

# A Discrete Geometric Optimal Control Framework for Systems with Symmetries

Marin Kobilarov  
USC

Mathieu Desbrun  
Caltech

Jerrold E. Marsden  
Caltech

Gaurav S. Sukhatme  
USC

**Abstract**—This paper studies the optimal motion control of mechanical systems through a discrete geometric approach. At the core of our formulation is a discrete *Lagrange-d’Alembert-Pontryagin* variational principle, from which are derived discrete equations of motion that serve as constraints in our optimization framework. We apply this discrete mechanical approach to holonomic systems with symmetries and, as a result, geometric structure and motion invariants are preserved. We illustrate our method by computing optimal trajectories for a simple model of an air vehicle flying through a digital terrain elevation map, and point out some of the numerical benefits that ensue.

## I. INTRODUCTION

The goal of this paper is to design *optimal motion control* algorithms for robotic systems with *symmetries*. That is, we consider the problem of computing the controls  $f(t)$  necessary to move a finite-dimensional mechanical system with configuration space  $Q$  from its initial state  $(q(0) = q_i, \dot{q}(0) = \dot{q}_i)$  to a goal state  $(q(T) = q_f, \dot{q}(T) = \dot{q}_f)$ , while minimizing a cost function of the form:

$$J(q, \dot{q}, f) = \int_0^T C(q(t), \dot{q}(t), f(t)) dt. \quad (1)$$

Minimum control effort problems can be implemented using  $C(q(t), \dot{q}(t), f(t)) = \|f(t)\|^2$ , while minimum-time problems involve  $C(q(t), \dot{q}(t), f(t)) = 1$ . Additional nonlinear equality or inequality constraints on the configuration (such as obstacle avoidance in the environment) and velocity variables can be imposed as well. Systems of interest captured by this formulation include autonomous vehicles such as unmanned helicopters, micro-air vehicles or underwater gliders.

### A. Related work

Trajectory design and motion control of robotic systems have been studied from many different perspectives. Of particular interest are geometric approaches [1, 2, 3] that use symmetry and reduction techniques [4, 5]. Reduction by symmetry can be used to greatly simplify the optimal control problem and provide a framework to compute motions for general nonholonomic systems [6]. A related approach, applied to an underwater eel-like robot, involves finding approximate solutions using truncated basis of cyclic input functions [7]. There are a number of successful methods for motion planning with obstacles—see [8] for references.

While standard optimization methods are based on shooting, multiple shooting, or collocation techniques, recent work on Discrete Mechanics and Optimal Control (DMOC, see [9, 10, 11]) proposes a different discretization strategy. At the core of DMOC is the use of *variational integrators* [12] that are derived from the discretization of variational principles such as Hamilton’s principle for conservative systems or Lagrange-D’Alembert for dissipative systems. Unlike other existing variational approaches [6, 13] where the continuous equations of motion are enforced as constraints and *subsequently* discretized, DMOC *first* discretizes the variational principles underlying the mechanical system dynamics; the resulting discrete equations are then used as constraints along with a discrete cost function to form the control problem. Because the discrete equations of motion result from a discrete variational principle, momenta preservation and symplecticity are automatically enforced, avoiding numerical issues (like numerical dissipation) that generic algorithms often possess.

### B. Contributions

In this paper, we extend the generalized variational principle of [14, 15, 16] to the DMOC framework to derive optimization algorithms based on structure-preserving, discrete-mechanical integrators. In particular, we employ a discrete *Lagrange-d’Alembert-Pontryagin* principle to characterize mechanical systems with symmetries and external forces. We use this new discrete geometric optimal control framework for holonomic systems (possibly underactuated and/or with symmetries) and illustrate the implemented algorithms with a simulated example of a simplified helicopter flying through a canyon.

The numerical benefits of our discrete geometric approach are numerous. First, it automatically preserves motion invariants and geometric structures of the continuous system, exhibits good energy behavior, and respects the work-energy balance due to its variational nature. Such properties are often crucial for numerical accuracy and stability, in particular for holonomic systems such as underwater gliders traveling at low energies along ocean currents. Second, it benefits from an exact reconstruction of curves in the Lie group configuration space from curves in its Lie algebra. Thus, numerical drift, for example associated with enforcing rotation frame orthogonality constraints, is avoided. Third, the simplicity of the variational principle allows flexibility of implementation.

