

Controllability and motion algorithms for underactuated Lagrangian systems on Lie groups

Francesco Bullo* Naomi E. Leonard† Andrew D. Lewis‡

29/11/1997

Last updated: 10/04/1998

Abstract

In this paper, we provide controllability tests and motion control algorithms for underactuated mechanical control systems on Lie groups with Lagrangian equal to kinetic energy. Examples include satellite and underwater vehicle control systems with the number of control inputs less than the dimension of the configuration space. Local controllability properties of these systems are characterised, and two algebraic tests are derived in terms of the symmetric product and the Lie bracket of the input vector fields. Perturbation theory is applied to compute approximate solutions for the system under small-amplitude forcing; in-phase signals play a crucial role in achieving motion along symmetric product directions. Motion control algorithms are then designed to solve problems of point-to-point reconfiguration, static interpolation and exponential stabilisation. We illustrate the theoretical results and the algorithms with applications to models of planar rigid bodies, satellites and underwater vehicles.

Keywords. nonlinear control, mechanical systems, nonlinear controllability, underactuated systems

AMS Subject Classifications. 70E15, 70Q05, 93B05, 93B29

1. Introduction

Underactuated mechanical control systems provide a challenging research area of increasing interest in both application and theory. In this paper, we examine an important class of underactuated mechanical control systems and address problems in both nonlinear controllability and nonlinear control design. It is precisely because we specialise to a rich class of mechanical systems that we can solve relevant controllability and control design problems; we derive controllability tests and motion control algorithms by making use of the mechanical system structure.

Underwater vehicles, satellites, surface vessels, airships and hovercraft are all examples of simple mechanical control systems on Lie groups. For these systems, relevant Lagrangian

models are available and lift/drag type effects are sometimes negligible. Key features are the following: (1) the configuration space is a Lie group, as, for example, the group of rotations $SO(3)$ in the case of a satellite, (2) the Lagrangian is equal to the kinetic energy, and (3) external forces are fixed with respect to the body. From a theoretical perspective, these systems when underactuated offer a control challenge as they have non-zero drift (e.g., their dynamics include Coriolis effects), their linearisation at zero velocity is not controllable, they are not stabilisable by continuous state feedback and exponential stabilisation cannot be achieved by smooth time-varying feedback [M'Closkey and Murray 1997]. Further, they are generically not nilpotent, not feedback linearisable, not “configuration flat,” as defined by Rathinam and Murray [1998], and no test is available to establish whether they are differentially flat. In other words, the motion planning and the stabilisation problem for this class of systems cannot be solved with any established method.

We focus on this class of systems with fewer actuators than degrees of freedom, and we study several motion control problems in the small-velocity range. From a practical point of view, we are motivated by vehicles that are underactuated either because of an actuator failure or because of a design choice. In the former case, our results will improve robustness to actuator failure and thus will provide autonomous vehicles with greater reliability. In the latter case, our results may allow for vehicle designs that include fewer actuators than typical leading to lighter, less costly designs. Provided that certain controllability conditions are satisfied, we show how underactuated vehicles can still perform important tasks such as stabilisation (station keeping) and short range reconfigurations (parking, tracking).

Relevant past contributions include work on both the nonlinear controllability problem and the constructive controllability problem (including both motion planning and stabilisation). For our work, the most important references for controllability are the works of Sussmann [1987] on small-time local controllability and of Lewis and Murray [1997a] (see also [Lewis and Murray 1997b]) on configuration controllability for simple mechanical systems. Other contributions include local controllability results for other classes of mechanical systems, see [Kelly and Murray 1995] and [Ostrowski and Burdick 1997], and work on global controllability issues, see [Bonnard 1984], [Crouch 1984] and [Manikonda and Krishnaprasad 1997]. Regarding the constructive controllability problem, we employ the same approach as Leonard and Krishnaprasad [1995] and Leonard [1995], where motion algorithms for a class of kinematic systems on Lie groups were designed with small-amplitude periodic inputs. In later work [Rui, Kolmanovsky, McNally, and McClamroch 1996] similar techniques were applied to a different class of mechanical system. Other contributions on oscillatory controls and Lagrangian systems include [Baillieul 1993, Gurvits 1992, Sussmann and Liu 1991]. A somewhat different approach, based on homogeneous time-varying strategies, was employed by Morin and Samson [1997] and Pettersen and Egeland [1997] to design exponentially stabilising control laws for underactuated satellites and surface vessels.

To derive controllability tests for our class of systems, we apply the controllability analysis described by Sussmann [1987] and Lewis and Murray [1997a] to simple mechanical control systems on Lie groups (see also [Bullo and Lewis 1996]). Key features of the analysis are a focus on the evolution of the system's configuration when the initial velocity is zero and the result that computations are performed on the Lie algebra of the Lie group. The local controllability properties are characterised by the algebraic operations of symmetric product and Lie bracket. The symmetric product, which is defined more formally in Section 2, depends upon the metric that defines the kinetic energy and, as we shall see,

*Research Assistant Professor, COORDINATED SCIENCE LABORATORY, UNIVERSITY OF ILLINOIS, URBANA-CHAMPAIGN, URBANA-CHAMPAIGN, IL 61801, U.S.A.

Email: bullo@uiuc.edu, URL: <http://motion.csl.uiuc.edu/~bullo/>

†Associate Professor, DEPARTMENT OF MECHANICAL AND AEROSPACE ENGINEERING, PRINCETON UNIVERSITY, PRINCETON, NJ 08544, U.S.A.

Email: naomi@princeton.edu, URL: <http://www.princeton.edu/~naomi/>

‡Assistant Professor, DEPARTMENT OF MATHEMATICS AND STATISTICS, QUEEN'S UNIVERSITY, KINGSTON, ON K7L 3N6, CANADA

Email: andrew@mast.queensu.ca, URL: <http://www.mast.queensu.ca/~andrew/>

explicitly describes motions that involve both input vector fields and the drift dynamics. Our tests describe which velocities and configurations are reachable, independent of the initial configuration. The notions of good and bad symmetric products play a central role.

Guided by our interpretation of the controllability tests, we apply perturbation theory to investigate the response of the mechanical system to small-amplitude forcing. The initial velocity is also assumed to have small amplitude. The approximations we obtain give further insight into the controllability tests and are instrumental in the subsequent control design. Numerous examples illustrate the meaning of good and bad symmetric products and the effects of in-phase and out-of-phase sinusoidal inputs.

On the basis of a controllability assumption, we design two motion primitives that perform the basic tasks of changing and maintaining velocity. These motion primitives use in-phase inputs and compensate for contributions along bad symmetric product directions (see also [Bullo and Leonard 1997]). The two motion primitives synthesise the controllability analysis and are the building blocks for designing high-level motion procedures. Using discrete-time feedback and multiple calls to the motion primitives, we design motion algorithms to solve the point-to-point reconfiguration problem (i.e., how to steer the system to a desired configuration) and the static interpolation problem (i.e., how to steer the system through a set of desired configurations). We solve point-to-point reconfiguration using a constant velocity algorithm. A second approach to point-to-point reconfiguration consists of interpolating a sequence of segments connecting initial to final configuration. We show the advantage of the latter solution in the case the segments are steady motions of the unforced mechanical system. Next, iterating an approximate stabilisation step we design an algorithm that locally exponentially stabilises the system to a desired configuration. Recall that exponential stabilisation cannot be achieved by smooth time-varying feedback, and indeed our motion primitives are continuous, but not smooth, functions of the state. Accordingly, our approach relies on discrete-time continuous feedback, see [Sordalen and Egeland 1995], and on the iteration of a motion planning step, see [Lafferriere and Sussmann 1991]. Finally, the three algorithms are implemented numerically to verify the approximations and illustrate the control design.

The paper is organised as follows. In Section 2 we present some mathematical preliminaries and provide many examples of mechanical systems on Lie groups. In Section 3 a complete controllability analysis is presented with definitions, tests and examples. In Section 4 formulas for approximate solutions are obtained and used to gain insight into the controllability tests. We design and simulate motion control algorithms in Section 5. Our conclusions are given in Section 6 and Appendix A contains various proofs.

2. Preliminaries and models

In this section we review useful notions from geometric mechanics and Lie group theory and introduce several examples of mechanical control systems on Lie groups. For a more detailed treatment of the mathematical background, we refer the reader to [Murray, Li, and Sastry 1994] and [Marsden and Ratiu 1999]. An overview in a related investigation is presented by Leonard and Krishnaprasad [1995].

2.1. Lie groups tools. A Lie group is a smooth manifold endowed with a smooth binary operation called group multiplication (satisfying associativity and existence of identity and

inverse elements). A Lie algebra is a vector space endowed with a skew symmetric, bilinear operation called the Lie bracket (satisfying the Jacobi identity). An example of a Lie group is the rotation group $SO(3)$ (the set of orthogonal matrices with positive determinant under matrix multiplication). Its associated Lie algebra is the space of skew symmetric matrices $\mathfrak{so}(3)$ (under matrix commutation). Other examples of Lie groups are the sets $SE(n)$ of rigid motions on the n -dimensional Euclidean space \mathbb{R}^n .

Let G denote a matrix Lie group and \mathfrak{g} its Lie algebra. The letters g and h denote elements in G , and Id is the identity. The Greek letters ξ and η denote elements in \mathfrak{g} and $\text{ad}_\xi \eta = [\xi, \eta]$ denotes the Lie bracket operation on \mathfrak{g} . Although most of the results in this paper hold more generally, for ease of presentation we make the assumption that:

(A1) The set G is the Cartesian product of an arbitrary number of copies of $SE(3)$ and its proper subgroups.

G , with the assumption (A1), can be represented as a matrix Lie group with group multiplication defined by matrix multiplication and the Lie bracket on the associated Lie algebra given by matrix commutation, i.e., $[\xi, \eta] = \xi\eta - \eta\xi$. Let \times denote the cross product on \mathbb{R}^3 and define the operator $\hat{\cdot} : \mathbb{R}^3 \rightarrow \mathfrak{so}(3)$ by $\hat{x}y \triangleq x \times y$ for all $x, y \in \mathbb{R}^3$. On $SE(3)$ and $\mathfrak{se}(3)$ we represent a group element $g = (R, p) \in SO(3) \times \mathbb{R}^3$ and a velocity $\xi = (\hat{\Omega}, V) \in \mathfrak{so}(3) \times \mathbb{R}^3$ using homogeneous coordinates

$$g = \begin{bmatrix} R & p \\ 0 & 1 \end{bmatrix}, \quad \text{and} \quad \xi = \begin{bmatrix} \hat{\Omega} & V \\ 0 & 0 \end{bmatrix}.$$

Writing ξ as column vector (Ω, V) , simple algebra shows

$$\text{ad}_\xi = \begin{bmatrix} \hat{\Omega} & 0 \\ \hat{V} & \hat{\Omega} \end{bmatrix}. \quad (2.1)$$

Under (A1), we can define a surjective map and local diffeomorphism called the **exponential map** $\exp : \mathfrak{g} \rightarrow G$; we refer to [Marsden and Ratiu 1999] for the general definition. For example, given $\hat{x} \in \mathfrak{so}(3)$, Rodrigues' formula gives

$$\exp(\hat{x}) = \text{Id} + \sin \|x\| \frac{\hat{x}}{\|x\|} + (1 - \cos \|x\|) \frac{\hat{x}^2}{\|x\|^2},$$

where $\|\cdot\|$ is the standard Euclidean norm. In an open neighbourhood of the origin $\text{Id} \in G$, we define $x = \log(g) \in \mathfrak{g}$ to be the **exponential coordinates** of the group element g and we regard the logarithmic map as a local chart on the manifold G . For example, if $R \in SO(3)$ is such that $\text{tr}(R) \neq -1$, then

$$\log(R) = \frac{\phi}{2 \sin \phi} (R - R^T) \in \mathfrak{so}(3),$$

where ϕ satisfies $2 \cos \phi = \text{tr}(R) - 1$ and $|\phi| < \pi$. In other words, $\log(R)$ is the product of the axis and angle of rotation of R . Corresponding definitions for the group $SE(3)$ are presented by Murray, Li, and Sastry [1994].

On the Lie algebra \mathfrak{g} an inner product is defined by a self-adjoint positive definite tensor $\mathbb{I} : \mathfrak{g} \rightarrow \mathfrak{g}^*$, so that, for example, the norm of ξ is computed as $\|\xi\|_{\mathfrak{g}} = (\mathbb{I}\xi \cdot \xi)^{1/2}$. This induces a metric on the group G using the logarithm map as $d(g, h) = \|\log(gh^{-1})\|_{\mathfrak{g}}$.

2.2. Mechanical control systems on Lie groups. A *simple mechanical control system on a Lie group* is described by the following objects: an n -dimensional Lie group G (defining the configuration space), an inertia tensor $\mathbb{I} : \mathfrak{g} \rightarrow \mathfrak{g}^*$ (defining the kinetic energy) and a set of input co-vectors¹ $\{f_1, \dots, f_m\} \subset \mathfrak{g}^*$ (defining the body-fixed forces). The system is said to be *underactuated* if the number of available input forces m is less than the number of degrees of freedom n .

Let $g \in G$ denote the configuration of the system and $\xi \in \mathfrak{g}$ the body-fixed velocity, so that the kinetic energy is $KE = \frac{1}{2}\xi^T \mathbb{I} \xi$. The kinematic and dynamic equations of motion for the system with Lagrangian equal to the kinetic energy are given by

$$\dot{g} = g \cdot \xi \quad (2.2)$$

$$\mathbb{I} \dot{\xi} = \text{ad}_\xi^* \mathbb{I} \xi + \sum_{i=1}^m f_i u_i(t), \quad (2.3)$$

where ad_ξ^* is the dual operator of ad_ξ , the scalar input functions $\{u_i, i = 1, \dots, m\}$ belong to the space of bounded measurable functions \mathcal{U}^m , and $\sum f_i u_i(t)$ is the resultant force acting on the mechanical system. In geometric mechanics, the dynamic equation (2.3) is called the Euler-Poincaré equation; in robotics, the kinematic equation (2.2) is usually expressed in some choice of coordinate system, as, for example, Euler angles for $SO(3)$.

For any vector η with the property that $\text{ad}_\eta^* \mathbb{I} \eta = 0$, the curve $t \in \mathbb{R} \mapsto (\exp(t\eta), \eta)$ is a solution to the system (2.2)–(2.3) with no inputs. These curves are studied in mechanics [Marsden and Ratiu 1999] under the name of *relative equilibria* and describe motion that corresponds to constant body-fixed velocity for the uncontrolled system.

