q], \tag{5.6}$$

where $L_{i,j}(n)$ denotes a lobe in a region R_i that is mapped to a region R_j after n iterates, using the notation in [21], and we show a general schematic in Fig. 1. We note here that this definition of turnstile lobe uses the preimage of the bip, and requires computing the intersection of the boundary, Eq. (5.5), and its preimages under the Poincaré map. Thus, we can express the particles of species S_i (material in the region R_i as shown in Fig. 1) that gets transported across the partial barriers in terms of intersection areas of the turnstile lobe, and iterates of the Poincaré map (5.1). Following the theory described in [21], the quantities used in computation of transport rate are

- $a_{i,j}(n)$: Flux of species S_i from region R_i into region R_j on the n^{th} iterate.
- $T_{i,j}(n)$: Total amount of species S_i contained in region R_j after the n^{th} iterate.

We have the following approaches for computing these quantities:

1. *Boundary method:* Generate the boundary by selecting a point q as bip, and compute the preimages of this boundary along with the preimages of q $(f^{-1}(q), f^{-2}(q), \ldots)$. This can be computed by evolving in backward time, or executing *Lober* with an option to define the iterated boundary. This is shown in Fig. 12, and these curves are now used as input to the *light* option of *Lober* (see command options in Appendix).

2. *Lobe method:* Obtain the turnstile lobes $L_{1,2}(1), L_{2,1}(1)$ and images of entraining (in the sense of entering the region R_1 in the next iterate) turnstile lobe $L_{2,1}(1)$ or preimages of detraining turnstile lobe $L_{1,2}(1)$. These are obtained from *Lober* as closed curves and given as input to the *light* option of *Lober* to compute intersection areas since the lobes and its iterate can form nontransverse intersections and may require the *densifier* option.

Thus, using Lemma 2.3 and Theorem 2.5 in [21], the quantity $a_{2,1}(n)$ in terms of the turnstiles and intersection areas can be expressed as

$$a_{2,1}(n) = T_{2,1}(n) - T_{2,1}(n-1) = \mu(L_{2,1}(1)) - \sum_{m=1}^{n} \mu(L_{2,1}(1) \cap f^{m-1}(L_{1,2}(1))), \tag{5.7}$$

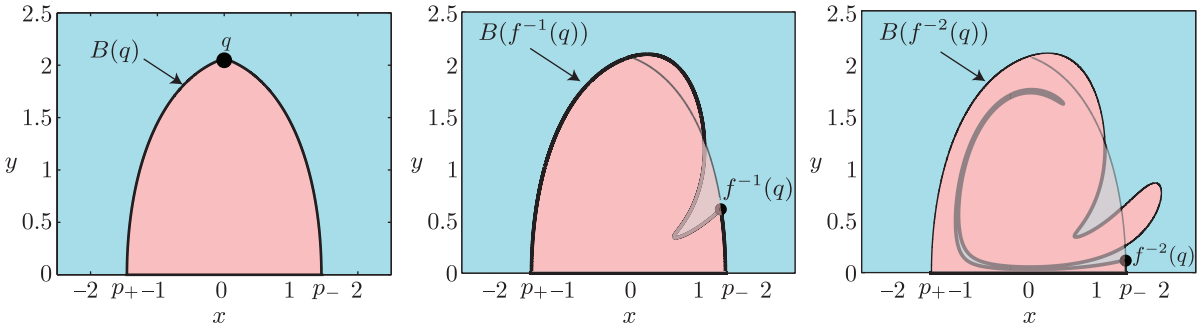


Fig. 12. The boundary with q as a bip and its backward evolution with the preimages of q as bips. The region R_1 is the transparent layer on the preimages.

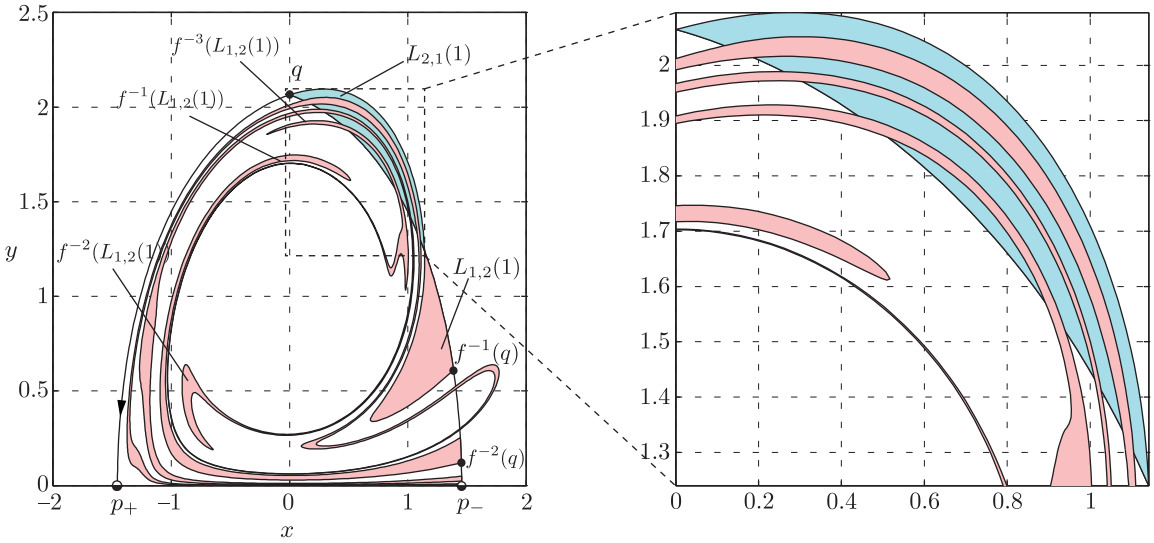


Fig. 13. The *turnstile* lobe in cyan (entraining lobe labeled $L_{1,2}(1)$), magenta (detraining lobe labeled $L_{2,1}(1)$) and its 3 preimages for the OVP flow. The zoom-in view shows the intersection regions that are used in quantifying the transport. We have used $\epsilon = 0.1, \gamma = 0.5$.