Finally, this framework is flexible enough to strike a balance between a desired order of accuracy and runtime efficiency.

In addition to these well-documented advantages of discrete variational methods, there is growing evidence that DMOC methods are especially well suited for optimization problems. In particular, their discrete variational nature seems to offer very good trajectory approximation even at low temporal resolutions. This stability vis-a-vis resolution is particularly suitable for design and exploration purposes as well as for hierarchical optimizations, as it leads to faster convergence towards optimal solutions.

It is also important to note that non-holonomic constraints can also be imposed in our framework. We refer to [17] for details on rolling constraints and Chaplygin systems. However, in this paper we focus solely on holonomic systems with symmetries.

## II. OVERVIEW OF MECHANICAL INTEGRATORS

A mechanical integrator integrates a dynamical system forward in time. The construction of such numerical algorithms usually involves some form of discretization or Taylor expansion that results in either implicit or explicit equations to compute the next state in time. In an optimal control setting, these equations are then used as constraints.

Instead, the integrators employed in this paper are based on the discretization of variational principles, i.e. *variational integrators*. In essence, they ensure the optimality (in the sense of Hamilton's principle, for instance) of the discrete path of the mechanical system in space-time. In addition, certain systems have group structure and symmetries that can be factored out directly in order to obtain more accurate and efficient integrators, e.g. *Lie group integrators*. After giving a brief overview of such integrators below we present a variational principle to derive more general integrators that account for symmetries.

### A. Variational Integrators

Variational integrators [12] are derived from a variational principle (e.g., Hamilton's principle) using a discrete Lagrangian. Unlike standard integration methods, variational integrators can preserve momenta, energy, and symplectic structure (i.e., a symplectic 2-form in phase space) for conservative systems; in the presence of forces and/or dissipation, they compute the change in these quantities with remarkable accuracy. Such features are obviously desirable for accurate dynamics simulation. The underlying theory has discrete analogs of Noether's theorem and the Legendre transform, and a Lagrange-d'Alembert principle to handle non-conservative forces and constraints. *Discrete mechanics*, therefore, stands as a self-contained theory similar to Hamiltonian or Lagrangian mechanics [15] and has already been applied to several domains: nonsmooth variational collision integration [18], elasticity simulation in computer graphics [14], satellite formation trajectory design [19], optimal control of rigid bodies [11], of

articulated bodies in fluid [10, 20], and optimal control of wheeled robots [21].

In the variational integration setting, the state space  $TQ$  is replaced by a product of two manifolds  $Q \times Q$  [12]. Thus, a velocity vector  $(q, \dot{q}) \in TQ$  is represented by a pair of points  $(q_0, q_1) \in Q \times Q$ . A path  $q : [0, T] \rightarrow Q$  is replaced by a discrete path  $q_d : \{kh\}_{k=0}^N \rightarrow Q$  ( $q_d = \{q_0, \dots, q_N\}$ ,  $q_k = q(kh)$ ),  $Nh = T$ . One formulates a discrete version of Hamilton's principle (i.e.  $\delta \int_0^T L(q, \dot{q}) dt = 0$ ) by approximating the integral of the Lagrangian  $L : TQ \rightarrow \mathbb{R}$  between  $q_k$  and  $q_{k+1}$  by a discrete Lagrangian  $L_d : Q \times Q \rightarrow \mathbb{R}$

$$L_d(q_k, q_{k+1}) \approx \int_{kh}^{(k+1)h} L(q(t), \dot{q}(t)) dt.$$

The discrete principle then requires that

$$\delta \sum_{k=0}^{N-1} L_d(q_k, q_{k+1}) = 0,$$

where variations are taken with respect to each position  $q_k$  along the path, and the resulting equations of motion become

$$D_2 L_d(q_{k-1}, q_k) + D_1 L_d(q_k, q_{k+1}) = 0.$$

*Example:* For example, consider a Lagrangian of the form  $L(q, \dot{q}) = \frac{1}{2} \dot{q}^T M \dot{q} - V(q)$  and define the discrete Lagrangian  $L_d(q_k, q_{k-1}) = hL\left(q_{k+\frac{1}{2}}, (q_{k+1} - q_k)/h\right)$ , using the notation  $q_{k+\frac{1}{2}} := (q_k + q_{k+1})/2$ . The resulting equations are

$$M \frac{q_{k+1} - 2q_k + q_{k-1}}{h^2} = -\frac{1}{2} (\nabla V(q_{k-\frac{1}{2}}) + \nabla V(q_{k+\frac{1}{2}})),$$

which is a discrete analog of Newton's law  $M\ddot{q} = -\nabla V(q)$ . For controlled (i.e., non conservative) systems, forces can be added using a discrete version of Lagrange-d'Alembert principle and discrete virtual work in a similar manner.