Next, we introduce the notion of the symmetric product on the Lie algebra \mathfrak{g} . This operation is useful for characterising controllability and approximate solutions for mechanical control systems. It has an elegant generalisation using Riemannian geometry as discussed by Lewis and Murray [1997a], Lewis [1998] and Crouch [1981]. Define the *symmetric product* $\langle \xi : \eta \rangle$ of two vectors ξ, η on \mathfrak{g} , as the vector

$$\langle \xi : \eta \rangle \triangleq -\mathbb{I}^{-1}(\text{ad}_\xi^* \mathbb{I} \eta + \text{ad}_\eta^* \mathbb{I} \xi).$$

For example, on $\mathfrak{so}(3) \approx \mathbb{R}^3$ with the inertia tensor \mathbb{J} and with the equality $\text{ad}_\xi^* = -\hat{\xi}$, we compute $\langle \xi : \eta \rangle = \mathbb{J}^{-1}(\xi \times \mathbb{J} \eta + \eta \times \mathbb{J} \xi)$. For later reference it is also useful to rewrite the dynamic equation (2.3) as

$$\dot{\xi} = -\frac{1}{2} \langle \xi : \xi \rangle + \sum_{i=1}^m b_i u_i(t), \quad (2.4)$$

where we define $b_i \triangleq \mathbb{I}^{-1} f_i$ for simplicity. Additional insight into the meaning of the symmetric product is provided by Lewis and Murray [1997a] and summarised below, where the operation is shown to be equivalent to a Lie bracket between certain vector fields on the full space TG .

2.1 REMARK: (Time scaling) For $\lambda > 0$ define $\tau = \lambda t$. Mechanical control systems verify the following property: if $(g(t), \xi(t))$ is a solution for $t \in [0, 1]$ to the forced system (2.2)–(2.3) with external forcing $u_i(t)$, then $(g(\tau/\lambda), \xi(\tau/\lambda)/\lambda)$ is a solution for $\tau \in [0, \lambda]$ with

¹To simplify notation, we denote the co-vectors f_i with subscripts instead of superscripts.

external forcing $u_i(\tau/\lambda)/\lambda^2$. In other words, if we find an input $u(t)$ that achieves a desired motion in time 1, then $u(t/\lambda)/\lambda^2$ achieves the same motion in time λ . This time/magnitude scaling property should be taken into account when applying the control laws described later in the paper.

2.3. Examples: planar bodies, satellites and underwater vehicles. The following examples of mechanical control systems on Lie groups illustrate the richness of the class of systems of interest. They will be often referred to later, as we study controllability and design control laws. To simplify notation, we let $\{e_1, \dots, e_n\}$ denote the standard basis on \mathbb{R}^n ; for example, for $n = 3$ we set $e_1 = (1, 0, 0)$, $e_2 = (0, 1, 0)$ and $e_3 = (0, 0, 1)$.

Planar rigid body. Let $g = (\theta, x, y) \in SE(2)$ denote the configuration of the planar body and $\xi = (\omega, v_1, v_2)$ its body-fixed velocity. The kinetic energy is $KE = \frac{1}{2}J\omega^2 + \frac{1}{2}m(v_1^2 + v_2^2)$ where J is the moment of inertia and m the mass of the body. On $\mathfrak{se}(2)$ the adjoint operator is computed as

$$\text{ad}_{(\omega, v_1, v_2)} = \begin{bmatrix} 0 & 0 & 0 \\ v_2 & 0 & -\omega \\ -v_1 & \omega & 0 \end{bmatrix}.$$

The two control inputs consist of forces applied at a distance h from the centre of mass, see Figure 1. After inverting $\mathbb{I} = \text{diag}\{J, m, m\}$, we have $b_1 = \frac{1}{m}e_2$ and $b_2 = \frac{-h}{J}e_1 + \frac{1}{m}e_3$. In coordinates the equations of motion (2.2)–(2.3) read

$$\begin{aligned} \dot{\theta} &= \omega & J\dot{\omega} &= -h u_2(t) \\ \dot{x} &= \cos(\theta)v_1 - \sin(\theta)v_2 & m\dot{v}_1 &= m\omega v_2 + u_1(t) \\ \dot{y} &= \sin(\theta)v_1 + \cos(\theta)v_2 & m\dot{v}_2 &= -m\omega v_1 + u_2(t). \end{aligned}$$

These equations provide a model for planar vehicles, for example, a hovercraft that glides on the surface of a body of water with negligible friction.

Satellite with thrusters. Let $R \in SO(3)$ be the rotation matrix describing the attitude of the satellite and let $\Omega = (\Omega_1, \Omega_2, \Omega_3) \in \mathfrak{so}(3) \approx \mathbb{R}^3$ be the body angular velocity. The kinetic energy is then $KE = \frac{1}{2}\Omega^T \mathbb{J} \Omega$, where $\mathbb{J} = \text{diag}\{J_1, J_2, J_3\}$ is the inertia matrix. The adjoint operator is $\text{ad}_\Omega = \hat{\Omega}$. Assuming we have two thrusters aligned with the first two principal axes, the equations of motion are

$$\begin{aligned} \dot{R} &= R\hat{\Omega} \\ \mathbb{J}\dot{\Omega} &= \mathbb{J}\Omega \times \Omega + e_1 u_1(t) + e_2 u_2(t). \end{aligned} \quad (2.5)$$

Accordingly, $b_1 = \frac{1}{J_1}e_1$ and $b_2 = \frac{1}{J_2}e_2$.

Satellite with rotors. Satellites can alternatively be equipped with internal rotors (momentum wheels). Consider the case in which there are two rotors aligned with two principal axes of the satellite. The configuration of the satellite plus rotor system is described by $R \in SO(3)$ and $(\theta_1, \theta_2) \in \mathbb{R}^2$ (describing the angular position of the wheels). Let

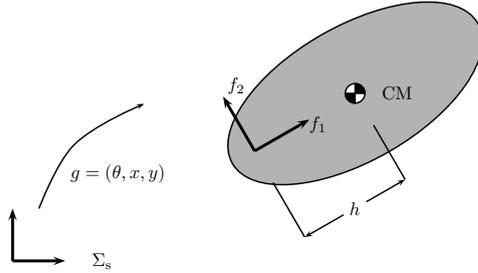


FIGURE 1. Rigid body in $SE(2)$ with two forces applied at a point a distance h from the centre of mass CM. Σ_s denotes an inertial reference system. $g = (\theta, x, y)$ denotes the position of the body. The body reference frame (not depicted) is aligned with the direction of application of f_1 and f_2 .

$\Omega_{\text{rot}} = (\dot{\theta}_1, \dot{\theta}_2, 0)$ denote the angular velocities of the rotors and Ω the angular velocities of the carrier. The kinetic energy is

$$KE = \frac{1}{2}\Omega^T(\mathbb{J}_{\text{lock}} - \mathbb{J}_{\text{rot}})\Omega + \frac{1}{2}(\Omega + \Omega_{\text{rot}})^T\mathbb{J}_{\text{rot}}(\Omega + \Omega_{\text{rot}}),$$

where $\mathbb{J}_{\text{lock}} = \text{diag}\{J_1, J_2, J_3\}$ is the inertia of the satellite-rotors system with the rotors locked, while $\mathbb{J}_{\text{rot}} = \text{diag}(J_{\text{rot}1}, J_{\text{rot}2}, 0)$ is the inertia of the rotors about their spin axes. From the kinetic energy we compute the inertia matrix as

$$\mathbb{J}_{\text{sat-rot}} = \begin{bmatrix} \mathbb{J}_{\text{lock}} & \mathbb{J}_{\text{rot}} \\ \mathbb{J}_{\text{rot}} & \mathbb{J}_{\text{rot}} \end{bmatrix}.$$

Also, the adjoint operator satisfies $\text{ad}_{(\Omega, \Omega_{\text{rot}})}(v, w) = (\Omega \times v, 0)$. The dynamic equations are

$$\begin{bmatrix} \mathbb{J}_{\text{lock}} & \mathbb{J}_{\text{rot}} \\ \mathbb{J}_{\text{rot}} & \mathbb{J}_{\text{rot}} \end{bmatrix} \begin{bmatrix} \dot{\Omega} \\ \dot{\Omega}_{\text{rot}} \end{bmatrix} = \begin{bmatrix} (\mathbb{J}_{\text{lock}}\Omega + \mathbb{J}_{\text{rot}}\Omega_{\text{rot}}) \times \Omega \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{e}_1 u_1(t) + \mathbf{e}_2 u_2(t) \end{bmatrix},$$

and, by inverting the inertia matrix, the input vectors are

$$\begin{aligned} b_1 &= \frac{1}{J_{\text{rot}1} - J_1} \mathbf{e}_1 + \frac{J_1}{J_{\text{rot}1}(J_{\text{rot}1} - J_1)} \mathbf{e}_4 \\ b_2 &= \frac{1}{J_{\text{rot}2} - J_2} \mathbf{e}_2 + \frac{J_2}{J_{\text{rot}2}(J_{\text{rot}2} - J_2)} \mathbf{e}_5. \end{aligned}$$

Underwater vehicle in ideal fluid. The motion of a rigid body in incompressible, irrotational and inviscid fluid is Hamiltonian with an inertia tensor which includes added masses and inertias, see [Lamb 1993] or the original work of Kirchhoff. Let $(R, p) \in SE(3)$ and $(\Omega, V) \in \mathfrak{se}(3)$ denote the configuration and body velocity of the vehicle. The kinematic

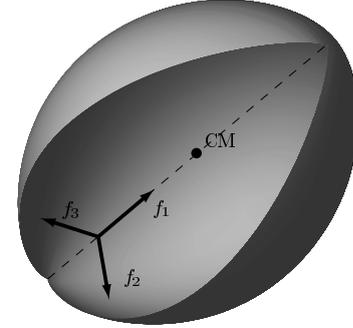


FIGURE 2. Rigid body in $SE(3)$ with three forces applied at a point a distance h from the centre of mass.

equations are

$$\begin{aligned} \dot{R} &= R\hat{\Omega} \\ \dot{p} &= RV. \end{aligned}$$

For a neutrally buoyant ellipsoidal body with uniformly distributed mass, the kinetic energy is $KE = \frac{1}{2}\Omega^T\mathbb{J}\Omega + \frac{1}{2}V^T\mathbb{M}V$, where the mass and inertia matrices of the body-fluid system are $\mathbb{M} = \text{diag}\{m_1, m_2, m_3\}$ and $\mathbb{J} = \text{diag}\{J_1, J_2, J_3\}$. The adjoint operator is given by equation (2.1). The unforced dynamic equations are therefore

$$\begin{aligned} \mathbb{J}\dot{\Omega} &= \mathbb{J}\Omega \times \Omega + \mathbb{M}V \times V \\ \mathbb{M}\dot{V} &= \mathbb{M}V \times \Omega. \end{aligned}$$

Finally, we assume there are three body-fixed forces applied at a point a distance h from the centre of mass, as depicted in Figure 2. The corresponding input vectors are

$$b_1 = \frac{1}{m_1} \mathbf{e}_4, \quad b_2 = -\frac{h}{J_3} \mathbf{e}_3 + \frac{1}{m_2} \mathbf{e}_5, \quad \text{and} \quad b_3 = \frac{h}{J_2} \mathbf{e}_2 + \frac{1}{m_3} \mathbf{e}_6.$$

3. Local controllability properties

This section deals with the nonlinear controllability properties of the systems described above. Our treatment was originally presented by Bullo and Lewis [1996]; it is based on the conditions for small-time local controllability (STLC) by Sussmann [1987] and for configuration controllability by Lewis and Murray [1997a]. While our accessibility analysis is similar to that of Bonnard [1984], Crouch [1984] and Manikonda and Krishnaprasad [1997],

important differences in studying controllability include focusing on small-time, local controllability properties (as opposed to global controllability) and investigating configuration controllability (as opposed to full-state controllability). Configuration controllability concerns the reachable set restricted to the configuration space G and is weaker than full-state controllability. The controllability conditions presented in this section provide us with enough information to allow control design for low-velocity maneuvers (described in Section 5). The main limitation is the assumption that the system is initially at rest. The examples at the end of this section and the perturbation analysis in the next section provide insight into these controllability results.

3.1. Definitions and tests. For $T > 0$, a **solution** of the system (2.2)–(2.3), is a pair (g, u) , where $g : [0, T] \rightarrow G$ is a piecewise smooth curve on G , $u : [0, T] \rightarrow \mathbb{R}^m$ is an admissible input in U^m and $(g(t), u(t))$ are a solution to the equations (2.2)–(2.3). Let $g_0 \in G$, let $V \subset G$ be a neighbourhood of g_0 and let $W \subset G \times \mathfrak{g}$ be a neighbourhood of $(g_0, 0_{\mathfrak{g}})$, where we let $0_{\mathfrak{g}}$ denote the zero vector in \mathfrak{g} . For $T > 0$, set

$$\mathcal{R}_G^V(g_0, T) = \{g_1 \in G \mid \text{there exists a solution } (g, u) \text{ of the system (2.2)–(2.3)} \\ \text{such that } (g, \xi)(0) = (g_0, 0_{\mathfrak{g}}), g(t) \in V \text{ for } t \in [0, T] \text{ and } g(T) = g_1\},$$

and define the set of reachable configurations as

$$\mathcal{R}_G^V(g_0, \leq T) = \bigcup_{0 \leq t \leq T} \mathcal{R}_G^V(g_0, T).$$

Similarly, set

$$\mathcal{R}_{G \times \mathfrak{g}}^W(g_0, T) = \{(g_1, \xi_1) \in G \times \mathfrak{g} \mid \text{there exists a solution } (g, u)(t) \text{ of the system (2.2)–(2.3)} \\ \text{such that } (g, \xi)(0) = (g_0, 0_{\mathfrak{g}}), (g, \xi)(t) \in W \text{ for } t \in [0, T] \text{ and } (g, \xi)(T) = (g_1, \xi_1)\},$$

and define the set of reachable states as

$$\mathcal{R}_{G \times \mathfrak{g}}^W(g_0, \leq T) = \bigcup_{0 \leq t \leq T} \mathcal{R}_{G \times \mathfrak{g}}^W(g_0, T).$$

Accessibility and configuration accessibility. We present definitions, tools and tests that characterise two notions of accessibility.