where the quantity under the sum denotes the intersection area of the lobe $L_{2,1}(1)$ and preimages of $L_{1,2}(1)$, and is shown in Fig. 13. While the lobe method is a reduced order calculation for transport, the boundary method is useful when multilobe, self-intersecting turnstiles generate near-tangent intersections. In terms of the boundary, the quantity $T_{1,2}(n)$ is given by

$$T_{1,2}(n) = [B(q) \setminus (B(q) \cap f^{-n}(B(q)))] \tag{5.8}$$

and the quantity $a_{1,2}(n)$ is

$$a_{1,2}(n) = T_{1,2}(n) - T_{1,2}(n-1). \tag{5.9}$$

For validating the output from *Lober*, invariant manifolds of the hyperbolic fixed points p_+, p_- are provided as input and the outputs are shown in Fig. 11. It clearly identifies the lobes, pips, and the boundary parameterized by the pip at $q = (0.0, 2.065)$, and indexed as $\lfloor \#\text{ pips}/2 \rfloor$ for a given number of pips. Using the boundary method, we obtain the turnstile lobe area, $\mu(L)$, for different circulation strength of the vortices, γ, values, and show the numerical result in Fig. 14a. However, for validation purposes we have performed the intersection of lobe area computations to obtain a_{ij} in Eq. (5.7) and show the results for the first 7 iterations of the map f in Fig. 14b.

5.2. Escape from a Potential Well

In this section, we will apply the numerical method to computation of the rate of escape from a potential well in the context of capsize of a ship studied as a nonlinear coupling of roll and pitch

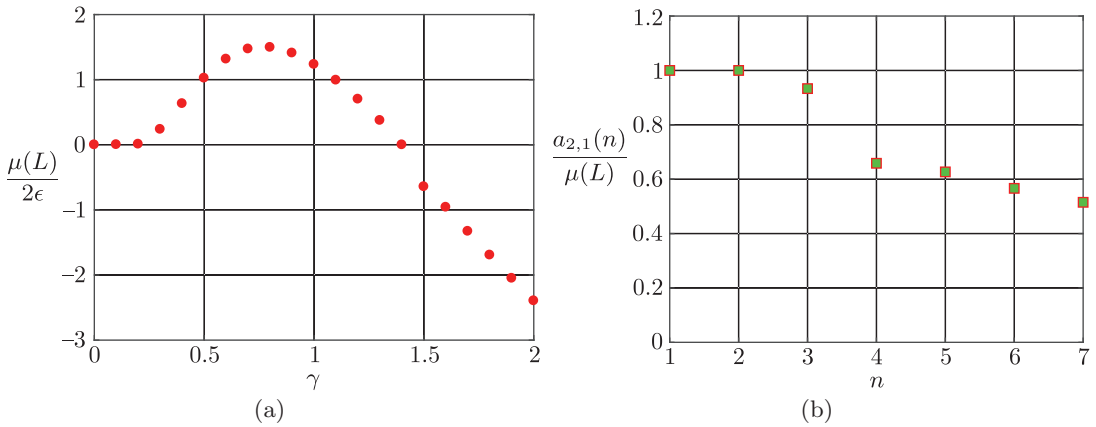


Fig. 14. (a) The area of lobes for different γ values with $\epsilon = 0.1$. It is to be noted that for $\gamma \geqslant 1.5$, the lobe area is shown as negative to imply the change in the geometry of the manifold intersection. This agrees well with Fig. 9 in [2], which compares the brute force lobe area calculations with the Melnikov function. (b) The normalized fluid volume that is given by the quantity $a_{2,1}(n)$ in Eq. (5.7) for 7 iterates of the map f, and for perturbation period $\gamma = 0.5$ and $\epsilon = 0.1$.

degrees of freedom. The full description and analysis of this system can be found in [34], which applies tube dynamics [8], a geometric method of phase space transport, to study the capsize of a ship in a 2-degree-of-freedom (DOF) system. The ship capsize model can be considered as an archetype for studying escaping dynamics in an engineering problem, and hence the rate of escape from the underlying potential well provides an estimate of the probability of capsize. As derived in [34], the rescaled Lagrangian is given by

$$\mathcal{L}(x, y, v_x, v_y) = \frac{1}{2}v_x^2 + \frac{1}{R^2}v_y^2 - V(x, y), \tag{5.10}$$

where

$$V(x, y) = \frac{1}{2}x^2 + y^2 - x^2 y \tag{5.11}$$

is the underlying potential energy, and is shown in Fig. 15. Using the Lagrangian of Eq. (5.10), the