### B. Lie Group Integrators

Lie group integrators preserve symmetry and group structure for systems with motion invariants. Consider a system on configuration manifold  $Q = G \times M$  where  $G$  is a Lie group (with Lie algebra  $\mathfrak{g}$ ) whose action leaves the system invariant, i.e., it preserves the induced momentum map. For example,  $G = SE(3)$  can represent the group of rigid body motions of a free-floating articulated body while  $M$  is a space of internal variables describing the joints of the body. The idea is to transform the system equations from the original state space  $TQ$  into equations on the *reduced* space  $\mathfrak{g} \times TM$  (elements of  $TG$  are translated to the origin and expressed in the algebra  $\mathfrak{g}$ ) which is a linear space where standard integration methods can be used. The inverse of this transformation is then used to map curves in the algebra variables back to the group. Two standard maps have been commonly used to achieve this transformation for any Lie group  $G$ :

- Exponential map  $\exp : \mathfrak{g} \rightarrow G$ , defined by  $\exp(\xi) = \gamma(1)$ , with  $\gamma : \mathbb{R} \rightarrow G$  is the integral curve through the

identity of the left invariant vector field associated with  $\xi \in \mathfrak{g}$  (hence, with  $\dot{\gamma}(0) = \xi$ );

- Canonical coordinates of the second kind  $\text{ccsk} : \mathfrak{g} \rightarrow G$ ,  $\text{ccsk}(\xi) = \exp(\xi^1 e_1) \cdot \exp(\xi^2 e_2) \cdot \dots \cdot \exp(\xi^n e_n)$ , where  $\{e_i\}$  is the Lie algebra basis.

A third choice, valid only for certain *quadratic* matrix groups [22] (which include the rigid motion groups  $SO(3)$ ,  $SE(2)$ , and  $SE(3)$ ), is the Cayley map  $\text{cay} : \mathfrak{g} \rightarrow G$ ,  $\text{cay}(\xi) = (e - \xi/2)^{-1}(e + \xi/2)$ . Although this last map provides only an approximation to the integral curve defined by  $\exp$ , we include it as one of our choices since it is very easy to compute and thus results in a more efficient implementation. Other approaches are also possible, e.g., using retraction and other commutator-free methods; we will however limit our exposition to the three aforementioned maps in the formulation of the discrete reduced principle presented in the next section.

### C. Unified View

The optimal control algorithms in this paper are based on a discrete version of the Lagrange-d'Alembert-Pontryagin (LDAP) principle [16]. The LDAP viewpoint unifies the Lagrangian and Hamiltonian descriptions of mechanics [15] and extends to systems with symmetries and constraints. The discrete version of this principle yields integration schemes that generalize both the variational and Lie group integrators mentioned above.

*The Lagrange-d'Alembert-Pontryagin Principle:* We briefly recall the general formulation of the continuous LDAP principle for a system with Lagrangian  $L : TQ \rightarrow \mathbb{R}$  and control force<sup>1</sup>  $f : [0, T] \rightarrow T^*Q$ . For a curve  $(q(t), v(t), p(t))$  in  $TQ \oplus T^*Q$ ,  $t \in [0, T]$  the principle states that

$$\delta \int_0^T \{L(q, v) + p \cdot (\dot{q} - v)\} dt + \int_0^T f(t) \cdot \delta q(t) dt = 0, \quad (2)$$

for variations that vanish at the endpoints. The curve  $v(t)$  describes the velocity determined from the dynamics of the system. In view of the formulation,  $v$  does not necessarily correspond to the rate of change of the configuration  $q$ . The additional variable  $p$ , though, indirectly enforces this dependence and corresponds to both Lagrange multipliers and the momenta of the system. Thus 2 generalizes the Lagrange-d'Alembert principle and is linked to the Pontryagin maximum principle of optimal control.