3.1 DEFINITION: The system (2.2)–(2.3) is **small-time locally accessible at g_0 and zero velocity** if $\mathcal{R}_{G \times \mathfrak{g}}^W(g_0, \leq T)$ contains a non-empty open subset of $G \times \mathfrak{g}$ for all $T > 0$ and for all neighbourhoods W of $(g_0, 0_{\mathfrak{g}})$. If this holds for any $g_0 \in G$, then the system is called **small-time locally accessible at zero velocity** (accessible at zero velocity).

The system (2.2)–(2.3) is **small-time locally configuration accessible at g_0** if $\mathcal{R}_G^V(g_0, \leq T)$ contains a non-empty open subset of G for all $T > 0$ and for all neighbourhoods V of g_0 . If this holds for each $g_0 \in G$, then the system is called **small-time locally configuration accessible** (configuration accessible).

The following concepts are helpful in providing computable tests for the notions of accessibility. Let us denote with $\mathcal{B} = \{b_1, \dots, b_m\}$ the family² of input vectors. Recall that the **involution closure** of \mathcal{B} , denoted by $\overline{\text{Lie}}(\mathcal{B})$, is the set of vectors obtained by taking iterated Lie brackets of the vectors $\{b_1, \dots, b_m\}$. Additionally, we define the **symmetric closure** of \mathcal{B} , denoted by $\overline{\text{Sym}}(\mathcal{B})$, as the set of vectors obtained by taking iterated symmetric products of the vectors $\{b_1, \dots, b_m\}$.

3.2 PROPOSITION: Consider the system (2.2)–(2.3) and let $\mathcal{B} = \{b_1, \dots, b_m\}$ be the family of input vectors.

1. The system is locally accessible at zero velocity if and only if the subspace defined by $\overline{\text{Sym}}(\mathcal{B})$ has full rank.
2. The system is locally configuration accessible if and only if the subspace defined by $\overline{\text{Lie}}(\overline{\text{Sym}}(\mathcal{B}))$ has full rank.

These results are a direct consequence of the accessibility computations of Lewis and Murray [1997a].

Controllability and configuration controllability. Next we present two corresponding notions of controllability: the “classic” small-time local controllability and a weaker version called small-time local configuration controllability.

3.3 DEFINITION: The system (2.2)–(2.3) is **small-time locally controllable at g_0 and at zero velocity** if $\mathcal{R}_{G \times \mathfrak{g}}^W(g_0, \leq T)$ contains a non-empty open subset of $G \times \mathfrak{g}$ for all $T > 0$ and for all neighbourhoods W of $(g_0, 0_{\mathfrak{g}})$, and $(g_0, 0_{\mathfrak{g}})$ belongs to the interior of this subset. If this holds for any $g_0 \in G$, then the system is called **small-time locally controllable at zero velocity** (STLC at zero velocity).

The system (2.2)–(2.3) is **small-time locally configuration controllable at g_0** if $\mathcal{R}_G^V(g_0, \leq T)$ contains a non-empty open subset of G for all $T > 0$ and for all neighbourhoods V of g_0 , and g_0 belongs to the interior of this subset. If this holds for each $g_0 \in G$, then the system is called **small-time locally configuration controllable** (STLCC).

To establish tests that distinguish controllability from accessibility, we need to introduce the notions of good and bad symmetric products and order of a symmetric product. Here we do this in a somewhat simplified way to avoid introducing too much mathematical machinery (i.e., the notion of free Lie algebras of indeterminates).

The **order** of an iterated symmetric product of factors from $\overline{\text{Sym}}(\mathcal{B})$ is the total number of factors. We say that a symmetric product from $\overline{\text{Sym}}(\mathcal{B})$ is **bad** if it contains an even number of each of the vectors in \mathcal{B} . Otherwise, we say that the symmetric product is **good**. For example, the symmetric product $\langle (b_1 : b_2) : b_1 \rangle$ has order three and it is good, the symmetric product $\langle \langle (b_1 : b_2) : b_2 \rangle : b_1 \rangle$ has order four and it is bad. More instances of good and bad symmetric products can be found in the next section.

3.4 PROPOSITION: Consider the system (2.2)–(2.3) and let $\mathcal{B} = \{b_1, \dots, b_m\}$ be the family of input vectors.

²Equivalently, we can think of \mathcal{B} as a linear subspace of \mathfrak{g} or a left-invariant distribution on G .

1. The system is STLC at zero velocity if the subspace defined by $\overline{\text{Sym}}(\mathcal{B})$ has full rank and every bad symmetric product is a linear combination of lower-order good symmetric products.
2. The system is STLCC if the subspace defined by $\overline{\text{Lie}}(\overline{\text{Sym}}(\mathcal{B}))$ has full rank and every bad symmetric product is a linear combination of lower-order good symmetric products.

These results are a direct consequence of the results of Sussmann [1987] and Lewis and Murray [1997a]. We note that symmetric and involutive closures are simple algebraic operations that characterise controllability for the (strongly) nonlinear system (2.2)–(2.3), independent of the base point $g_0 \in G$. These tests have a simple interpretation; symmetric products of input vectors identify which velocities are reachable, whereas Lie brackets of reachable velocities identify which configurations are reachable. The only restriction we impose is the requirement that the system be initially at rest.

Note that the distinction between good and bad symmetric products is analogous to the notion of good and bad Lie brackets in [Sussmann 1987]. Similar characterisations are usually introduced when dealing with controllability properties for systems with drift. In the next section we present some approximate solutions that give some insight into the requirement that “bad products are spanned by lower-order good products.”

Single-input systems ($n > m = 1$) always fail the sufficient condition for both controllability notions; if only one input vector is available, the only possible nontrivial second-order symmetric product is bad. It can further be proven that single-input systems are neither STLC at zero velocity nor STLCC, see [Lewis 1997].

3.2. Applications to the examples. We investigate the controllability properties of the systems introduced in Section 2.3. Our examples are selected to be instructive.

Planar rigid body. Consider the planar rigid body described in Section 2.3 with input vectors $b_1 = \frac{1}{m}e_2$ and $b_2 = \frac{-h}{J}e_1 + \frac{1}{m}e_3$. The relevant symmetric products are computed as follows:

$$\langle b_1 : b_1 \rangle = 0, \quad \langle b_1 : b_2 \rangle = \frac{-h}{Jm}e_3, \quad \langle b_2 : b_2 \rangle = \frac{2h}{Jm}e_2, \quad \text{and} \quad \langle b_2 : \langle b_2 : b_2 \rangle \rangle = \frac{-2h}{J^2m}e_3.$$

We distinguish the following cases which depend on the availability of the two input vectors:

- [PRB1]** $\mathcal{B} = \{b_1\}$: the system is neither accessible at zero velocity nor configuration accessible, as all symmetric products and Lie brackets vanish. An interpretation of this result is that, for all possible inputs, the body is only allowed to translate parallel to the body fixed x -axis.
- [PRB2]** $\mathcal{B} = \{b_2\}$: the system is (small-time locally) accessible at zero velocity since the subspace generated by the vectors $\{b_2, \langle b_2 : b_2 \rangle, \langle b_2 : \langle b_2 : b_2 \rangle \rangle\}$ has full rank. However, the sufficient condition for controllability fails to hold, as $\langle b_2 : b_2 \rangle$ is a bad symmetric product and it is not a multiple of any lower-order symmetric product (b_2 is the only one). Additionally, as mentioned above, the results in [Lewis 1997] show that the system is neither STLC at zero velocity nor STLCC.

[PRB3] $\mathcal{B} = \{b_1, b_2\}$: the system is STLC at zero velocity, since the subspace generated by the vectors $\{b_1, b_2, \langle b_1 : b_2 \rangle\}$ has full rank and the bad symmetric product $\langle b_2 : b_2 \rangle$ is a linear combination of lower-order good symmetric products: $b_2 = -2b_1$.

Satellite with thrusters. Consider the satellite with thrusters described in Section 2.3. Input vectors are $b_1 = \frac{1}{J_1}e_1$ and $b_2 = \frac{1}{J_2}e_2$. The relevant symmetric products and Lie brackets are computed as

$$\langle b_1 : b_1 \rangle = \langle b_2 : b_2 \rangle = 0, \quad \langle b_1 : b_2 \rangle = \frac{J_2 - J_1}{J_1 J_2 J_3} e_3, \quad \text{and} \quad [b_1, b_2] = \frac{1}{J_1 J_2} e_3.$$

The controllability properties are as follows:

[ST] $\mathcal{B} = \{b_1, b_2\}$ and $J_1 \neq J_2$: if the satellite is not axisymmetric³, then the rank of $\{b_1, b_2, \langle b_1 : b_2 \rangle\}$ is full and there are no bad symmetric products. Therefore, the system is STLC at zero velocity.

Satellite with rotors. Consider the satellite with rotors introduced in Section 2.3. For b_1 and b_2 defined in Section 2.3, we compute symmetric products and Lie brackets as

$$\langle b_1 : b_1 \rangle = \langle b_2 : b_2 \rangle = \langle b_1 : b_2 \rangle = 0,$$

and

$$\begin{aligned} [b_1, b_2] &= \frac{1}{(J_{\text{rot1}} - J_1)(J_{\text{rot2}} - J_2)} e_3, \\ [[b_1, b_2], b_1] &= \frac{1}{(J_{\text{rot1}} - J_1)^2 (J_2 - J_{\text{rot2}})} e_2, \\ [[b_1, b_2], b_2] &= \frac{1}{(J_{\text{rot1}} - J_1)(J_{\text{rot2}} - J_2)^2} e_1. \end{aligned}$$

[SR] $\mathcal{B} = \{b_1, b_2\}$: the system is not accessible at zero velocity (every symmetric product vanishes) and hence not STLC, but it is STLCC since the involutive closure has full rank.

This result was partly expected but not trivial. Since the satellite-rotors system is not subject to any external force, its total angular momentum is conserved. Therefore, it is intuitively clear that the system cannot be accessible in both configurations and velocities. However, the less trivial fact is that the system is STLCC. This means that, despite the conservation law, any configuration can be reached, that is, any orientation R together with any rotor angles (θ_1, θ_2) .

Underwater vehicle in ideal fluid. Consider the underwater vehicle introduced in Section 2.3, with the input forces depicted in Figure 2. We compute some good symmetric products as

$$\begin{aligned} \langle b_1 : b_2 \rangle &= \frac{m_2 - m_1}{J_3 m_1 m_2} e_3 - \frac{h}{J_3 m_2} e_5, \\ \langle b_1 : b_3 \rangle &= \frac{m_1 - m_3}{J_2 m_1 m_3} e_2 - \frac{h}{J_2 m_3} e_6, \\ \langle b_2 : b_3 \rangle &= \frac{1}{J_1} \left(\frac{h^2}{J_3} - \frac{h^2}{J_2} - \frac{1}{m_3} + \frac{1}{m_2} \right) e_1 \end{aligned}$$

³If the satellite is axisymmetric, i.e., $J_1 = J_2$, then a simple analysis shows that the system is STLCC.

and some bad ones as

$$\langle b_1 : b_1 \rangle = 0, \quad \langle b_2 : b_2 \rangle = \frac{2h}{I_3 m_1} \mathbf{e}_4, \quad \langle b_3 : b_3 \rangle = \frac{2h}{I_2 m_1} \mathbf{e}_4.$$

[UV] $\mathcal{B} = \{b_1, b_2, b_3\}$: Consider the 6×6 matrix defined by the good symmetric products of order one and two, that is $\{b_1, b_2, b_3, \langle b_1 : b_2 \rangle, \langle b_1 : b_3 \rangle, \langle b_2 : b_3 \rangle\}$. This matrix is generically nonsingular.⁴ Hence, the system is small-time locally accessible at zero velocity. Additionally, since the bad second-order symmetric products are proportional to b_1 , they are spanned by good lower-order symmetric products (b_1 is a good symmetric product of order 1). Therefore, the system is generically STLC at zero velocity.

4. Approximate solutions under small-amplitude forcing

In this section we investigate the behaviour of system (2.2)–(2.3) under small-amplitude forcing. The key analysis tool is the standard perturbation method as described by Khalil [1996]. Assuming a small-amplitude input (say of order ϵ , for $0 < \epsilon \ll 1$), this method provides us with a solution to system (2.2)–(2.3) in the form of a Taylor series in ϵ . Since the computation of only a few terms in the series is tractable, we obtain an approximate expansion. However, this estimate illustrates the role of symmetric products and Lie brackets in determining the solution of the forced system (2.2)–(2.3). Therefore, this estimate provides insight into the controllability tests introduced above and, as we shall see, it is instrumental in designing the motion algorithms of the next section.

4.1. Notation and results. We introduce the following notation. Given a possibly vector-valued function $h(t)$ with $t \in \mathbb{R}_+$, define its first integral function $\bar{h}(t)$ with $t \in \mathbb{R}_+$, as the finite integral from 0 to t

$$\bar{h}(t) \triangleq \int_0^t h(\tau) d\tau.$$

Higher-order integrals, as for example $\bar{\bar{h}}(t)$ are defined recursively. In the following, we consider inputs of the form

$$u_i(t, \epsilon) = \epsilon u_i^1(t) + \epsilon^2 u_i^2(t)$$

where $0 < \epsilon \ll 1$ and where u_i^1, u_i^2 are $O(1)$. Accordingly, we write the resultant forcing $\sum_i b_i u_i(t, \epsilon)$ as the sum of two terms of different order in ϵ

$$\begin{aligned} \sum_{i=1}^m b_i u_i(t, \epsilon) &= \sum_{i=1}^m b_i (\epsilon u_i^1(t) + \epsilon^2 u_i^2(t)) \\ &= \epsilon b^1(t) + \epsilon^2 b^2(t), \end{aligned} \quad (4.1)$$

where we define $b^1(t) = \sum_{i=1}^m b_i u_i^1(t)$ and $b^2(t) = \sum_{i=1}^m b_i u_i^2(t)$. In the following, given any quantity $y(\epsilon)$, we let y^k denote the k th term in the Taylor expansion of $y(\epsilon)$ about $\epsilon = 0$; for example, we will write $\xi(t, \epsilon) = \epsilon \xi^1(t) + \epsilon^2 \xi^2(t) + O(\epsilon^3)$. The following proposition describes the system's behaviour when forced by small (order ϵ and order ϵ^2) amplitude inputs as defined in equation (4.1).