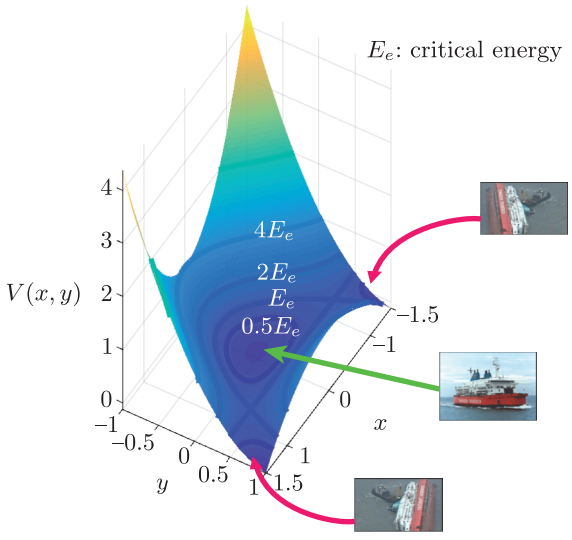


Fig. 15. The effective potential energy with an upright ship in the region that corresponds to bounded motion inside the well and a capsized ship in the region that corresponds to unbounded motion. The total energy of the system can be considered as fixing a height of this potential well and is shown here as contour lines of $0.5E_e, E_e, 2E_e, 4E_e$ on the configuration space (x, y) for different values above and below the critical energy E_e.

equation of motion is given by

$$\begin{aligned} \ddot{x} &= -x + 2xy, \\ \ddot{y} &= -R^2 y + \frac{1}{2} R^2 x^2. \end{aligned} \tag{5.12}$$

In this rescaled form, the only parameter is $R = \omega_\theta/\omega_\phi$, which is the ratio of natural pitch and roll frequencies. The rescaling of the coordinates and time has made the nonlinear coupling term of the potential into unity and the original coordinates can always be recovered using the transformation

$$x = \frac{\phi}{\phi_e}, \qquad y = \frac{\theta}{2\theta_e}, \qquad \bar{t} = \omega_\phi t. \tag{5.13}$$

We note that Eqs. (5.12) are identical to those derived in [36], where they were called *symmetric internal resonance* equations. The potential energy (5.11) is also similar to the Barbaris potential studied by the chemistry community in [37–39].

The equation of motion of a ship in the absence of nonconservative, time-dependent wave forcing is thus given by the first-order ordinary differential equations as

$$\begin{aligned} \dot{x} &= v_x, \\ \dot{y} &= v_y, \\ \dot{v}_x &= -x + 2xy, \\ \dot{v}_y &= -R^2 y + \frac{1}{2} R^2 x^2, \end{aligned} \tag{5.14}$$

which conserves the energy

$$E(x, y, v_x, v_y) = \frac{1}{2} v_x^2 + \frac{1}{R^2} v_y^2 + \frac{1}{2} x^2 + y^2 - x^2 y. \tag{5.15}$$

Let $\mathcal{M}(e)$ be the energy surface given by setting the total energy (5.15) equal to a constant, that is,

$$\mathcal{M}(e) = \{(x, y, v_x, v_y) | E(x, y, v_x, v_y) = e\}, \tag{5.16}$$

where e denotes the constant value of energy. The projection of the energy surface onto the configuration space (x, y), which is historically known as *Hill's region*, defines the region that is energetically accessible and is shown in gray in Fig. 16. For a fixed energy e, the surface $\mathcal{M}(e)$ is a three-dimensional surface embedded in the four-dimensional phase space, $\mathbb{R}^4$. Furthermore, we can consider a cross-section of this three-dimensional surface using a two-dimensional Poincaré surface of section (SOS) given by

$$U_1 = \{(y, v_y) | x = 0, v_x(y, v_y; e) > 0\}, \qquad \text{motion to the right.} \tag{5.17}$$