The LDAP principle is conceptually equivalent to the Lagrange-d'Alembert principle. Nevertheless, in the discrete setting, the LDAP principle provides a more powerful framework for designing mechanical integrators. One notable benefit lies in the ability to derive higher-order integrators and, in the case of systems with symmetries, to tailor the algorithm

<sup>1</sup>In the Lagrangian setting a force is an element of the cotangent bundle  $T^*Q$ , i.e. a one-form  $\langle f, \cdot \rangle$  that pairs with velocity vectors to produce the total work  $\int_0^T \langle f, \dot{q} \rangle dt$  done by the force along a path between  $q(0)$  and  $q(T)$ .

structure to achieve a desired accuracy or efficiency [15]. While in this paper we do not explore higher order approximations and, in essence, our formulation in Sec. III-B could be alternatively derived using the discrete Euler-Poincaré (DEP) approach [23], we follow the LDAP formulation because of its greater flexibility. Another benefit appears in the discretization of systems with nonholonomic constraints. In particular, the optimal control method proposed in this paper is extended to nonholonomic systems of Chaplygin type in [17] (with the general case soon to follow) in a unified variational formulation.

## III. SYSTEMS WITH SYMMETRIES

In this section we develop the optimal control formulation for mechanical systems with symmetries. Assume that the configuration space is an  $n$ -dimensional Lie group  $G$  with algebra  $\mathfrak{g}$  and Lagrangian  $L : TG \rightarrow \mathbb{R}$  that is left invariant under the action of  $G$ . Using the invariance we can reduce such systems by introducing the *body-fixed* velocity  $\xi \in \mathfrak{g}$  defined by translation to the origin  $\xi = TL_{g^{-1}}\dot{g}$  and the reduced Lagrangian  $\ell : TG/G \rightarrow \mathbb{R}$  such that  $\ell(\xi) = L(g^{-1}g, g^{-1}\dot{g}) = L(e, \xi)$ . The system is controlled using a control vector  $u : [0, T] \rightarrow \mathbb{U}$ , where  $\mathbb{U} \subset \mathbb{R}^c$ ,  $c \leq n$ , is the set of controls applied with respect to a body-fixed basis  $\{F^1, \dots, F^c\}$ ,  $F^i : [0, T] \rightarrow \mathfrak{g}^*$ .

### A. The Continuous System

The continuous equations of motion are derived from the reduced Hamilton (or Lagrange-d'Alembert in the presence of forces) principle [4, 5] and have the standard form

$$\dot{g} = g\xi, \quad (3)$$

$$\mu = \ell'(\xi), \quad (4)$$

$$\dot{\mu} = \text{ad}_\xi^* \mu + u_i F^i. \quad (5)$$

Eq. (5) are the forced Euler-Poincaré equations, with  $\mu \in \mathfrak{g}^*$  denoting the system momentum, and (3) are the reconstruction equations. Note that  $\text{ad}_\xi^* \mu$  is defined by  $\langle \text{ad}_\xi^* \mu, \eta \rangle = \langle \mu, \text{ad}_\xi \eta \rangle$ , where  $\text{ad}_\xi \eta = [\xi, \eta]$  for  $\eta \in \mathfrak{g}$ .

#### Example: A Simple Helicopter

Consider the following simplistic model of a helicopter-like vehicle (Fig. 1). The vehicle is modeled as a single underactuated rigid body with mass  $m$  and principal moments of inertia  $I_1, I_2, I_3$  (the inertia matrix is denoted  $\mathbb{I} = \text{diag}(I_1, I_2, I_3)$ ). The vehicle is controlled through a *collective*  $u_c$  (lift produced by the main rotor) and a *yaw*  $u_\psi$  (force produced by the rear rotor), while the direction of the lift is controlled by tilting the main blades forward or backward through a *pitch*  $\alpha_p$  and a sideways *roll*  $\alpha_r$ . The configuration space is  $Q = SO(3) \times \mathbb{R}^3$  with  $(R, p) \in Q$  denoting orientation and position. Ignoring aerodynamic effects, we treat the rotational dynamics separately from the translational dynamics.