⁴The matrix is singular when $h^2 m_1 m_2 + J_3(m_1 - m_2) = 0$ or when $h^2 m_1 m_3 + J_2(m_1 - m_3) = 0$ or when $h^2(1/J_3 - 1/J_2) = 1/m_3 - 1/m_2$.

4.1 PROPOSITION: (Approximate evolution) For $0 < \epsilon \ll 1$ and for inputs of the form in equation (4.1), let $(g(t), \xi(t))$ be the solutions of system (2.2)–(2.3). Let $x(t)$ be the exponential coordinates of $g(t)$ about the initial condition $g(0) = \text{Id}$. Also, write the initial velocity as $\xi(0) = \epsilon \xi_0^1 + \epsilon^2 \xi_0^2$ where ξ_0^1 and ξ_0^2 are $O(1)$.

Then for $t \in [0, 2\pi]$ it holds that $\xi(t, \epsilon) = \epsilon \xi^1(t) + \epsilon^2 \xi^2(t) + \epsilon^3 \xi^3(t) + O(\epsilon^4)$, with

$$\begin{aligned} \xi^1(t) &= \xi_0^1 + \bar{b}^1(t), \\ \xi^2(t) &= \xi_0^2 - \langle \xi_0^1 : \xi_0^1 \rangle \frac{t}{2} - \langle \xi_0^1 : \bar{b}^1(t) \rangle + \left(b^2 - \frac{1}{2} \langle \bar{b}^1 : \bar{b}^1 \rangle \right) (t), \\ \xi^3(t) &= - \langle \xi_0^1 : \xi_0^2 \rangle t + \langle \xi_0^1 : \langle \xi_0^1 : \xi_0^1 \rangle \rangle \frac{t^2}{4} + \langle \xi_0^1 : \langle \xi_0^1 : \bar{b}^1(t) \rangle \rangle - \langle \xi_0^1 : \left(b^2 - \frac{1}{2} \langle \bar{b}^1 : \bar{b}^1 \rangle \right) (t) \rangle \\ &\quad - \langle \bar{b}^1(t) : \xi_0^2 \rangle + \frac{1}{2} \langle \langle \xi_0^1 : \xi_0^1 \rangle t : \bar{b}^1(t) \rangle + \langle \bar{b}^1 : \langle \xi_0^1 : \bar{b}^1 \rangle \rangle (t) - \langle \bar{b}^1 : \left(b^2 - \frac{1}{2} \langle \bar{b}^1 : \bar{b}^1 \rangle \right) \rangle (t), \end{aligned}$$

and $x(t, \epsilon) = \epsilon x^1(t) + \epsilon^2 x^2(t) + O(\epsilon^3)$, with

$$\begin{aligned} x^1(t) &= \xi_0^1 t + \bar{b}^1(t), \\ x^2(t) &= \xi_0^2 t - \langle \xi_0^1 : \xi_0^1 \rangle \frac{t^2}{4} + \left(b^2 - \frac{1}{2} \langle \bar{b}^1 : \bar{b}^1 \rangle \right) (t) - \langle \xi_0^1 : \bar{b}^1(t) \rangle + \frac{1}{2} \left[\xi_0^1 + \bar{b}^1, \xi_0^1 t + \bar{b}^1 \right] (t). \end{aligned}$$

The proof is based on the standard perturbation method as described by Khalil [1996] and on the approximate solutions for the kinematic system obtained by Fomenko and Chakon [1990]; see Appendix A.1 for a detailed account. Note that both symmetric products and Lie brackets show up in the Taylor expansions and this agrees with the controllability tests presented above. Also, note that the approximations in Proposition 4.1 hold only over a finite period of time and particular care is needed in order to compute approximations valid over a time interval of order $1/\epsilon$.

4.2. Application to examples. We now relate the approximations above to the controllability tests of the previous section. To simplify the expansions above and to investigate the nonlinear second-order effects of the inputs, we let the initial velocity vanish, $\xi(0) = 0_{\mathfrak{g}}$, and the first order input $b^1(t)$ verify $\bar{b}^1(2\pi) = \bar{\bar{b}}^1(2\pi) = 0_{\mathfrak{g}}$. It holds that

$$\xi(2\pi) \approx \epsilon^2 \left(b^2 - \frac{1}{2} \langle \bar{b}^1 : \bar{b}^1 \rangle \right) (2\pi), \quad \text{and} \quad x(2\pi) \approx \epsilon^2 \left(\bar{\bar{b}}^2 - \frac{1}{2} \langle \bar{b}^1 : \bar{b}^1 \rangle + \frac{1}{2} \left[\bar{b}^1, \bar{b}^1 \right] \right) (2\pi), \quad (4.2)$$

where, for the remainder of this section, the symbol \approx denotes an equality up to a third order error in ϵ . Also, if we set $b^2(t) = 0_{\mathfrak{g}}$, it holds that

$$\xi(2\pi) \approx -\frac{1}{2} \epsilon^2 \langle \bar{b}^1 : \bar{b}^1 \rangle (2\pi), \quad \text{and} \quad x(2\pi) \approx -\frac{1}{2} \epsilon^2 \left(\langle \bar{b}^1 : \bar{b}^1 \rangle + \left[\bar{b}^1, \bar{b}^1 \right] \right) (2\pi). \quad (4.3)$$

Up to a higher-order error in ϵ , the final velocity $\xi(2\pi)$ is determined by certain symmetric products and the final configuration variable $x(2\pi)$ is determined by certain symmetric products and Lie brackets. Next, we study in more detail these remaining terms to gain some insight into what terms are “good,” what are “bad” and which ones we can exploit to design motion algorithms.

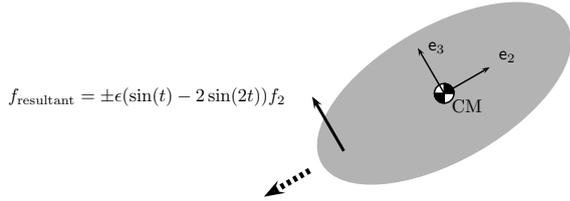


FIGURE 3. Planar rigid body with single forcing [PRB2]. With a resultant external force $\pm\epsilon\cos(t)$ and after a period 2π , the approximate final reconfiguration is $\log(g(0)^{-1}g(2\pi)) = -2\pi\epsilon^2\frac{h}{Jm}\mathbf{e}_2$ and the approximate final velocity is $\xi(2\pi) = -2\pi^2\epsilon^2\frac{h}{Jm}\mathbf{e}_2$.

Single-input systems: relative equilibria and bad symmetric products. Both examples of planar rigid bodies, [PRB1] and [PRB2], are single-input systems. Recall that [PRB1] denotes the system with a single force b_1 with the line of action through the centre of mass, and [PRB2] denotes the system with the single force b_2 applied at a point a distance h from the centre of mass and perpendicular to b_1 , see Figure 1.

Let b_{si} denote the single input vector, e.g., $b_{si} = b_1$ in [PRB1] and $b_{si} = b_2$ in [PRB2]. If the symmetric product $\langle b_{si} : b_{si} \rangle$ vanishes, see the [PRB1] example, the system is neither accessible nor configuration accessibility, and the final state $(x, \xi)(2\pi)$ vanishes. Recall from Section 2, that for any vector η such that $\langle \eta : \eta \rangle \equiv \text{ad}_\eta^* \mathbb{I}\eta = 0$, the curve $t \in \mathbb{R} \mapsto (\exp(t\eta), \eta)$ is a relative equilibria, i.e., a motion corresponding to constant body-fixed velocity. Thus, an actuator b_{si} aligned with a relative equilibria has vanishing bad symmetric product $\langle b_{si} : b_{si} \rangle$.

Also instructive is the case in which the bad symmetric product $\langle b_{si} : b_{si} \rangle$ does not vanish, e.g., the [PRB2] system. Assuming $b^1(t) = b_{si}\phi(t)$ and $\bar{\phi}(2\pi) = \overline{\bar{\phi}}(2\pi) = 0$, equations (4.3) lead to

$$\xi(2\pi) \approx -\frac{1}{2}\epsilon^2 \int_0^{2\pi} \bar{\phi}^2 dt \langle b_{si} : b_{si} \rangle, \quad \text{and} \quad x(2\pi) \approx -\frac{1}{2}\epsilon^2 \int_0^{2\pi} \int_0^s \bar{\phi}^2 ds dt \langle b_{si} : b_{si} \rangle. \quad (4.4)$$

As already mentioned, configuration and velocity change an amount proportional to ϵ^2 along the direction $\langle b_{si} : b_{si} \rangle$. Additionally, notice that it is impossible to change the sign of the motion, which will always be along $-\langle b_{si} : b_{si} \rangle$. For example the [PRB2] system with forcing amplitude $\pm\epsilon\cos(t)$, always moves in the direction $-\mathbf{e}_2$, i.e., to the left (see Figure 3). This phenomenon suggests that the system is not locally controllable, as certain configurations appear to be not reachable. However, equation (4.4) does not prove this claim as it only specifies the final value $x(2\pi)$. The sharper analysis by Lewis [1997] is needed to show that single-input systems are neither STLC at zero velocity nor STLCC.

Multi-input systems with no bad symmetric products. Next we examine systems with (at least) two input forces. We focus on an example with two input vectors b_1 and b_2 that have vanishing bad symmetric products $\langle b_1 : b_1 \rangle = \langle b_2 : b_2 \rangle = 0$ and either non-zero good symmetric product $\langle b_1 : b_2 \rangle \neq 0$ or non-zero Lie bracket $[b_1, b_2] \neq 0$. The satellite with two thrusters [ST] and the satellite with two rotors [SR] are such examples. Plugging $b^1(t) = b_1u_1(t) + b_2u_2(t)$ into equations (4.3), we have

$$\begin{aligned} \xi(2\pi) &\approx -\frac{1}{2}\epsilon^2 \overline{\langle u_1b_1 + u_2b_2 : u_1b_1 + u_2b_2 \rangle}(2\pi) \\ &= -\epsilon^2 \langle b_1 : b_2 \rangle \overline{\bar{u}_1\bar{u}_2}(2\pi) \end{aligned}$$

and

$$\begin{aligned} x(2\pi) &\approx -\frac{1}{2}\epsilon^2 \left(\overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} + \overline{[\bar{b}^1, \bar{b}^1]} \right) (2\pi) \\ &= -\epsilon^2 \langle b_1 : b_2 \rangle \overline{\bar{u}_1\bar{u}_2}(2\pi) - \epsilon^2 [b_1, b_2] \overline{\bar{u}_1\bar{u}_2} - \overline{\bar{u}_1\bar{u}_2}(2\pi). \end{aligned}$$

We interpret the operations performed on the input signals $u_1(t)$ and $u_2(t)$ as follows: $\overline{\bar{u}_1\bar{u}_2}(2\pi)$ is the inner product in the $L_2[0, 2\pi]$ function space between $\bar{u}_1(t)$ and $\bar{u}_2(t)$, whereas $\overline{\bar{u}_1\bar{u}_2} - \overline{\bar{u}_1\bar{u}_2}(2\pi)$ is the area enclosed by the plot of signals $\bar{u}_1(t)$ versus $\bar{u}_2(t)$. We distinguish two cases:

- *Out-of-phase sinusoidal inputs generate motion along Lie brackets:* First, consider the satellite with rotors [SR] example that is STLCC but not STLC at zero velocity. The symmetric product $\langle b_1 : b_2 \rangle$ vanishes, so that we have from equation (4.3)

$$\xi(2\pi) \approx 0 \quad \text{and} \quad x(2\pi) \approx -\epsilon^2 [b_1, b_2] \overline{\bar{u}_1\bar{u}_2} - \overline{\bar{u}_1\bar{u}_2}(2\pi).$$

If we want to steer the configuration $x(2\pi)$ in the direction $[b_1, b_2]$, sinusoidal signals at the same frequency and *out-of-phase* are a simple standard choice. This is one of the basic ideas behind the algorithms presented by Leonard and Krishnaprasad [1995] and other literature on motion planning for driftless control systems.

- *In-phase sinusoidal inputs generate motion along good symmetric products:* Second, consider the satellite with thrusters example [ST] that is STLC at zero velocity since the symmetric product $\langle b_1 : b_2 \rangle \neq 0$. If we pick sinusoidal inputs at the same frequency and *in-phase*, e.g., $u_1(t) = u_2(t) = \cos(t)$, the contribution proportional to the Lie bracket $[b_1, b_2]$ vanishes, since the area included by two identical signal is zero. Further, it holds that

$$\xi(2\pi) \approx -\epsilon^2 \langle b_1 : b_2 \rangle \overline{(\bar{u}_1)^2}(2\pi) \quad \text{and} \quad x(2\pi) \approx -\epsilon^2 \langle b_1 : b_2 \rangle \overline{(\bar{u}_1)^2}(2\pi),$$

and both velocity and configuration variables vary along $-\langle b_1 : b_2 \rangle$.⁵ Motion in the symmetric product direction is generated with sinusoidal inputs at the same frequency and *in-phase*. This is in contrast with the previous case and it is reminiscent of some results on gait selection for locomotion systems with drift, see the 1:1 gait in [Ostrowski and Burdick 1997].

⁵Also, the velocity change is maximal in the sense that on the $L_2[0, 2\pi]$ function space the Cauchy-Schwartz inequality on the inner product of $\bar{u}_1(t)$ and $\bar{u}_2(t)$ holds with equality if $\bar{u}_1(t) = \bar{u}_2(t)$.

Multi-input systems with bad symmetric products. Finally, we examine systems with non-vanishing bad symmetric products. We focus on the planar rigid body with two forces applied at a point distant from the centre of mass [PRB3]. Recall that this system is STLC at zero velocity since the subspace $\{b_1, b_2, \langle b_1 : b_2 \rangle\}$ has full rank and since the good/bad products condition is verified by the equality $\langle b_2 : b_2 \rangle = \frac{2h}{J} b_1$. Setting $b^1 = b_1 u_1(t) + b_2 u_2(t)$ as above, the existence of a non-vanishing bad symmetric product causes

$$-\frac{1}{2} \epsilon^2 \overline{\langle b^1 : b^1 \rangle} (2\pi) = -\epsilon^2 \langle b_1 : b_2 \rangle \overline{u_1} \overline{u_2} (2\pi) - \frac{1}{2} \epsilon^2 \langle b_2 : b_2 \rangle \overline{u_2}^2 (2\pi),$$

where the sign of the second term is independent of $u_2(t)$. However, motion in the $\langle b_2 : b_2 \rangle$ direction can be affected by a second-order input along b_1 . In particular by setting

$$b^2(t) = \frac{h}{2\pi J} \overline{u_2}^2 (2\pi) b_1$$

we obtain from equation (4.2)

$$\xi(2\pi) \approx -\epsilon^2 \langle b_1 : b_2 \rangle \overline{u_1} \overline{u_2} (2\pi),$$

recovering this way the result for the case without bad symmetric products. In other words, the ‘‘bad’’ contribution due to $\langle b_2 : b_2 \rangle$ is ‘‘annihilated’’ by means of the second-order input $b^2(t)$, and this is possible only because the good/bad products condition is verified.