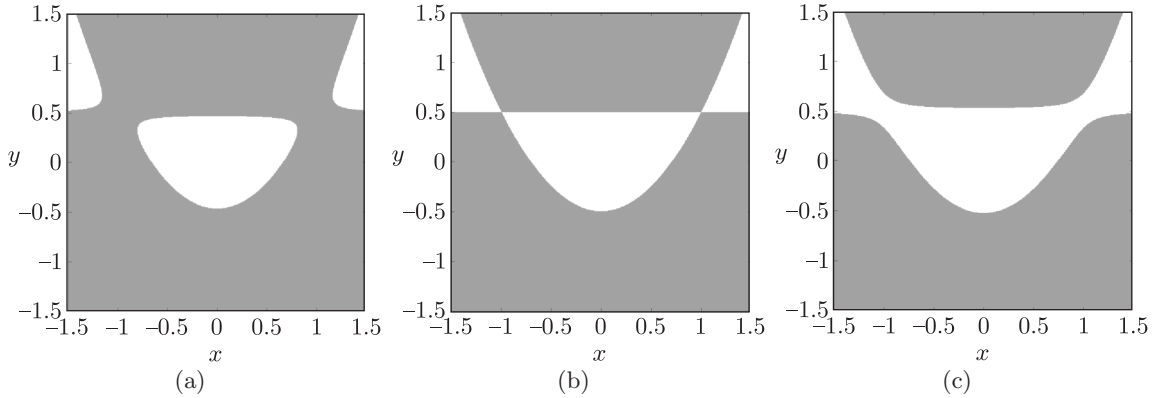
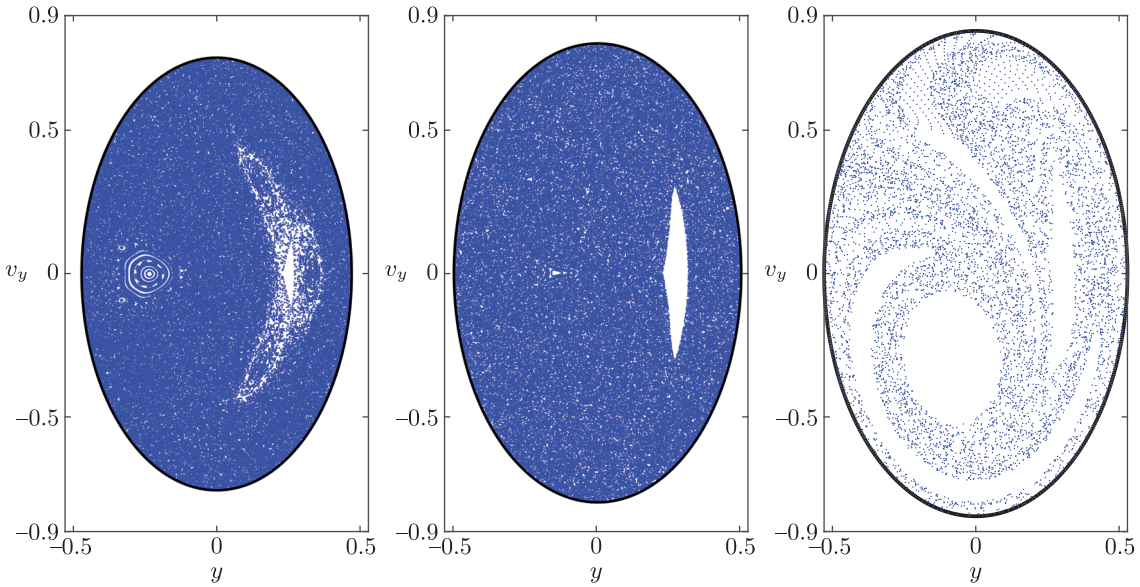


Fig. 16. Hill's region for (a) $e < E_e$, (b) $e = E_e$, and (c) $e > E_e$, where E_e denotes the critical energy. The white region is the energetically accessible region bounded by the zero velocity curve, while the gray region is the energetically forbidden realm where kinetic energy is negative and motion is impossible.

The SOS can be used to define a two-dimensional ($\mathbb{R}^2 \to \mathbb{R}^2$) return map

$$g : U^1 \to U^1 \tag{5.18}$$

for a constant energy, e, and $v_x > 0$ is used to enforce a directional crossing of the surface. This is shown for successively increasing energy in Fig. 17 for the SOS (5.17).



(a) $e = 0.22$ (b) $e = 0.25$ (c) $e = 0.28$

Fig. 17. Poincaré surface of section (SOS) of the energy surface showing orbits of the return map for different energy values in (a) $e = 0.22$, (b) $e = 0.25$, and (c) $e = 0.28$. In the absence of damping and wave forcing, the system conserves energy. When energy is below the critical value, E_e, trajectories cannot escape the potential well, and must intersect the SOS given by Eq. (5.17). When the energy is above the critical value, trajectories escape via the right or left saddles and do not intersect the surface (5.17) infinitely often. $R = 1.6$ is used in all the cases.

The energy of the equilibrium points is called the critical energy (or escaping energy), which is given by $E_e = 0.25$. As the trajectories approach this energy from below, capsize becomes inevitable and this can be interpreted in terms of the potential energy well. Since the potential energy (5.11) is independent of any system parameter, the discussion based on this potential well will be more general.

The dynamical system given by Eq. (5.14) has saddle equilibrium points at $(\pm 1, 0.5, 0, 0)$ and the realm of possible motion in configuration space between the saddles is called the *non-capsize realm*, and all possible states beyond as *capsize realm*. When the energy is above the critical energy, E_e, bottlenecks appear around the saddle points (as shown in Fig. 16c), which act as partial barriers of capsize because trajectories escaping the potential well, in the absence of forcing, reside inside the cylindrical manifolds of geometry $\mathbb{R}^1 \times \mathbb{S}^1$. Traditionally, the escape rate of trajectories from the potential well, that is, the non-capsize realm, is computed using *tube dynamics* [8, 9, 33], or *transition state theory* [40] along with the Monte-Carlo method to estimate the area that escapes in a given time interval.

However, we will use the implementation in *Lober* to compute the escape rate from the potential well, and briefly summarize the steps involved in computing the curves that enclose the escaping regions on the SOS (see [34] for details).

a. We select a suitable Poincaré SOS which is intersected by almost all of the trajectories that escape from the potential well. It is discussed in [41] as selecting a periodic orbit dividing the surface that avoids a pathological intersection of manifolds and Poincaré SOS. However, we make an educated guess for the present system, and use the SOS given by Eq. (5.17), which defines a plane to capture the motion of trajectories to the right, and is shown as magenta plane in Fig. 18a.