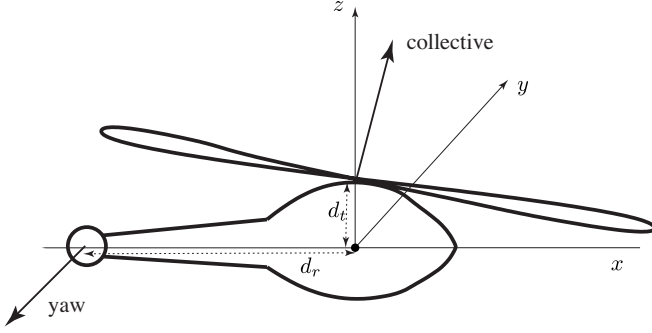


Fig. 1. Air vehicle model.

*Rotations:* Let  $\Omega \in \mathbb{R}^3 \sim \mathfrak{so}(3)$  denote the body angular velocity. The Lagrangian of the rotational dynamics is  $\ell(\Omega) = \frac{1}{2}\Omega^T \mathbb{I}\Omega$ . In the absence of forces the system is invariant under rotations and we can set  $G = SO(3)$  as the symmetry group. The torque control input basis can be written in matrix form as

$$F_\Omega(\alpha_p, \alpha_r) := \begin{bmatrix} d_t \sin \alpha_r & 0 \\ d_t \sin \alpha_p \cos \alpha_r & 0 \\ 0 & d_r \end{bmatrix},$$

where  $d_t$  and  $d_r$  are the distances from the top and rear blades to the center of mass of the helicopter. The reduced equations of the rotational dynamics corresponding to (3)-(5) are

$$\dot{R} = R\Omega, \quad (6)$$

$$\dot{\Pi} = \mathbb{I}\Omega, \quad (7)$$

$$\dot{\Pi} = \Pi \times \Omega + F_\Omega(\alpha)u, \quad (8)$$

where  $\Pi \in \mathbb{R}^3 \sim \mathfrak{so}(3)^*$  denotes angular momentum,  $\alpha = (\alpha_p, \alpha_r)$ , and  $u = (u_c, u_\psi)$ .

*Translations:* The translational dynamics is described by

$$m\ddot{p} = RF_V(\alpha)u + f^g,$$

where  $f^g = (0, 0, -9.8m)$  denotes gravity, and

$$F_V(\alpha_p, \alpha_r) := \begin{bmatrix} \sin \alpha_p \cos \alpha_r & 0 \\ -\sin \alpha_r & -1 \\ \cos \alpha_p \cos \alpha_r & 0 \end{bmatrix}.$$

Next we present the derivation of the corresponding discrete equations of motion and describe how to use them as constraints in the optimization problem.

### B. Discrete Reduced LDAP Principle

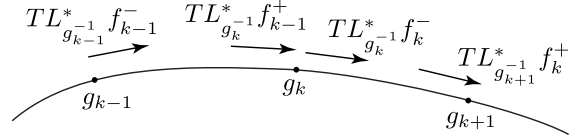
The *discrete* reduced Hamilton-Pontryagin principle for conservative systems was introduced in [15]. We now propose a simple extension to systems with internal forces. The principle is formulated using the *discrete reduced path* denoted by  $(g, \xi, \mu)_d : \{t_k\}_{k=0}^N \rightarrow G \times \mathfrak{g} \times \mathfrak{g}^*$ , where  $g_d = \{g_0, \dots, g_N\}$  and  $\xi_d, \mu_d$  analogously defined. The discrete control force which approximates a continuous control force is defined as

$f_d : \{t_k\}_{k=0}^N \rightarrow \mathfrak{g}^*$ . The map  $TL_{g(t)^{-1}}^* : \mathfrak{g}^* \rightarrow TG^*$  transforms the body-fixed momentum or force back to the cotangent space at point  $g(t)$ . The reduced LDAP principle results from expressing the principle (2) in terms of the reduced variables, i.e. by substituting  $L(g, \dot{g}) \Rightarrow \ell(\xi)$ ,  $v \Rightarrow TL_g \xi$ ,  $p \Rightarrow TL_{g^{-1}}^* \mu$  in (2). The discrete principle is then obtained by approximating the integrals in (2) according to:

$$\begin{aligned} & \delta \sum_{k=0}^{N-1} h [\ell(\xi_k) + \langle \mu_k, \tau^{-1}(g_k^{-1}g_{k+1})/h - \xi_k \rangle] \\ & + \sum_{k=0}^{N-1} [TL_{g_k}^* f_k^- \cdot \delta g_k + TL_{g_{k+1}}^* f_k^+ \cdot \delta g_{k+1}] = 0, \end{aligned} \quad (9)$$