4.3. Inversion algorithm for systems controllable with second-order symmetric products. Motivated by the heuristic analysis in the last two examples, we introduce an additional definition. A system is *STLC at zero velocity with second-order symmetric products* if it satisfies the following property:

(A2) The subspace span $\{b_i, \langle b_j : b_k \rangle, 1 \leq i \leq m, 1 \leq j < k \leq m\}$ has full rank and each bad symmetric product $\langle b_i : b_i \rangle$ is a linear combination of the vectors $\{b_1, \dots, b_m\}$.

The planar rigid body with two forces [PRB3], the satellite with two thrusters [ST] and the underwater vehicle [UV] satisfy this controllability condition. On the basis of this assumption, we design inputs $(b^1(t), b^2(t))$, that allow us to simplify the approximations in Proposition 4.1 and steer the velocity of the system to an arbitrary value.

4.2 LEMMA: (Inversion Algorithm) *Let the assumption (A2) hold and let η be an arbitrary element in \mathfrak{g} . Define the input functions $(b^1(t), b^2(t))$ as follows:*

1. Set $N = m(m-1)/2$ and let P denote the set of ordered pairs $\{(j, k) \mid 1 \leq j < k \leq m\}$. Identify the elements in P with the set of integers $1, \dots, N$, and let $a(j, k)$ be the integer associated with the pair (j, k) . In other words, $a : P \mapsto \{1, \dots, N\}$ is a enumeration of P . For $\alpha = 1, \dots, N$, define the scalar functions

$$\psi_\alpha(t) = \frac{1}{\sqrt{2\pi}} \left(\alpha \sin(\alpha t) - (\alpha + N) \sin((\alpha + N)t) \right).$$

2. Given the assumption (A2), the matrix with columns $b_i, 1 \leq i \leq m$, and $\langle b_j : b_k \rangle, 1 \leq j < k \leq m$, has full rank. By means of its pseudo-inverse, compute $(m + N)$ real numbers z_i and z_{jk} such that

$$\eta = \sum_{1 \leq i \leq m} z_i b_i + \sum_{1 \leq j < k \leq m} z_{jk} \langle b_j : b_k \rangle.$$

3. Finally, set

$$b^1(t) = \sum_{1 \leq j < k \leq m} \sqrt{|z_{jk}|} \left(b_j - \text{sign}(z_{jk}) b_k \right) \psi_{a(j,k)}(t), \quad (4.5)$$

$$b^2(t) \equiv b^2 = \frac{1}{2\pi} \sum_{1 \leq i \leq m} z_i b_i + \frac{1}{4\pi} \sum_{1 \leq j < k \leq m} |z_{jk}| \left(\langle b_j : b_j \rangle + \langle b_k : b_k \rangle \right). \quad (4.6)$$

The input functions $(b^1(t), b^2(t))$ designed in equation (4.5) and (4.6) verify

$$\left(b^2 - \frac{1}{2} \overline{\langle b^1 : b^1 \rangle} \right) (2\pi) = \eta. \quad (4.7)$$

Appendix A.2 contains the lemma’s proof. In what follows, we denote the procedure described in the inversion algorithm with the notation:

$$(b^1(t), b^2(t)) = \text{Inverse}(\eta).$$

A direct manipulation of equation (4.5) and of $b^1(t) = \sum_{i=1}^m b_i u_i^1(t)$ leads to the equivalent statement

$$u_i^1(t) = - \sum_{j=1}^{i-1} \sqrt{|z_{ji}|} \text{sign}(z_{ji}) \psi_{a(j,i)}(t) + \sum_{j=i+1}^m \sqrt{|z_{ij}|} \psi_{a(i,j)}(t).$$

Note that motion along the good symmetric product direction $\langle b_i : b_j \rangle$ (for $i < j$) is generated by the term $\sqrt{|z_{ij}|} \psi_{a(i,j)}(t)$ in $u_i^1(t)$ and the term $-\sqrt{|z_{ij}|} \text{sign}(z_{ij}) \psi_{a(i,j)}(t)$ in $u_j^1(t)$. Hence the inputs $u_i^1(t)$ and $u_j^1(t)$ have the common factor $\psi_{a(i,j)}$. The other terms in the definition of $u_k^1(t)$ for all k , are at different frequencies. Therefore, they are orthogonal to $\psi_{a(i,j)}$ in the inner product space $L_2[0, 2\pi]$, and so do not generate motion in any other symmetric product direction. The second term in the second order input b^2 compensates for the motion excited along bad symmetric product directions. Its presence is a key difference with respect to the algorithms in [Leonard and Krishnaprasad 1995] for driftless systems.

One of the drawbacks of the previous algorithm is that the input functions contain relatively high frequencies, e.g., in an $m = 3$ input system, the input functions contain sinusoids with frequency from 1 to $m(m-1) = 6$. This can be mitigated by optimising the design of the coefficients $\{z_i, z_{jk}\}$ and the numbering of the set P . For example, the design of the coefficients can be optimised by employing a weighted pseudo-inverse.

5. Control algorithms from motion primitives

In this section we design motion control algorithms based on the approximations in Proposition 4.1 and the inversion algorithm in Lemma 4.2. Condition (A1) on the group G , and condition (A2) on the controllability of the system are assumed. We start by designing two primitive motion patterns, **Maintain-Velocity** and **Change-Velocity**, that provide the system with some basic motion capabilities. We then focus on more complex control algorithms to solve the point-to-point reconfiguration problem, the local exponential stabilisation problem and the static interpolation problem.

5.1. Primitives of motion. We describe two basic maneuvers that each last 2π units of time. The parameter $\sigma \ll 1$ is a small positive constant. To maintain a velocity of order $O(\sigma)$, an input of order $O(\sigma)$ suffices, while to obtain a change in velocity of order $O(\sigma)$, we employ a control input of order $O(\sqrt{\sigma})$. Each primitive is described in terms of initial configuration and velocity, input design, and final configuration and velocity.

Maintain-Velocity(σ, ξ_{ref}): keeps the body velocity $\xi(t)$ close to a reference value $\sigma\xi_{\text{ref}}$.

$$\text{Initial state: } \begin{aligned} g(0) &= g_0, \\ \xi(0) &= \sigma\xi_{\text{ref}} + \sigma^2\xi_{\text{error}}, \end{aligned}$$

$$\text{Input: } \begin{aligned} \epsilon &= \sigma, \\ (b^1, b^2) &= \text{Inverse}(\pi(\xi_{\text{ref}} : \xi_{\text{ref}}) - \xi_{\text{error}}), \end{aligned}$$

$$\text{Final state: } \begin{aligned} \log(g_0^{-1}g(2\pi)) &= 2\pi\sigma\xi_{\text{ref}} + \pi\sigma^2\xi_{\text{error}} + O(\sigma^3), \\ \xi(2\pi) &= \sigma\xi_{\text{ref}} + O(\sigma^3). \end{aligned}$$

Change-Velocity($\sigma, \xi_{\text{final}}$): steer the body velocity $\xi(t)$ to a final value $\sigma\xi_{\text{final}}$.

$$\text{Initial state: } \begin{aligned} g(0) &= g_0, \\ \xi(0) &= \sigma\xi_0, \end{aligned}$$

$$\text{Input } \begin{aligned} \epsilon &= \sqrt{\sigma}, \\ (b^1, b^2) &= \text{Inverse}(\xi_{\text{final}} - \xi_0), \end{aligned}$$

$$\text{Final state: } \begin{aligned} \log(g_0^{-1}g(2\pi)) &= \pi\sigma(\xi_0 + \xi_{\text{final}}) + O(\sigma^{3/2}), \\ \xi(2\pi) &= \sigma\xi_{\text{final}} + O(\sigma^2). \end{aligned}$$

The statements on the final configuration and velocity of the primitives are proved in Appendix A.3. Note that the magnitude of control input is

$$\begin{aligned} \|\pi(\xi_{\text{ref}} : \xi_{\text{ref}}) - \xi_{\text{error}}\|O(\sigma), & \quad \text{during a Maintain-Velocity}(\sigma, \xi_{\text{ref}}) \text{ primitive,} \\ \|\xi_{\text{final}} - \xi_0\|O(\sqrt{\sigma}), & \quad \text{during a Change-Velocity}(\sigma, \xi_{\text{final}}) \text{ primitive.} \end{aligned}$$

We conclude this section by showing how to compute estimates of final configurations after multiple periods of control. The following result is a direct consequence of the Campbell-Baker-Hausdorff formula, see for example [Sattinger and Weaver 1986].

5.1 LEMMA: *Let $\sigma \ll 1$ be a positive constant and let g_0, g_1 be group elements. Define the exponential coordinates $y_0 = \log(g_0) \in \mathfrak{g}$ and $y_1 = \log(g_1) \in \mathfrak{g}$. If the vector $[y_0, y_1]$ is higher order in σ than $(y_0 + y_1)$, then it holds*

$$\log(g_0g_1) = y_0 + y_1 + O([y_0, y_1]).$$

5.2. Control algorithms. We present three algorithms to solve various motion control problems. These algorithms combine the two motion primitives with a discrete-time feedback. This makes the approximations hold over multiple time intervals; for example, over a time interval of order $1/\sigma$.

Point-to-point reconfiguration problem. This motion task reconfigures the system, i.e., changes its position and orientation, starting and ending at zero velocity. We assume that the initial state is $(g(0), \xi(0)) = (g_0, 0_{\mathfrak{g}})$ and the final desired state is $(g_1, 0_{\mathfrak{g}})$. For

Goal:	drive system from $(g_0, 0_{\mathfrak{g}})$ to $(g_1, 0_{\mathfrak{g}})$.
Arguments:	(g_0, g_1, σ) .
Require:	$\log(g_0^{-1}g_1)$ well defined.
1:	$N \leftarrow \text{Floor}(\ \log(g_0^{-1}g_1)\ /(2\pi\sigma))$ {Floor(x) is the greatest integer less than or equal to x .}
2:	$\xi_{\text{nom}} \leftarrow \log(g_0^{-1}g_1)/(2\pi\sigma N)$
3:	Change-Velocity (σ, ξ_{nom}) {start maneuver}
4:	for $k = 1$ to $(N - 1)$ do
5:	Maintain-Velocity (σ, ξ_{nom})
6:	end for
7:	Change-Velocity ($\sigma, 0_{\mathfrak{g}}$) {stop maneuver}

TABLE 1. Constant Velocity Algorithm for point-to-point reconfiguration.

simplicity, we require $\log(g_0^{-1}g_1)$ to be well defined, even though this assumption can be removed. For example, on $SO(3)$ the logarithm is well defined whenever the change in attitude is less than π .

The algorithm consists of three steps. Over the first time interval, we change the velocity to an appropriate reference value. We then maintain the velocity close to this constant reference value for an appropriate number of periods. Finally, we stop the system when close to the desired configuration. The details are described in Table 1 and the proof of the following lemma can be found in Appendix A.4. An alternative algorithm which uses a constant acceleration followed by a constant deceleration to achieve the same point-to-point reconfiguration is described in [Bullo and Leonard 1997].

5.2 LEMMA: (Constant Velocity Algorithm) *Let σ be a sufficiently small positive constant and let $(g(0), \xi(0)) = (g_0, O(\sigma^2))$ and let g_1 be a group element such that $\log(g_0^{-1}g_1)$ is well defined. Let $N \in \mathbb{N}$ and the inputs $(b^1, b^2)(t)$ for $t \in [0, 2(N+1)\pi]$ be determined according to the algorithm in Table 1. At final time it holds*

$$\begin{aligned} \log(g(2(N+1)\pi)^{-1}g_1) &= O(\sigma^{3/2}), \\ \xi(2(N+1)\pi) &= O(\sigma^2). \end{aligned}$$

The final state is not exactly as desired, instead there are errors of order $O(\sigma^{3/2})$ and $O(\sigma^2)$. This undesirable feature can be compensated for by solving the next motion problem, the point stabilisation problem.

Point stabilisation problem. This motion task asymptotically stabilises the configuration $g(t)$ to a desired value that we assume without loss of generality to be the identity. Convergence is ensured as long as the initial condition satisfies

$$\|(\log(g(0)), \xi(0))\| \leq \sigma, \tag{5.1}$$

where σ is a sufficiently small positive constant. Should equation (5.1) not hold, then the previous algorithm can be employed to steer the state to an admissible value.

Goal:	drive system to the state $(\text{Id}, 0_{\mathfrak{g}})$ exponentially as $t \rightarrow \infty$.
Arguments:	σ .
Require:	$\ (\log(g(0)), \xi(0))\ \leq \sigma$.
1:	for $k = 1$ to $+\infty$ do
2:	$t_k \leftarrow 4k\pi$ {t_k is the current time}
3:	$\sigma_k \leftarrow \ (\log(g(t_k)), \xi(t_k))\ $
4:	Change-Velocity $(\sigma_k, -(\log(g(t_k)) + \pi\xi(t_k))/(2\pi\sigma_k))$
5:	Change-Velocity $(\sigma_k, 0_{\mathfrak{g}})$
6:	end for

TABLE 2. Local Exponential Stabilisation Algorithm.

The key idea of the algorithm is to iterate the following procedure: measure the state at time t_k and design control inputs that try to steer the state to the desired value $(\text{Id}, 0_{\mathfrak{g}})$ at time $t_{k+1} = t_k + 4\pi$. Since we impose two requirements, one on the final configuration and one on the final velocity, two calls to the **Change-Velocity** primitive are needed. The idea of iterating an approximate stabilisation step for fast convergence can be found, for example, in [Lafferriere and Sussmann 1991]. The details are described in Table 2 and the proof of the following lemma is in Appendix A.5.