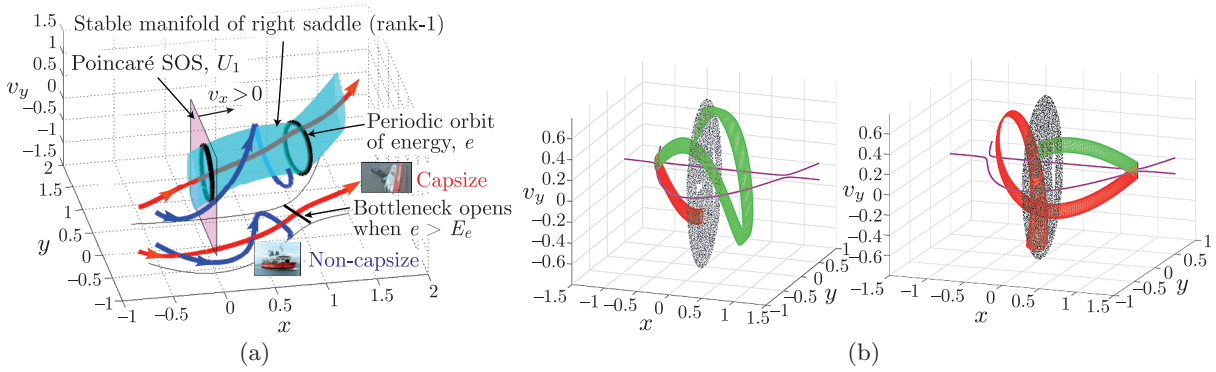


Fig. 18. (a) The Poincaré SOS, U_1 (5.17), as the magenta plane and the stable manifold of right saddle is shown as the cyan surface for the energy $e = 0.28$. The stable manifold of a given saddle up to its first intersection with U_1 is the pathway that leads to imminent capsize via the saddle. The trajectory that leads to escape from the potential well and corresponds to the imminent capsize of a ship, for example, the red trajectory, lies inside the tube. Similarly, a trajectory that does not escape the potential well, for example, the blue trajectory, and corresponds to an upright ship, lies outside the tube. These example trajectories are shown here in the $x - y - v_y$ space for the energy $e = 0.28$, and also as projection in the configuration space $x - y$. (b) Tube manifolds associated with the left and right saddles in the $x - y - v_y$ space for a given energy e. The green and red surfaces denote the stable and unstable tube manifolds, respectively.

The $(y - v_y)$ SOS captures the escape trajectories when the energy is greater than the critical energy as shown in Fig. 17.

b. We obtain the periodic orbit associated with the rank-1 saddle using differential correction. The periodic orbit (p.o.) corresponding to an energy level $e + \Delta e$ is the bounding p.o., and projects as a line on the configuration space, (x, y), as shown in Fig. 18a.

c. We compute the invariant manifolds associated with the p.o. of energy, $e + \Delta e$, using globalization and numerical continuation for the left and right saddle equilibrium points, and as shown in Fig. 18b. The invariant manifolds being codimension-1 in the 3-dimensional energy surface and having geometry $\mathbb{R}^1 \times \mathbb{S}^1$ are cylindrical and hence are referred to as *tube manifolds* [8].

d. We obtain the first intersection of the unstable and stable tube manifolds with the Poincaré SOS, Eq. (5.17), which are shown as red and green curves in Fig. 19a. By the geometry of the stable tube manifolds, the first intersection is the boundary of the trajectories that lead to imminent escape, that is, they do not return to the surface and lead to escape via the right or left bottlenecks. Similarly, the unstable tube manifolds lead to imminent entry, that is, the trajectories entering the potential well via the respective bottlenecks and intersecting the SOS.

Using the boundary of these intersections and the preimages under the map, Eq. (5.18), we can partition the SOS into regions with different exit time. This time is measured as the number of intersections with the SOS a trajectory undergoes before exiting or after entering the potential well. Hence, we can calculate what fraction of the set of trajectories will lead to escape by computing the area of intersection of preimages of the first intersection with itself. We denote the first intersection of the stable tube manifolds with the SOS by Γ_r^s and Γ_l^s, where the subscripts denote exit via the right and left bottlenecks, respectively. The geometry of the manifolds tells us that the trajectories that lead to imminent capsize in n iterates of the return map, Eq. (5.18), must be inside either Γ_r^s or Γ_l^s in the n^{th} preimage. Thus, the trajectories that lead to escape in n after starting inside the unstable tube manifold's first intersection will be inside the n^{th} preimage of the first intersection, say $g^{-n}(\Gamma_r^u)$, and Γ_r^s. The results of this computation are shown in Table 5 as the fraction of trajectories that start inside Γ_r^u or Γ_l^u.

The rate of escape computation based on the geometry of manifolds is advantageous, and the numerical methods presented herein make some progress in this direction. This becomes much more desirable when considering the diverse applications of this approach to chemical physics [41], celestial mechanics [40], and ship capsize [34].