where the map  $\tau : \mathfrak{g} \rightarrow G$  defines the *difference* between two configurations  $g_k$  and  $g_{k+1}$  on the group through an element in the Lie algebra.  $\tau$  is selected as a local diffeomorphism [15] such that  $\tau(\xi) \cdot \tau(-\xi) = e$ . The *left* (resp., *right*) discrete force  $f_k^- \in \mathfrak{g}^*$  (resp.,  $f_k^+ \in \mathfrak{g}^*$ , as shown below) is such that the work done by  $f$  along each discrete segment is approximated using the work done at the beginning (resp., the end) of the segment; that is, as described in [12], these two discrete forces provide an approximation of the continuous forcing through:

$$\begin{aligned} & \int_{kh}^{(k+1)h} TL_{g(t)^{-1}}^* f(t) \cdot \delta g(t) dt \\ & \approx TL_{g_k}^* f_k^- \cdot \delta g_k + TL_{g_{k+1}}^* f_k^+ \cdot \delta g_{k+1}. \end{aligned}$$



After taking variations in (9) we obtain the following discrete equations of motion (see Sec. 4.2 in [15] for details).

$$g_k^{-1}g_{k+1} = \tau(h\xi_k), \quad (10)$$

$$\mu_k = \ell'(\xi_k), \quad (11)$$

$$(d\tau_{h\xi_k}^{-1})^* \mu_k - (d\tau_{-h\xi_{k-1}}^{-1})^* \mu_{k-1} = f_{k-1}^+ + f_k^-, \quad (12)$$

where  $d\tau_\xi : \mathfrak{g} \rightarrow \mathfrak{g}$  is the *right-trivialized tangent* of  $\tau(\xi)$  defined by  $D\tau(\xi) \cdot \delta = TR_{\tau(\xi)}(d\tau_\xi \cdot \delta)$  and  $d\tau_\xi^{-1} : \mathfrak{g} \rightarrow \mathfrak{g}$  is its inverse<sup>2</sup>. Equations (10)-(12) can be understood as a discrete approximation of equations (3)-(5). They define a second-order accurate integrator that is part of the *variational Euler* family of methods [15].

The exact form of (12) depends on the choice of  $\tau$ . It is important to point out that this choice will influence the computational efficiency of the optimization framework when the equalities above are enforced as constraints. There are several choices commonly used for integration on Lie groups. We give details of two particularly representative examples: the exponential map (*exp*), and the Cayley map (*cay*)—see Sec. II-B for their definitions. Note that other maps, such as canonical coordinates of the second kind (*ccsk*) (also based on the exponential map), can be similarly derived.

<sup>2</sup>D is the standard derivative map (here taken in the direction  $\delta$ )

*Exponential map:* The right-trivialized derivative of the map  $\text{exp}$  and its inverse are defined as

$$\begin{aligned} \text{dexp}(x)y &= \sum_{j=0}^{\infty} \frac{1}{(j+1)!} \text{ad}_x^j y, \\ \text{dexp}^{-1}(x)y &= \sum_{j=0}^{\infty} \frac{B_j}{j!} \text{ad}_x^j y, \end{aligned} \quad (13)$$

where  $B_j$  are the Bernoulli numbers. Typically, these expressions are truncated in order to achieve a desired order of accuracy. The first few Bernoulli numbers are  $B_0 = 1$ ,  $B_1 = -1/2$ ,  $B_2 = 1/6$ ,  $B_3 = 0$  (see [22, 24] for details). Setting  $\tau = \text{exp}$ , (12) becomes

$$\text{dexp}^{-1}(h\xi_k)^* \mu_k - \text{dexp}^{-1}(-h\xi_{k-1})^* \mu_{k-1} = f_{k-1}^+ + f_k^-.$$

*Cayley map:* The derivative maps of  $\text{cay}$  (see Sec.IV.8.3 in [24] for derivation) are

$$\begin{aligned} \text{dcay}(x)y &= \left(e - \frac{x}{2}\right)^{-1} y \left(e + \frac{x}{2}\right)^{-1}, \\ \text{dcay}^{-1}(x)y &= \left(e - \frac{x}{2}\right) y \left(e + \frac{x}{2}\right). \end{aligned} \quad (14)$$