5.3 LEMMA: (Local Exponential Stabilisation Algorithm) *Let σ be a sufficiently small positive constant and assume the initial condition satisfies equation (5.1). Let the inputs $(b^1(t), b^2(t))$ be determined according to the algorithm in Table 2 and let $t_k = 4k\pi$. Then there exists a $\lambda > 0$ such that*

$$\|(\log(g(t_k)), \xi(t_k))\| \leq \|(\log(g(0)), \xi(0))\| e^{-\lambda t_k}, \quad \forall k \in \mathbb{N}.$$

Additionally, for $t \in [4k\pi, 4(k+1)\pi]$ it holds that $\|(\log(g(t)), \xi(t))\| = O(e^{-\lambda k/2})$.

Static interpolation problem. This motion task steers the system's configuration along a path connecting the set of the ordered points $\{g_0, g_1, \dots, g_M\}$. As above, we require $\log(g_{k-1}^{-1}g_k)$ to be well defined for $1 \leq k \leq M$. The algorithm consists of M repeated constant velocity (point-to-point) maneuvers (Table 1), with the only difference being that when the configuration reaches the the k th desired value g_k , the velocity gets changed directly to the reference value for the next interval, i.e., without stopping. The details are described in Table 3. It can be shown that the configuration $g(t)$ follows a path passing through the points $\{g_0, g_1, \dots, g_M\}$ with an error of order σ . We do not include a full proof of convergence as it is very similar to the one for Lemma 5.2.

5.4 REMARK: (Interpolating sequences of relative equilibria versus constant velocity motions) The Constant Velocity and the Static Interpolation Algorithms provide two different solutions to the reconfiguration problem. These two algorithms can be compared on the basis of an input cost of the form

$$\|u\|_{[0, T]} = \int_0^T L(u(t)) dt,$$

Goal:	drive system through points $\{g_0, g_1, \dots, g_M\}$.
Arguments:	$(g_0, g_1, \dots, g_M, \sigma)$.
Require:	$(g(0), \xi(0)) = (g_0, 0_{\mathfrak{g}})$ and $\log(g_i^{-1}g_j)$ well defined for all $0 \leq i \leq M$.
1:	for $j = 1$ to M do
2:	$g_{\text{tmp}, j} \leftarrow g(t) \exp(\pi\xi(t))$ {t is the current time}
3:	$N_j \leftarrow \text{Floor}(\ \log(g_{\text{tmp}, j}^{-1}g_j)\ /(2\pi\sigma))$
4:	$\xi_{\text{nom}, j} \leftarrow \log(g_{\text{tmp}, j}^{-1}g_j)/(2\pi\sigma N_j)$
5:	Change-Velocity $(\sigma, \xi_{\text{nom}, j})$
6:	for $k = 1$ to $(N_j - 1)$ do
7:	Maintain-Velocity $(\sigma, \xi_{\text{nom}, j})$
8:	end for
9:	end for
10:	Change-Velocity $(\sigma, 0_{\mathfrak{g}})$

TABLE 3. Static Interpolation Algorithm.

where $T = T(\sigma)$ is the time required to complete the maneuver and $L : \mathcal{U}^m \mapsto \mathbb{R}$ is a cost on the space of input functions. In the following we let g_i and g_f denote initial and final (desired) configurations and we let $\mathcal{P} = \{g_0 = g_i, g_1, \dots, g_M = g_f\}$ be a sequence of configurations such that $\log(g_{j-1}^{-1}g_j)$ is a relative equilibrium vector for all $j = 1, \dots, M$. Recall that $\eta \in \mathfrak{g}$ is a relative equilibrium vector if $\langle \eta : \eta \rangle$ vanishes.

1. The Constant Velocity Algorithm to go from g_0 to g_f involves 2 calls to the **Change-Velocity** primitive and $(N - 1)$ calls to the **Maintain-Velocity** primitive. Using notation from Table 1 and some of the details in Appendix A.4, the cost of the complete maneuver can be computed as

$$\|u\|_{[0, T]} = 2O(\sqrt{\sigma}) + (N - 1) \|\langle \xi_{\text{nom}} : \xi_{\text{nom}} \rangle\| O(\sigma) = O(1),$$

since $\|\langle \xi_{\text{nom}} : \xi_{\text{nom}} \rangle\|$ is of order 1 and N is of order $1/\sigma$.

2. The Static Interpolation Algorithm applied to the set of configurations \mathcal{P} involves $(M + 2)$ calls to the **Change-Velocity** primitive and $(\sum_{j=1}^M N_j)$ calls to the **Maintain-Velocity** primitive. With the notation in Table 3, a little algebra shows that

$$\|u\|_{[0, T]} = (M + 2)O(\sqrt{\sigma}) + (\sum_j N_j) \|\langle \xi_{\text{nom}, j} : \xi_{\text{nom}, j} \rangle\| O(\sigma).$$

Since the configuration $g(t)$ follows the path determined by the set \mathcal{P} with an error of order σ , and since $\log(g_{j-1}^{-1}g_j)$ is a relative equilibrium vector, it can be shown that $\|\langle \xi_{\text{nom}, j} : \xi_{\text{nom}, j} \rangle\| = O(\sigma)$. Summarising, the total cost is

$$\|u\|_{[0, T]} = (M + 2)O(\sqrt{\sigma}) + (\sum_j N_j)O(\sigma^2) = O(\sqrt{\sigma}).$$

We conclude that for small σ (or equivalently, for long final times $T = O(1/\sigma)$), moving along a set of relative equilibria is a more efficient strategy than the Constant Velocity Algorithm. In other words, planning a path along relative equilibria takes into account proper account the full system's dynamics and leads to some notion of optimality.

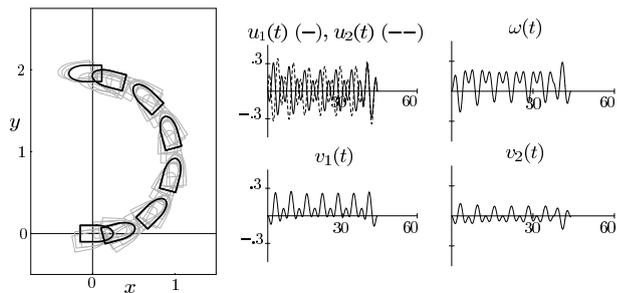


FIGURE 4. Constant Velocity Algorithm. The bullet-shaped objects drawn in the left picture represent the location of the planar body. Darker objects correspond to the location of the body at the beginning and end of a primitive. The plots on the right display the time-evolution of variables $(u_1, u_2, \omega, v_1, v_2)$.

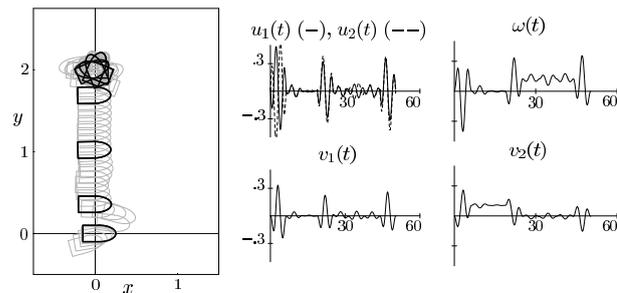


FIGURE 5. Static Interpolation Algorithm. See the figure above for an explanation of pictures. The planar body moves first along the y -axis (from $(0, 0, 0)$ to $(0, 0, 2)$) and then rotates to the desired final configuration $(\pi, 0, 2)$.

5.3. Numerical simulations. The three algorithms introduced above have been implemented on a planar rigid body with two forces a distance h from the centre of mass and with two different masses along the body-fixed axes (to account for added mass when the body is in a fluid). This example is very similar to [PRB3] of Section 2.3. The parameter values in normalised units were chosen to be $J = 1, m_1 = .6, m_2 = 1, h = 2$. For both the Constant Velocity Algorithm and the Static Interpolation Algorithm, we let the initial configuration be the identity and the final (desired) configuration consist of a rotation of π and a translation of 2 units along the y -axis, i.e., $g_{\text{initial}} = (0, 0, 0)$ and $g_{\text{final}} = (\pi, 0, 2)$. We set $\sigma = .1$. For all three algorithms, the numerical results were in agreement with the theoretical analysis presented above.

Constant Velocity, Table 1: Figure 4 illustrates how the velocity variables have a constant average value plus an oscillatory component. Despite the oscillations (see the light gray configurations in Figure 4), the configuration variables evolve along a screw motion toward the desired configuration.

Static Interpolation, Table 3: For comparison, we next present the numerical results of the Static Interpolation Algorithm. The initial and final (desired) configurations are the same as in the previous run. The set of ordered configuration points is $\{(0, 0, 0), (0, 0, 2), (\pi, 0, 2)\}$. In Figure 5 one can notice the path in the x, y plane (consisting of a straight line and a rotation) and the various calls to the **Change-Velocity** and **Maintain-Velocity** primitives (for example, the time history of $u_1, u_2(t)$ shows peaks whenever a **Change-Velocity** maneuver occurs).

Local Exponential Stabilisation, Table 2: Finally, we present the stabilisation algorithm. Starting from the final condition of the Constant Velocity Algorithm, we

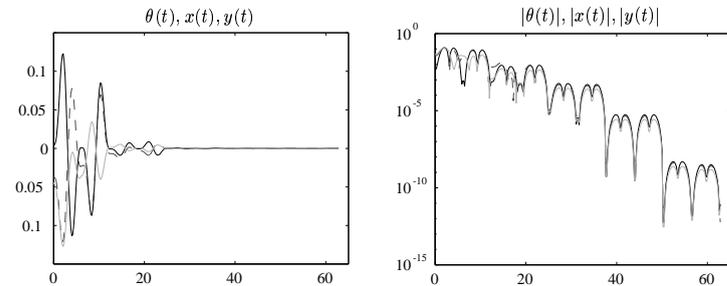


FIGURE 6. Local Stabilisation algorithm. We show only the configuration variables θ (with a solid line), x (with a dashed line) and y (with a light gray line). The initial condition of the simulation is the final state from the simulation of the Constant Velocity algorithm.

applied the local stabilisation algorithm to steer the system exactly to the identity. Figure 6 illustrates how the convergence is exponential.

6. Conclusions

In this paper, we study motion planning and control of underactuated mechanical systems with a focus on underactuated vehicles. We propose a geometric framework encompassing analysis and synthesis tools and show its application to numerous examples. First, we characterise the controllability properties of these systems and investigate their behaviour under small-amplitude forcing. Next, we design two motion primitives which we use in higher-level motion control algorithms to steer the vehicle from point to point, to move the vehicle between points in configuration space without stopping, and to provide exponential stabilisation of the vehicle to a desired configuration. Exponential stabilisation is achieved using time-varying, continuous feedback control. The results in this paper complement the controllability analysis of Lewis and Murray [1997a] and the averaging techniques of Leonard and Krishnaprasad [1995]. Future research will focus on how to overcome the assumption of small-amplitude forcing and the absence of dissipation in our models, see [Leonard 1995] for some results in this direction.

Acknowledgements. The first author thanks Prof. Richard M. Murray for his constant support. The research of the first author is, and of the third author was, supported in part by the National Science Foundation under Grant CMS-9502224. The research of the second author is supported in part by the National Science Foundation under grant BES-9502477 and by the Office of Naval Research under grant N00014-96-1-0052. Dr Lewis wishes to acknowledge the Department of Control and Dynamical Systems which employed him during the term in which this research was conducted.

References

- Baillieul, J. [1993] *Stable average motions of mechanical systems subject to periodic forcing*, in *Dynamics and Control of Mechanical Systems*, M. J. Enos, editor, volume 1, pages 1–23, Fields Institute, ISBN 0-821-89200-2.
- Bonnard, B. [1984] *Contrôlabilité de systèmes mécaniques sur les groupes de Lie*, SIAM Journal on Control and Optimization, **22**(5), 711–722.
- Bullo, F. and Leonard, N. E. [1997] *Motion control for underactuated mechanical systems on Lie groups*, in *Proceedings of the 1997 European Control Conference*, Brussels.
- Bullo, F. and Lewis, A. D. [1996] *Configuration controllability of mechanical systems on Lie groups*, Proceedings of MTNS '96.
- Crouch, P. E. [1981] *Geometric structures in systems theory*, Institution of Electrical Engineers. Proceedings. D. Control Theory and Applications, **128**(5), 242–252.
- [1984] *Spacecraft attitude control and stabilization: Application of geometric control theory to rigid body models*, Institute of Electrical and Electronics Engineers. Transactions on Automatic Control, **29**(4), 321–331.
- Fomenko, A. T. and Chakon, R. V. [1990] *Recursion relations for homogeneous terms of a convergent series of the logarithm of a multiplicative integral on Lie groups*, Functional Analysis and its Applications, **24**(1), 48–58, translated from Russian.

- Gurvits, L. [1992] *Averaging approach to nonholonomic motion planning*, in *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2541–2546, Institute of Electrical and Electronics Engineers.
- Kelly, S. D. and Murray, R. M. [1995] *Geometric phases and locomotion*, The Journal of Robotic Systems, **12**(6), 417–431.
- Khalil, H. K. [1996] *Nonlinear Systems*, second edition, Prentice-Hall, Englewood Cliffs, NJ, ISBN 0-13-228024-8.
- Lafferriere, G. and Sussmann, H. J. [1991] *Motion planning for controllable systems without drift*, in *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1148–1153, Institute of Electrical and Electronics Engineers.
- Lamb, H. [1993] *Hydrodynamics*, Cambridge Mathematical Library, Cambridge University Press, New York/Port Chester/Melbourne/Sydney, reprint of 1932 6th edition.
- Leonard, N. E. [1995] *Periodic forcing, dynamics and control of underactuated spacecraft and underwater vehicles*, in *Proceedings of the 34th IEEE Conference on Decision and Control*, pages 1131–1136, Institute of Electrical and Electronics Engineers, New Orleans, LA.
- Leonard, N. E. and Krishnaprasad, P. S. [1995] *Motion control of drift-free, left-invariant systems on Lie groups*, Institute of Electrical and Electronics Engineers. Transactions on Automatic Control, **40**(9), 1539–1554.
- Lewis, A. D. [1997] *Local configuration controllability for a class of mechanical systems with a single input*, in *Proceedings of the European Control Conference*, Brussels, Belgium.
- [1998] *Affine connections and distributions with applications to nonholonomic mechanics*, Reports on Mathematical Physics, **42**(1/2), 135–164.
- Lewis, A. D. and Murray, R. M. [1997a] *Controllability of simple mechanical control systems*, SIAM Journal on Control and Optimization, **35**(3), 766–790.
- [1997b] *Decompositions of control systems on manifolds with an affine connection*, Systems & Control Letters, **31**(4), 199–205.
- Manikonda, V. and Krishnaprasad, P. S. [1997] *Controllability of Lie-Poisson reduced dynamics*, in *Proceedings of the American Control Conference*, Albuquerque, NM.
- Marsden, J. E. and Ratiu, T. S. [1999] *Introduction to Mechanics and Symmetry*, second edition, number 17 in Texts in Applied Mathematics, Springer-Verlag, New York-Heidelberg-Berlin, ISBN 0-387-98643-X.
- M'Closkey, R. T. and Murray, R. M. [1997] *Exponential stabilization of driftless nonlinear control systems using homogeneous feedback*, Institute of Electrical and Electronics Engineers. Transactions on Automatic Control, **42**(5), 614–628.
- Morin, P. and Samson, C. [1997] *Time-varying exponential stabilization of a rigid spacecraft with two control torques*, Institute of Electrical and Electronics Engineers. Transactions on Automatic Control, **42**(4), 528–534.