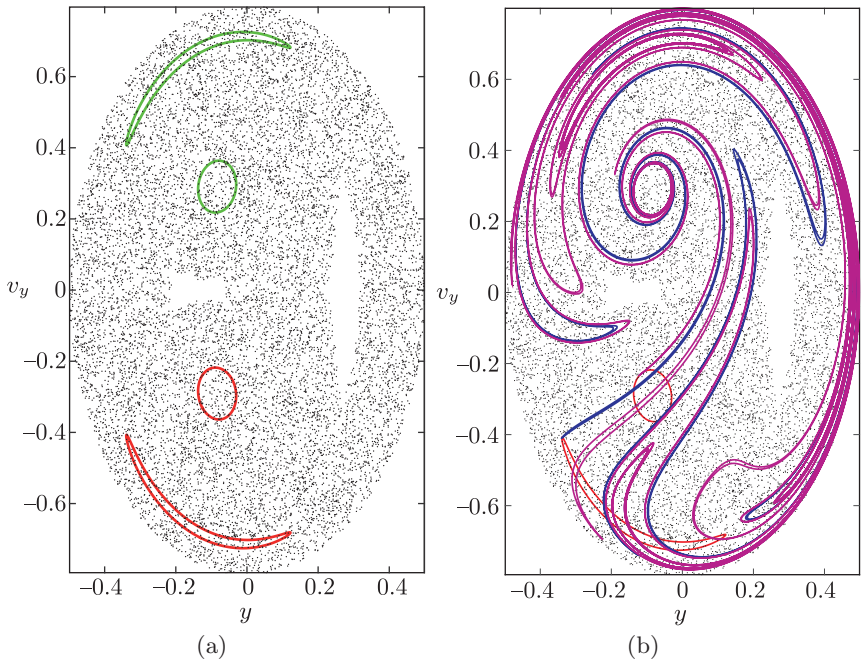


Fig. 19. (a) The tube manifolds shown as closed curves on the Poincaré SOS, U_1 (5.17). The green and red curves denote stable and unstable manifolds, respectively, of right and left saddle equilibrium points. The black dots correspond to the trajectories at the critical energy that intersect the SOS. (b) The preimages of the unstable manifolds shown as magenta curves. We have used $\Delta e = 0.00307$ above the critical energy for the escape rate computations.

Table 5. Percentage of trajectories escaping via left/right stable tube that entered via left and right unstable tubes.

Exit via left after intersection #	Entrance via left	Entrance via right
1	0 %	0 %
2	0 %	11.5 %
3	2.93 %	0.016 %
4	1.87 %	1.441 %
Exit via right after intersection #	Entrance via left	Entrance via right
1	0 %	0 %
2	0 %	0 %
3	11.2 %	2.90 %
4	0.0246 %	0.278 %

6. CONCLUSION

Lobe dynamics describes global transport in terms of lobes, parcels of phase space bounded by stable and unstable invariant manifolds associated to hyperbolic fixed points of the system. Escape from a potential well describes the phase space structures that lead to critical events in dynamical systems by crossing certain barriers. Both of these frameworks — in the circumstances where the dynamics can be reduced to two-dimensional maps — require computation of curves, intersection points, and the area bounded by the curves to quantify phase space transport.

In this paper, we have developed a method for computing phase space transport that uses the area bounded between curves. These curves, however, may be obtained from a higher-dimensional

and/or finite-time dynamical system. We presented this method using a theory for the classification of intersection points to compute the area bounded between the segments of the curves. This involves the partition of the intersection points into equivalence classes to apply a discrete form of Green's theorem. An alternate method for curves with nontransverse intersections — related to identifying lobes by iterating a boundary parameterized by the bip — was also presented along with a method to insert points in the curve for increasing the density of points near intersections. The code for increasing the resolution of a curve is implemented in *Curve_densifier* using an interpolation and insertion method developed for contour surgery in [42], and is made available as open-source repository in Github (`https://github.com/Shibabrat/curve_densifier`). In this case, the notion of primary intersection point and secondary intersection point is not suitable, and hence we propose a generalization of the notion of lobes for two intersecting curves. We have presented a numerical implementation of the approaches for computation of intersection points between two closed curves, lobes defined by these intersections, and the area of these lobes. The software package *Lober* based on the theory and numerical methods presented in Sections 2, 3, and 4 is made available as an open-source repository in Github (`https://github.com/shibabrat/lober`). The framework described was applied to chaotic transport in fluid flow and escape from a potential well in the context of ship capsize, which were reducible to the study of transport in two-dimensional maps. Furthermore, the methods presented in this article can also be used in conjunction with the homotopic lobe dynamics technique, which has been extended to higher-dimensional systems in [43].

APPENDIX. USAGE AND OUTPUTS

This section provides low level details for using the package *Lober* and processing the output of intersection points, lobes, and lobe areas. There are two primary numerical methods that are implemented in the package and are derived in Sections 3 and 4. These two methods deal with transverse and nontransverse intersections and compute the intersection points and the area enclosed by the piecewise linear segments. The transverse intersection module is activated when the syntax

```
lober <c1> <c2> <rslt> [ -DENS <nPass> <nDens> ]
```

is entered at the terminal, while the module for nontransverse intersection is activated when *-light* is added to the syntax in the form

```
lober -light <c1> <c2> <rslt> [ -DENS <nPass> <nDens> ],
```

where `<c1>` and `<c2>` are the names of the files containing the curves C_1 and C_2, respectively, and `<rslt>` is the file to be created by *Lober* to save the output. The optional arguments of the built-in *densifier* (`-DENS <nPass> <nDens>`) has been described in Section 4.3.2. The input curves need to be specified in the files in a *Tecplot* ASCII format given by

```
VARIABLES=''x''''y''
ZONE T=''the curve C1''
0.2    0.4
0.23   0.45
0.35   0.35
...
```

The output file (`<rslt>`)contains one line with 4 numbers: the area of the lobes inside, the area of the lobes outside and the relative error on these two values. This is useful to get an estimate of the error involved in computing the area, and provides a first-order check of the output. In addition, *Lober* generates 6 output files in *Tecplot* ASCII format with one header line and points arranged in $N \times 2$ array of N points in $\mathbb{R}^2$. The intersection points are stored in the files `c10.dat`and `c20.dat`. The set of points which are on the boundary of $C_1 \cap C_2$ and $C_2 \cap C_1$ are stored in `c11.dat`and `c22.dat`,respectively, and those on the boundary of $C_1 \setminus C_2$, $C_2 \setminus C_1$, are stored in `c12.dat and \verb c21.dat, respectively.`

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under awards 1150456, 1520825, and 1537349. The authors would like to thank the anonymous reviewers for their constructive and fruitful suggestions.