Using  $\tau = \text{cay}$  (see also [15]) (12) simplifies to

$$\begin{aligned} \mu_k - \mu_{k-1} - \frac{h}{2} \left( \text{ad}_{\xi_k}^* \mu_k + \text{ad}_{\xi_{k-1}}^* \mu_{k-1} \right) \\ - \frac{h^2}{4} \left( \xi_k^* \mu_k \xi_k^* - \xi_{k-1}^* \mu_{k-1} \xi_{k-1}^* \right) = f_{k-1}^+ + f_k^-. \end{aligned} \quad (15)$$

The Cayley map provides a coarser approximation than the exponential map  $\text{exp}$ , but its simple form is suitable for efficient implementation.

*Discrete Forces:* There are various ways to construct valid discrete forces  $f^+$  and  $f^-$ . A simple approach, in the spirit of the midpoint rule, is to assume that the left and right discrete forces at each segment are equal, and defined as the average of the forces applied in the beginning and the end of the segment:

$$f_k^- = f_k^+ = \frac{h}{2} \left[ \frac{f_k + f_{k+1}}{2} \right].$$

*Example:* Assume that we use the Cayley map and the midpoint rule to construct a variational integrator for the air vehicle model defined in Sec. III-A. The discrete equations of rotational motion (corresponding to eqs. (10)-(12), and making use of (15)) become

$$\begin{aligned} R_k^T R_{k+1} &= \text{cay}(h\widehat{\Omega}_k), \\ \Pi_k &= \mathbb{I} \Omega_k, \\ \Pi_k &= \Pi_{k-1} + \frac{h}{2} (\Pi_{k-1} \times \Omega_{k-1} + \Pi_k \times \Omega_k) \\ &+ \frac{h^2}{4} ((\Omega_{k-1}^T \Pi_{k-1}) \Omega_{k-1} - (\Omega_k^T \Pi_k) \Omega_k) \\ &+ \frac{h}{4} (F_{\Omega}(\alpha_{k-1}) u_{k-1} + 2F_{\Omega}(\alpha_k) u_k + F_{\Omega}(\alpha_{k+1}) u_{k+1}), \end{aligned} \quad (16)$$

where the map  $\widehat{\cdot}: \mathbb{R}^3 \rightarrow \mathfrak{so}(3)$  is defined by

$$\widehat{\Omega} = \begin{bmatrix} 0 & -\Omega^3 & \Omega^2 \\ \Omega^3 & 0 & -\Omega^1 \\ -\Omega^2 & \Omega^1 & 0 \end{bmatrix}.$$

The translational motion is derived using a standard variational integrator (see the example in Sec. II-A) and, with the addition of forces, becomes

$$\begin{aligned} m \left[ \frac{p_{k+1} - 2p_k + p_{k-1}}{h^2} \right] &= \frac{1}{4} [R_{k-1} F_V(\alpha_{k-1}) u_{k-1} \\ &+ 2R_k F_V(\alpha_k) u_k + R_{k+1} F_V(\alpha_{k+1}) u_{k+1}] + f_g. \end{aligned} \quad (17)$$

#### IV. DIRECT OPTIMAL CONTROL FORMULATION

A straightforward way to find a numerical solution to the optimal control problem is to formulate a nonlinear program that minimizes the cost function over all discrete configurations, velocities, and forces, while satisfying the boundary conditions and the discrete equations of motion enforced as equality constraints. Additional equality or inequality constraints can also be enforced.

##### A. Problem Formulation

The optimal control problem can be directly formulated as

$$\begin{aligned} \text{Compute: } & g_d, \xi_d, f_d, h \\ \text{minimizing } & \sum_{k=0}^{N-1} C_d(g_k, \xi_k, f_k^{\pm}, h) \\ \text{subject to: } & \begin{cases} g_0 = g_i, \xi_0 = \xi_i, g_N = g_f, \xi_{N-1} = \xi_f, \\ \text{Equations (10) - (12) for } k = 0, \dots, N-1, \\ H(g_d, \xi_d, h) \geq 0, \\ \xi_k \in [\xi_l, \xi_u], f_k \in [f_l, f_u], h \in [h_l, h_u], \end{cases} \end{aligned} \quad (18)$$

where  $C_d$  is a discrete approximation of  $C$  defined in (1) and  $H: G \times \mathfrak{g} \times \mathbb{R} \rightarrow \mathbb{R}^p$  are inequality constraints. The formulation allows time to vary and the last constraint places bounds on the time variation as well as bounds on all other variables.