Murray, R. M., Li, Z. X., and Sastry, S. S. [1994] *A Mathematical Introduction to Robotic Manipulation*, CRC Press, 2000 Corporate Blvd., N.W., Boca Raton, Florida 33431, ISBN 0-8493-7981-4.

Ostrowski, J. P. and Burdick, J. W. [1997] *Controllability tests for mechanical systems with constraints and symmetries*, Journal of Applied Mathematics and Computer Science, **7**(2), 101–127.

Pettersen, K. Y. and Egeland, O. [1997] *Robust attitude stabilization of an underactuated AUV*, in *Proceedings of the European Control Conference*, Brussels, Belgium.

Rathinam, M. and Murray, R. M. [1998] *Configuration flatness of Lagrangian systems underactuated by one control*, SIAM Journal on Control and Optimization, **36**(1), 164–179.

Rui, C., Kolmanovsky, I. V., McNally, P. J., and McClamroch, N. H. [1996] *Attitude control of underactuated multibody spacecraft*, in *IFAC World Conference*, San Francisco, CA.

Sattinger, D. H. and Weaver, O. L. [1986] *Lie Groups and Algebras, with Applications to Physics, Geometry, and Mechanics*, number 61 in Applied Mathematical Sciences, Springer-Verlag, New York-Heidelberg-Berlin.

Sordalen, O. J. and Egeland, O. [1995] *Exponential stabilization of nonholonomic chained systems*, Institute of Electrical and Electronics Engineers. Transactions on Automatic Control, **40**(1), 35–49.

Sussmann, H. J. [1987] *A general theorem on local controllability*, SIAM Journal on Control and Optimization, **25**(1), 158–194.

Sussmann, H. J. and Liu, W. [1991] *Limits of highly oscillatory controls and the approximation of general paths by admissible trajectories*, in *Proceedings of the 31st IEEE Conference on Decision and Control*, pages 437–442, Institute of Electrical and Electronics Engineers, Brighton, England.

A. Proofs

A.1. Proposition 4.1.

Proof. We start by proving the validity of the expansion in $\xi(t, \epsilon)$ by means of the standard perturbation method, as presented, for example, by Khalil [1996]. Consider the ordinary differential equation

$$\dot{x} = f(x) + g(t, \epsilon)$$

and let $x(t, \epsilon)$ denote the solution from initial condition $x_0(\epsilon)$. At $\epsilon = 0$, suppose that $f(x_0(0)) = g(t, 0) = 0$, so that $x(t, 0) = x_0(0)$ is a constant solution. We now expand $x(t, \epsilon)$ and $g(t, \epsilon)$ in a Taylor series about the value $\epsilon = 0$ and write

$$x(t, \epsilon) = \sum_{i=0}^{\infty} \epsilon^i x^i(t) \quad \text{and} \quad g(t, \epsilon) = \sum_{i=0}^{\infty} \epsilon^i g^i(t).$$

As shown by Khalil [1996], the components in the expansion of x satisfy the following differential equations

$$\dot{x}^n(t) = \frac{1}{n!} \frac{\partial^n}{\partial \epsilon^n} \Big|_{\epsilon=0} f(x(t, \epsilon)) + g^n(t),$$

with initial condition $x^n(0) = \frac{1}{n!} \frac{\partial^n}{\partial \epsilon^n} \Big|_{\epsilon=0} x_0(\epsilon)$.

The differential equation of interest in our case is equation (2.4):

$$\dot{\xi} = -\frac{1}{2} \langle \xi : \xi \rangle + \epsilon b^1(t) + \epsilon^2 b^2(t),$$

and the initial condition is $\xi(0, \epsilon) = \epsilon \xi_0^1 + \epsilon^2 \xi_0^2$. The constant solution we expand about is $\xi(t, 0) = \xi^0(t) = 0$.

Differentiating the function $f(\xi(\epsilon)) = -\frac{1}{2} \langle \xi : \xi \rangle$, we have

$$\begin{aligned} \frac{\partial f}{\partial \epsilon} &= - \left\langle \xi : \frac{\partial \xi}{\partial \epsilon} \right\rangle \\ \frac{\partial^2 f}{\partial \epsilon^2} &= - \left\langle \frac{\partial \xi}{\partial \epsilon} : \frac{\partial \xi}{\partial \epsilon} \right\rangle - \left\langle \xi : \frac{\partial^2 \xi}{\partial \epsilon^2} \right\rangle \\ \frac{\partial^3 f}{\partial \epsilon^3} &= -3 \left\langle \frac{\partial \xi}{\partial \epsilon} : \frac{\partial^2 \xi}{\partial \epsilon^2} \right\rangle - \left\langle \xi : \frac{\partial^3 \xi}{\partial \epsilon^3} \right\rangle, \end{aligned}$$

and noting that $\frac{\partial^n}{\partial \epsilon^n} \Big|_{\epsilon=0} \xi = n! \xi^n$, we have

$$\begin{aligned} \frac{\partial f}{\partial \epsilon} \Big|_{\epsilon=0} &= - \langle \xi^0 : \xi^1 \rangle \\ \frac{\partial^2 f}{\partial \epsilon^2} \Big|_{\epsilon=0} &= - \langle \xi^1 : \xi^1 \rangle - 2 \langle \xi^0 : \xi^2 \rangle \\ \frac{\partial^3 f}{\partial \epsilon^3} \Big|_{\epsilon=0} &= -6 \langle \xi^1 : \xi^2 \rangle - 6 \langle \xi^0 : \xi^3 \rangle. \end{aligned}$$

Next, we write the differential equations as described above. Recalling that $\xi^0(t) = 0$ we have

$$\begin{aligned} \dot{\xi}^1 &= b^1 \\ \dot{\xi}^2 &= -\frac{1}{2} \langle \xi^1 : \xi^1 \rangle + b^2 \\ \dot{\xi}^3 &= - \langle \xi^1 : \xi^2 \rangle. \end{aligned}$$

Initial conditions are $\xi^1(0) = \xi_0^1$, $\xi^2(0) = \xi_0^2$, $\xi^3(0) = 0$. Finally, we employ the notation introduced in Section 5 to integrate the three ODEs,

$$\begin{aligned} \xi^1(t) &= \xi_0^1 + \overline{b^1}(t) \\ \xi^2(t) &= \xi_0^2 - \frac{1}{2} \overline{\langle \xi_0^1 + \overline{b^1}(t) : \xi_0^1 + \overline{b^1}(t) \rangle} + \overline{b^2}(t) \\ \xi^3(t) &= - \overline{\langle \xi_0^1 + \overline{b^1}(t) : \xi_0^2 - \frac{1}{2} \overline{\langle \xi_0^1 + \overline{b^1}(t) : \xi_0^1 + \overline{b^1}(t) \rangle} + \overline{b^2}(t) \rangle}. \end{aligned}$$

Expanding the terms on the right hand side, one recovers all of the terms in the expansions of $\xi(t, \epsilon)$ in Proposition 4.1.

In the second part of the proof we prove the validity of the expansion of $x(t, \epsilon)$ by means of the approximate solutions for kinematic systems obtained by Fomenko and Chakon [1990] and used by Leonard and Krishnaprasad [1995]. From these references we know that, if $\xi(t, \epsilon) = O(\epsilon)$, then

$$x(t, \epsilon) = \bar{\xi}(t) + \frac{1}{2} \overline{[\xi, \bar{\xi}]}(t) + O(\epsilon^3).$$

Substituting $\xi(t, \epsilon) = \epsilon \xi^1(t) + \epsilon^2 \xi^2(t) + O(\epsilon^3)$, we have:

$$x(t, \epsilon) = \overline{\epsilon \xi^1(t) + \epsilon^2 \xi^2(t)} + \frac{1}{2} \epsilon^2 \overline{[\xi^1, \bar{\xi}^1]}(t) + O(\epsilon^3).$$

And substituting the values for $\xi^1(t)$ and $\xi^2(t)$, and writing $x(t, \epsilon) = \epsilon x^1(t) + \epsilon^2 x^2(t) + O(\epsilon^3)$, we have

$$\begin{aligned} x^1(t) &= \overline{\xi_0^1 + \bar{b}^1}(t) \\ x^2(t) &= \left(\xi_0^2 t - \langle \xi_0^1 : \xi_0^1 \rangle \frac{t^2}{4} - \left\langle \xi_0^1 : \overline{\bar{b}^1} \right\rangle(t) + \left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right)(t) \right) + \frac{1}{2} \overline{[\xi_0^1 + \bar{b}^1, \xi_0^1 + \bar{b}^1]}(t). \end{aligned}$$

Expanding the terms on the right hand side, one recovers all of the terms in the expansions of $x(t, \epsilon)$ in Proposition 4.1. \blacksquare

A.2. Lemma 4.2. Here, we prove the claim (4.7) in Lemma 4.2.

Proof: We start by studying the properties of the functions $\psi_a(t)$. A direct computation shows that for all a, b, c

$$\overline{\psi_a}(2\pi) = \overline{\psi_a}(2\pi) = \overline{\overline{\psi_a}}(2\pi) = 0 \quad (\text{A.1})$$

$$\overline{\psi_a \psi_b}(t) = \frac{\delta_{ab}}{2\pi} t + r_{ab}(t), \quad \text{where} \quad r_{ab}(2\pi) = \overline{r_{ab}}(2\pi) = 0, \quad (\text{A.2})$$

$$\overline{\psi_a t}(2\pi) = \overline{\overline{\psi_a \psi_b}}(2\pi) = \overline{\psi_a r_{bc}}(2\pi) = 0, \quad (\text{A.3})$$

where δ_{ab} is the Kronecker delta function. The proof of these properties is straightforward but tedious.

Next, we prove the claim (4.7) in Lemma 4.2. Given the definition in equation (4.5) and the property (A.2) of the functions $\psi_a(t)$, we compute the quantity $\overline{\langle \bar{b}^1 : \bar{b}^1 \rangle}(2\pi)$ as

$$\begin{aligned} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle}(2\pi) &= \sum_{1 \leq j < k \leq m} \sum_{1 \leq p < q \leq m} \sqrt{|z_{jk} z_{pq}|} \langle (b_j - \text{sign}(z_{jk}) b_k) : (b_p - \text{sign}(z_{pq}) b_q) \rangle \overline{\psi_{a(j,k)} \overline{\psi_{a(p,q)}}}(2\pi) \\ &= \sum_{1 \leq j < k \leq m} |z_{jk}| \langle (b_j - \text{sign}(z_{jk}) b_k) : (b_j - \text{sign}(z_{jk}) b_k) \rangle \\ &= \sum_{1 \leq j < k \leq m} |z_{jk}| \left(\langle b_j : b_j \rangle - 2 \text{sign}(z_{jk}) \langle b_j : b_k \rangle + \langle b_k : b_k \rangle \right) \\ &= -2 \sum_{1 \leq j < k \leq m} z_{jk} \langle b_j : b_k \rangle + \sum_{1 \leq j < k \leq m} |z_{jk}| \left(\langle b_j : b_j \rangle + \langle b_k : b_k \rangle \right). \end{aligned}$$

Summing up with $\overline{b^2}(2\pi) = 2\pi b^2$, we have

$$\left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right)(2\pi) = \sum_{1 \leq i \leq m} z_i b_i + \sum_{1 \leq j < k \leq m} z_{jk} \langle b_j : b_k \rangle,$$

which proves equation (4.7). \blacksquare

A.3. Primitives of Motion. The statements in the description of both primitives are direct consequences of the following result.