REFERENCES

1. Ross, Sh. D. and Tallapragada, P., Detecting and Exploiting Chaotic Transport in Mechanical Systems, in *Applications of Chaos and Nonlinear Dynamics in Science and Engineering: Vol. 2,* S. Banerjee, L. Rondoni, M. Mitra (Eds.), Berlin: Springer, 2012, pp. 155–183.
2. Rom-Kedar, V., Leonard, A., and Wiggins, S., An Analytical Study of Transport, Mixing and Chaos in an Unsteady Vortical Flow, *J. Fluid Mech.,* 1990, vol. 214, pp. 347–394.
3. Wiggins, S., On the Geometry of Transport in Phase Space: 1. Transport in k-Degree-of-Freedom Hamiltonian Systems, $2 \leqslant k < \infty$, *Phys. D,* 1990, vol. 44, no. 3, pp. 471–501.
4. Gillilan, R. E. and Ezra, G. S., Transport and Turnstiles in Multidimensional Hamiltonian Mappings for Unimolecular Fragmentation: Application to van der Waals Predissociation, *J. Chem. Phys.,* 1991, vol. 94, no. 4, pp. 2648–2668.
5. Beigie, D., Multiple Separatrix Crossing in Multi-Degree-of-Freedom Hamiltonian Flows, *J. Nonlinear Sci.,* 1995, vol. 5, no. 1, pp. 57–103.
6. Beigie, D., Codimension-One Partitioning and Phase Space Transport in Multi-Degree-of-Freedom Hamiltonian Systems with Non-Toroidal Invariant Manifold Intersections. Decidability and Predictability in the Theory of Dynamical Systems, *Chaos Solitons Fractals,* 1995, vol. 5, no. 2, pp. 177–211.
7. Lekien, F., Shadden, Sh. C., and Marsden, J. E., Lagrangian Coherent Structures in n-Dimensional Systems, *J. Math. Phys.,* 2007, vol. 48, no. 6, 065404, 19 pp.
8. Koon, W. S., Lo, M. W., Marsden, J. E., and Ross, Sh. D., Heteroclinic Connections between Periodic Orbits and Resonance Transitions in Celestial Mechanics, *Chaos,* 2000, vol. 10, no. 2, pp. 427–469.
9. Dellnitz, M., Junge, O., Koon, W. S., Lekien, F., Lo, M. W., Marsden, J. E., Padberg, K., Preis, R., Ross, Sh. D., and Thiere, B., Transport in Dynamical Astronomy and Multibody Problems, *Internat. J. Bifur. Chaos Appl. Sci. Engrg.,* 2005, vol. 15, no. 3, pp. 699–727.
10. Zotos, E. E., Escape Dynamics and Fractal Basins Boundaries in the Three-Dimensional Earth – Moon System, *Astrophys. Space Sci.,* 2016, vol. 361, no. 3, Art. 94, 23 pp.
11. Zotos, E. E., Fractal Basin Boundaries and Escape Dynamics in a Multiwell Potential, *Nonlinear Dynam.,* 2016, vol. 85, no. 3, pp. 1613–1633.
12. McRobie, F. A. and Thompson, J. M. T., Lobe Dynamics and the Escape from a Potential Well, *Proc. Roy. Soc. London Ser. A,* 1991, vol. 435, no. 1895, pp. 659–672.
13. Martens, C. C., Davis, M. J., and Ezra, G. S., Local Frequency Analysis of Chaotic Motion in Multidimensional Systems: Energy Transport and Bottlenecks in Planar OCS, *Chem. Phys. Lett.,* 1987, vol. 142, no. 6, pp. 519–528.
14. Toda, M., Crisis in Chaotic Scattering of a Highly Excited van der Waals Complex, *Phys. Rev. Lett.,* 1995, vol. 74, no. 14, pp. 2670–2673.
15. Aref, H., Stirring by Chaotic Advection, *J. Fluid Mech.,* 1984, vol. 143, pp. 1–21.
16. Aref, H. and Balachandar, S., Chaotic Advection in a Stokes Flow, *Phys. Fluids,* 1986, vol. 29, no. 11, pp. 3515–3521.
17. Duan, J. and Wiggins, S., Lagrangian Transport and Chaos in the Near Wake of the Flow around an Obstacle: A Numerical Implementation of Lobe Dynamics, *Nonlinear Proc. Geophys.,* 1999, vol. 4, no. 3, pp. 125–136.
18. Samelson, R. M. and Wiggins, S., *Lagrangian Transport in Geophysical Jets and Waves: The Dynamical Systems Approach,* Interdiscip. Appl. Math., vol. 31, New York: Springer, 2006.
19. Rom-Kedar, V. and Wiggins, S., Transport in Two-Dimensional Maps, *Arch. Rational Mech. Anal.,* 1990, vol. 109, no. 3, pp. 239–298.
20. Mackay, R. S., Meiss, J. D., and Percival, I. C., Transport in Hamiltonian Systems, *Phys. D,* 1984, vol. 13, nos. 1–2, pp. 55–81.
21. Wiggins, S., *Chaotic Transport in Dynamical Systems,* Interdiscip. Appl. Math., vol. 2, New York: Springer, 1992.
22. Rom-Kedar, V., Transport in a Class of n-D.o.F. Systems, in *Hamiltonian Systems with Three or More Degrees of Freedom: Proc. of the NATO Advanced Study Institute (S'Agaró, 1995),* C. Simó (Ed.), NATO Adv. Sci. Inst. Ser. C Math. Phys. Sci., vol. 533, Dordrecht: Kluwer, 1999, pp. 538–543.
23. Sandstede, B., Balasuriya, S., Jones, Ch. K. R. T., and Miller, P., Melnikov Theory for Finite-Time Vector Fields, *Nonlinearity,* 2000, vol. 13, no. 4, pp. 1357–1377.
24. Balasuriya, S., Approach for Maximizing Chaotic Mixing in Microfluidic Devices, *Phys. Fluids,* 2005, vol. 17, no. 11, 118103, 4 pp.
25. Balasuriya, S., Cross-Separatrix Flux in Time-Aperiodic and Time-Impulsive Flows, *Nonlinearity,* 2006, vol. 19, no. 12, pp. 2775–2795.