A.1 PROPOSITION: *Let the assumptions of Proposition 4.1 hold and let $(b^1(t), b^2(t)) = \text{Inverse}(\eta)$. If $\xi(0) = \epsilon \xi_0^1 + \epsilon^2 \xi_0^2$, we have*

$$\xi(2\pi) = \epsilon \xi_0^1 + \epsilon^2 \left(\xi_0^2 - \pi \langle \xi_0^1 : \xi_0^1 \rangle + \eta \right) + \epsilon^3 \pi \left(\pi \langle \xi_0^1 : \langle \xi_0^1 : \xi_0^1 \rangle \rangle - 2 \langle \xi_0^1 : \xi_0^2 \rangle - \langle \xi_0^1 : \eta \rangle \right) + O(\epsilon^4) \quad (\text{A.4})$$

$$x(2\pi) = \epsilon 2\pi \xi_0^1 + \epsilon^2 \pi \left(2\xi_0^2 - \pi \langle \xi_0^1 : \xi_0^1 \rangle + \eta \right) + O(\epsilon^3). \quad (\text{A.5})$$

Proof: Note that property (A.1) implies directly that $\overline{b^1}(2\pi) = \overline{\bar{b}^1}(2\pi) = \overline{\overline{\bar{b}^1}}(2\pi) = 0$, so that the terms in the Taylor expansion in Proposition 4.1 simplify to

$$\xi^1(2\pi) = \xi_0^1$$

$$\xi^2(2\pi) = \xi_0^2 - \pi \langle \xi_0^1 : \xi_0^1 \rangle + \left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right)(2\pi),$$

$$\begin{aligned} \xi^3(2\pi) &= -2\pi \langle \xi_0^1 : \xi_0^2 \rangle + \pi^2 \langle \xi_0^1 : \langle \xi_0^1 : \xi_0^1 \rangle \rangle - \left\langle \xi_0^1 : \overline{\left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right)} \right\rangle(2\pi) \\ &\quad + \overline{\langle \langle \xi_0^1 : \xi_0^1 \rangle : \bar{b}^1 \rangle} \frac{t}{2}(2\pi) + \overline{\langle \bar{b}^1 : \langle \xi_0^1 : \bar{b}^1 \rangle \rangle}(2\pi) - \overline{\langle \bar{b}^1 : \left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right) \rangle}(2\pi), \end{aligned}$$

$$x^1(2\pi) = 2\pi \xi_0^1,$$

$$x^2(2\pi) = 2\pi \xi_0^2 - \pi^2 \langle \xi_0^1 : \xi_0^1 \rangle + \left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right)(2\pi) + \frac{1}{2} \overline{[\xi_0^1 + \bar{b}^1, \xi_0^1 t + \bar{b}^1]}(2\pi).$$

Hence, $\xi^1(2\pi)$ and $x^1(2\pi)$ are as computed above. We employ Lemma 4.2 and property (A.2) to simplify the remaining terms into

$$\xi^2(2\pi) = \xi_0^2 - \pi \langle \xi_0^1 : \xi_0^1 \rangle + \eta,$$

$$\begin{aligned} \xi^3(2\pi) &= -2\pi \langle \xi_0^1 : \xi_0^2 \rangle + \pi^2 \langle \xi_0^1 : \langle \xi_0^1 : \xi_0^1 \rangle \rangle - \pi \langle \xi_0^1 : \eta \rangle \\ &\quad + \overline{\langle \langle \xi_0^1 : \xi_0^1 \rangle : \bar{b}^1 \rangle} \frac{t}{2}(2\pi) + \overline{\langle \bar{b}^1 : \langle \xi_0^1 : \bar{b}^1 \rangle \rangle}(2\pi) - \overline{\langle \bar{b}^1 : \left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right) \rangle}(2\pi), \end{aligned}$$

$$x^2(2\pi) = 2\pi \xi_0^2 - \pi^2 \langle \xi_0^1 : \xi_0^1 \rangle + \pi \eta + \frac{1}{2} \overline{[\xi_0^1 + \bar{b}^1, \xi_0^1 t + \bar{b}^1]}(2\pi).$$

Regarding the term $\xi^3(t)$, the claim is proven if

$$\overline{\langle \langle \xi_0^1 : \xi_0^1 \rangle : \bar{b}^1 \rangle} \frac{t}{2}(2\pi) = \overline{\langle \bar{b}^1 : \langle \xi_0^1 : \bar{b}^1 \rangle \rangle}(2\pi) = \overline{\langle \bar{b}^1 : \left(b^2 - \frac{1}{2} \overline{\langle \bar{b}^1 : \bar{b}^1 \rangle} \right) \rangle}(2\pi) = 0.$$

However, since $b^1(t)$ is linear combination of the functions $\psi_a(t)$, the latter relations correspond equality sign by equality sign to the properties in equation (A.3). Regarding the term $x^2(t)$, it holds that

$$\left[\xi_0^1 + \overline{b^1}, \xi_0^1 t + \overline{b^1} \right] (2\pi) = 2\pi^2 [\xi_0^1, \xi_0^1] + \left[\xi_0^1, \overline{b^1}(2\pi) \right] + \left[\overline{b^1}, \xi_0^1 t \right] (2\pi) + \left[\overline{b^1}, \overline{b^1} \right] (2\pi) = 0,$$

as all terms in the middle expression vanish. \blacksquare

A.4. Lemma 5.2.

Proof: Given the descriptions of the primitives **Change-Velocity** and **Maintain-Velocity**, we compute the evolution of $\xi(t)$ as follows. Starting from $\xi(0) = O(\sigma^2)$, we have:

$$\begin{aligned} \text{after first Change-Velocity}(\sigma, \xi_{\text{nom}}) &: \xi(2\pi) = \sigma \xi_{\text{nom}} + O(\sigma^2), \\ \text{after first Maintain-Velocity}(\sigma, \xi_{\text{nom}}) &: \xi(4\pi) = \sigma \xi_{\text{nom}} + O(\sigma^3), \\ \text{after } k\text{th step in the for loop} &: \xi(2(k+1)\pi) = \sigma \xi_{\text{nom}} + O(\sigma^3), \\ \text{after the final Change-Velocity}(\sigma, 0) &: \xi(2(N+1)\pi) = O(\sigma^2). \end{aligned}$$

The final value of ξ is therefore as in the claim. Similarly, we can compute the change in configuration during each interval:

$$\begin{aligned} \text{after first Change-Velocity}(\sigma, \xi_{\text{nom}}) &: \log(g(0)^{-1}g(2\pi)) = \pi\sigma\xi_{\text{nom}} + O(\sigma^{3/2}), \\ \text{after first Maintain-Velocity}(\sigma, \xi_{\text{nom}}) &: \log(g(2\pi)^{-1}g(4\pi)) = 2\pi\sigma\xi_{\text{nom}} + O(\sigma^2), \\ \text{after } k\text{th step in the for loop} &: \log(g(2k\pi)^{-1}g(2(k+1)\pi)) = 2\pi\sigma\xi_{\text{nom}} + O(\sigma^3), \\ \text{after final Change-Velocity}(\sigma, 0) &: \log(g(2N\pi)^{-1}g(2(N+1)\pi)) = \pi\sigma\xi_{\text{nom}} + O(\sigma^{3/2}). \end{aligned}$$

We now need to sum the changes in configuration due to each interval by means of the approximation in Lemma 5.1. Combining the contributions during the first two intervals, and recalling that $[\xi_{\text{nom}}, \xi_{\text{nom}}]$ vanishes, we have

$$\log(g_0^{-1}g(4\pi)) = 3\pi\sigma\xi_{\text{nom}} + O(\sigma^{3/2}) =: 3\pi\sigma\xi_{\text{nom}} + \sigma^{3/2}\eta_1, \quad (\text{A.6})$$

where $\eta_1 = O(1)$ is an appropriate vector in \mathfrak{g} . Next, we claim that for all $k = 1, \dots, N-1$, it holds

$$\log(g_0^{-1}g(2(k+1)\pi)) = a_k\xi_{\text{nom}} + \sigma^{3/2}\eta_k,$$

where the scalar a_k and the vector η_k are of order at most 1. We prove the claim by induction. At $k = 1$, we recover equation (A.6), with $a_1 = 3\pi\sigma$. Next, we assume that the claim holds at k , and we prove it for $k+1$. As dictated by Lemma 5.1, we compute the bracket between the current value $a_k\xi_{\text{nom}} + \sigma^{3/2}\eta_k$ and the contribution $2\pi\sigma\xi_{\text{nom}} + O(\sigma^3) =: 2\pi\sigma\xi_{\text{nom}} + \sigma^3\zeta_k$, where $\zeta_k = O(1)$ is an appropriate vector in \mathfrak{g} . We have:

$$[a_k\xi_{\text{nom}} + \sigma^{3/2}\eta_k, 2\pi\sigma\xi_{\text{nom}} + \sigma^3\zeta_k] = O(\sigma^{5/2}),$$

so that

$$\begin{aligned} \log(g_0^{-1}g(2(k+2)\pi)) &= (a_k\xi_{\text{nom}} + \sigma^{3/2}\eta_k) + (2\pi\sigma\xi_{\text{nom}} + \sigma^3\zeta_k) + O(\sigma^{5/2}) \\ &= (a_k + 2\pi\sigma)\xi_{\text{nom}} + \sigma^{3/2}(\eta_k + \sigma^{3/2}\zeta_k + \sigma\nu_k), \end{aligned}$$

where $\nu_k = O(1)$ is an appropriate vector in \mathfrak{g} . Hence, the claim holds at $k+1$, with $a_{k+1} = a_k + 2\pi\sigma$ and $\eta_{k+1} = \eta_k + \sigma^{3/2}\zeta_k + \sigma\nu_k$. At the end of the for loop, as $k = N-1$, we have

$$\log(g_0^{-1}g(2N\pi)) = a_{N-1}\xi_{\text{nom}} + \sigma^{3/2}\eta_{N-1},$$

where we can compute the coefficients as

$$a_{N-1} = a_1 + \sum_{k=2}^{N-1} 2\pi\sigma = (2N-1)\pi\sigma \quad \text{and} \quad \eta_{N-1} = \eta_1 + \sum_{k=2}^{N-1} (\sigma^{3/2}\zeta_k + \sigma\nu_k) = O(1).$$

The contribution of the last interval is $\sigma\pi\xi_{\text{nom}}$ plus some higher-order terms, so that

$$\log(g_0^{-1}g(2(N+1)\pi)) = 2N\pi\sigma\xi_{\text{nom}} + O(\sigma^{3/2}).$$

Finally, we apply the approximation in Lemma 5.1 for a last time to obtain

$$\log(g(2(N+1)\pi)^{-1}g_1) = \log\left((g_0^{-1}g(2(N+1)\pi))^{-1}(g_0^{-1}g_1)\right) = O(\sigma^{3/2}),$$

where we recall that $\log(g_0^{-1}g_1) = 2N\pi\sigma\xi_{\text{nom}}$ and $\log(h^{-1}) = -\log(h)$. \blacksquare

A.5. Lemma 5.3.

Proof: We start by investigating the two **Change-Velocity** primitives described inside the while statement in Algorithm 2. Assuming that at time t_k it holds

$$\|(\log(g(t_k)), \xi(t_k))\| = \sigma_k \ll 1,$$

we claim that

$$\|(\log(g(t_{k+1})), \xi(t_{k+1}))\| = O(\sigma_k^{3/2}). \quad (\text{A.7})$$

This can be seen as follows. By assumption there exist two vectors x_{err} and ξ_{err} of order $O(1)$ such that

$$\begin{aligned} \log(g(t_k)) &= \sigma_k x_{\text{err}} \\ \xi(t_k) &= \sigma_k \xi_{\text{err}}. \end{aligned}$$

With this notation, it holds $-(\log(g(t_k)) + \pi\xi(t_k))/(2\pi\sigma_k) = -(x_{\text{err}} + \pi\xi_{\text{err}})/(2\pi)$. After the primitive **Change-Velocity**($\sigma_k, -(x_{\text{err}} + \pi\xi_{\text{err}})/(2\pi)$), we compute

$$\begin{aligned} \log(g(t_k + 2\pi)) &= \frac{1}{2}\sigma_k(x_{\text{err}} + \pi\xi_{\text{err}}) + O(\sigma_k^{3/2}) \\ \xi(t_k + 2\pi) &= -\sigma_k(x_{\text{err}} + \pi\xi_{\text{err}})/(2\pi) + O(\sigma_k^2), \end{aligned}$$

and after the final **Change-Velocity**($\sigma_k, 0$), we have

$$\begin{aligned} \log(g(t_k + 4\pi)) &= O(\sigma_k^{3/2}), \\ \xi(t_k + 4\pi) &= O(\sigma_k^2). \end{aligned}$$

As $t_{k+1} = t_k + 4\pi$, this proves equation (A.7). The latter equations are equivalent to

$$\|(\log(g(t_{k+1})), \xi(t_{k+1}))\| \leq M_k \sigma_k^{3/2}, \quad (\text{A.8})$$

where the positive scalar M_k depends continuously on initial state and parameters of the system of ordinary differential equations (2.2) and (2.3). The parameters are σ_k and the coefficients in the design of $(b^1(t), b^2(t))$, for $t_k < t < t_{k+1}$. By looking at the details of the

inversion algorithm in Lemma 4.2, these parameters are seen to be continuous function of the initial conditions $(\log(g(t_k)), \xi(t_k))$. Hence, we know that $M_k(g(t_k), \xi(t_k))$ is a continuous function of its arguments and it is therefore bounded in a neighbourhood of the point $(g(t_k), \xi(t_k)) = (\text{Id}, 0)$. In other words, there exist positive constants B_1, B_2 such that

$$\|(\log(g(t_k)), \xi(t_k))\| < B_1 \implies M_k(g(t_k), \xi(t_k)) < B_2.$$

Finally, for some $\alpha < 1$, we set $\sigma = \alpha \min(B_1, 1/B_2^2)$ and we prove by induction that $\sigma_k < \sigma$ and $M_k \sigma_k^{1/2} \leq \alpha$. At $k = 0$, we have by assumption

$$\sigma_0 = \|(\log(g(0)), \xi(0))\| \leq \sigma < B_1,$$

so that $M_0 < B_2$ and

$$M_0 \sigma_0^{1/2} < B_2 \sigma^{1/2} < \alpha < 1.$$

Therefore, the claim holds at $k = 0$. Next, we assume it at k , and prove it for $k + 1$. We rewrite equation (A.8) as

$$\sigma_{k+1} = \|(\log(g(t_{k+1})), \xi(t_{k+1}))\| \leq (M_k \sigma_k^{1/2}) \sigma_k \leq \alpha \sigma < B_1.$$

Hence, M_{k+1} is also bounded by B_2 and we have

$$\sigma_{k+1}^{1/2} M_{k+1} \leq \sigma^{1/2} B_2 < \alpha.$$

This proves that $M_k \sigma_k^{1/2} \leq \alpha$ for all k . In other words, we have that the sequence $\{\sigma_k, k \geq 0\}$ satisfies $\sigma_{k+1} \leq \alpha \sigma_k$ with $\alpha < 1$, or equivalently $\sigma_k \leq \alpha^k \sigma_0$. Therefore, for $\lambda = -\ln \alpha > 0$,

$$\|(\log(g(t_k)), \xi(t_k))\| \leq \|(\log(g(0)), \xi(0))\| e^{-\lambda k}.$$

Finally we prove the last statement in Lemma 5.3. From time t_k to t_{k+1} , the system undergoes two **Change-Velocity** primitives and evolves starting from a state of order $O(\sigma_k) = O(e^{-\lambda k})$ to a final state of higher order. During the two **Change-Velocity** primitives, the input is of order $\sqrt{\sigma_k} = e^{-\lambda k/2}$ (with the notation in Section 5.1 and in Proposition 4.1, it holds $\epsilon = \sqrt{\sigma}$). Therefore, the expansions in Proposition 4.1 show that the state is of order $\sqrt{\sigma_k} = e^{-\lambda k/2}$ from time t_k to t_{k+1} .

■