26. Balasuriya, S., Froyland, G., and Santitissadeekorn, N., Absolute Flux Optimising Curves of Flows on a Surface, *J. Math. Anal. Appl.,* 2014, vol. 409, no. 1, pp. 119–139.
27. Mitchell, K. A., Handley, J. P., Tighe, B., Delos, J. B., and Knudson, S. K., Geometry and Topology of Escape: 1. Epistrophes, *Chaos,* 2003, vol. 13, no. 3, pp. 880–891.
28. Mitchell, K. A., Handley, J. P., Delos, J. B., and Knudson, S. K., Geometry and Topology of Escape: 2. Homotopic Lobe Dynamics, *Chaos,* 2003, vol. 13, no. 3, pp. 892–902.
29. Mitchell, K. A. and Delos, J. B., A New Topological Technique for Characterizing Homoclinic Tangles, *Phys. D,* 2006, vol. 221, no. 2, pp. 170–187.
30. Shamos, M. I. and Hoey, D., Geometric Intersection Problems, in *Proc. of the 17th Annual Symposium on Foundations of Computer Science (SFCS'76),* Washington, D.C.: IEEE Computer Society, 1976, pp. 208–215.
31. O'Rourke, J., *Computational Geometry in C,* 2nd ed., Cambridge: Cambridge Univ. Press, 1998.
32. Koon, W. S., Marsden, J. E., Ross, Sh. D., Lo, M., and Scheeres, D. J., Geometric Mechanics and the Dynamics of Asteroid Pairs, *Ann. N. Y. Acad. Sci.,* 2004, vol. 1017, pp. 11–38.
33. Koon, W. S., Lo, M. W., Marsden, J. E., and Ross, Sh. D., *Dynamical Systems, the Three-Body Problem and Space Mission Design,* Marsden Books, 2011.
34. Naik, Sh. and Ross, Sh. D., Geometry of Escaping Dynamics in Nonlinear Ship Motion, *Commun. Nonlinear Sci. Numer. Simul.,* 2017, vol. 47, pp. 48–70.
35. Mancho, A. M., Small, D., Wiggins, S., and Ide, K., Computation of Stable and Unstable Manifolds of Hyperbolic Trajectories in Two-Dimensional, Aperiodically Time-Dependent Vector Fields, *Phys. D,* 2003, vol. 182, nos. 3–4, pp. 188–222.
36. Thompson, J. M. T. and De Souza, J. R., Suppression of Escape by Resonant Modal Interactions: In Shell Vibration and Heave-Roll Capsize, *Proc. R. Soc. A,* 1996, vol. 452, no. 1954, pp. 2527–2550.
37. Sepúlveda, M. A. and Heller, E. J., Semiclassical Calculation and Analysis of Dynamical Systems with Mixed Phase Space, *J. Chem. Phys.,* 1994, vol. 101, no. 9, pp. 8004–8015.
38. Babyuk, D., Wyatt, R. E., and Frederick, J. H., Hydrodynamic Analysis of Dynamical Tunneling, *J. Chem. Phys.,* 2003, vol. 119, no. 13, pp. 6482–6488.
39. Barrio, R., Blesa, F., and Serrano, S., Bifurcations and Safe Regions in Open Hamiltonians, *New J. Phys.,* 2009, vol. 11, no. 5, 053004, 15 pp.
40. Jaffé, C., Ross, Sh. D., Lo, M. W., Marsden, J. E., Farrelly, D., and Uzer, T., Statistical Theory of Asteroid Escape Rates, *Phys. Rev. Lett.,* 2002, vol. 89, no. 1, 011101, 4 pp.
41. Jaffé, C., Farrelly, D., and Uzer, T., Transition State in Atomic Physics, *Phys. Rev. A,* 1999, vol. 60, no. 5, pp. 3833–3850.
42. Dritschel, D. G., Contour Surgery: A Topological Reconnection Scheme for Extended Integrations Using Contour Dynamics, *J. Comput. Phys.,* 1988, vol. 77, no. 1, pp. 240–266.
43. Maelfeyt, B., Smith, S. A., and Mitchell, K. A., Using Invariant Manifolds to Construct Symbolic Dynamics for Three-Dimensional Volume-Preserving Maps, *SIAM J. Appl. Dyn. Syst.,* 2017, vol. 16, no. 1, pp. 729–769.