##### Remarks

*Average Velocity:* The variables denoted  $\xi_N$  and  $\mu_N$  have no effect on the trajectory  $g_d$  so we can treat these last points as irrelevant to the optimization. This is coherent with thinking of each velocity  $\xi_k$  as the average body-fixed velocity along the  $k^{\text{th}}$  path segment between configurations  $g_k$  and  $g_{k+1}$ .

*Velocity at the boundary:* A second remark must be made regarding velocity boundary conditions. For simplicity, we work with the boundary conditions  $\xi_0 = \xi_i$  and  $\xi_{N-1} = \xi_f$  which are not exact, since according to the above assumption  $\xi_k$  represents an average velocity. A proper treatment of the exact velocity boundary conditions given by  $\xi(0)$  and  $\xi(T)$  requires further constraints such as

$$\begin{aligned} (d\tau_{h/2\xi(0)}^{-1})^* \ell'(\xi(0)) - (d\tau_{-h/2\xi_0}^{-1})^* \mu_0 &= f_0^-, \\ (d\tau_{h/2\xi_{N-1}}^{-1})^* \mu_{N-1} - (d\tau_{-h/2\xi(T)}^{-1})^* \ell'(\xi(T)) &= f_{N-1}^+. \end{aligned}$$

However, for simplicity and computational efficient, we assume that the boundary condition is in terms of the initial and final *average* velocity.

## B. Algorithm Construction

*Midpoint Rule:* The discrete cost function can be constructed using the midpoint rule as

$$C_d(g_k, \xi_k, f_k^\pm, h) = hC \left( g_{k+\frac{1}{2}}, \xi_k, \frac{f_k + f_{k+1}}{2} \right) \quad (19)$$

where  $g_{k+\frac{1}{2}} = g_k \tau(\frac{h}{2} \xi_k)$ , i.e. the midpoint along the flow defined by  $\tau$ . There are other choices besides the midpoint rule that can lead to integrators of arbitrary high order of accuracy, e.g., composition methods and symplectic Runge-Kutta methods [12]. The midpoint rule is a particularly pertinent choice for optimization problems since it provides a good balance between accuracy and efficiency.

*Implementation:* The optimal control formulation (18) can be solved using a standard constrained optimization technique such as sequential quadratic programming (SQP). A continuous control curve  $f$  can be obtained from a discrete solution curve  $f_d$  using linear interpolation of  $f_d$  (in case of the midpoint rule) or some higher order interpolation consistent with the order of accuracy of the chosen discretization.

## V. APPLICATION

We have implemented our framework for solving general optimal control problems of the form (18). It is written in C++ and uses the sparse SQP solver SNOPT [25]. The system is used to compute a control-effort minimizing trajectory for the simulated helicopter between two zero-velocity states in an artificial canyon. We use the discrete system described in the example of Sec.III-B with equations of motion defined by (16) and (17) and a discrete cost function defined by (19). Fig. 2 shows a typical output of our system and the resulting trajectory and control curves are shown on Fig. 5.

*Controllability:* In order to establish the controllability of the system one can use the good symmetric products [5] of the two vectors obtained from the columns of the matrix

$$\begin{bmatrix} \mathbb{I}^{-1} F_\Omega(\alpha) \\ \frac{1}{m} F_V(\alpha) \end{bmatrix},$$

and show that the system is locally configuration controllable at zero velocity. Since the actuator inputs have bounds, one has to design an algorithm which allows time to vary in order to accommodate these limits. In our implementation, the time-step  $h$  (and hence the final time  $T$ ) is part of the optimization state vector and is allowed to vary within some prescribed bounds.

*Optimization Variables:* For efficiency the components of the matrices  $R_k$  are not part of the optimization state vector and the trajectory  $R_d$  is reconstructed from the trajectory  $\Omega_d$  internally during optimization. Alternatively, one could parametrize the rotations (e.g., using quaternions) and optimize over these additional coordinates as well. In our experience, both approaches perform similarly well.

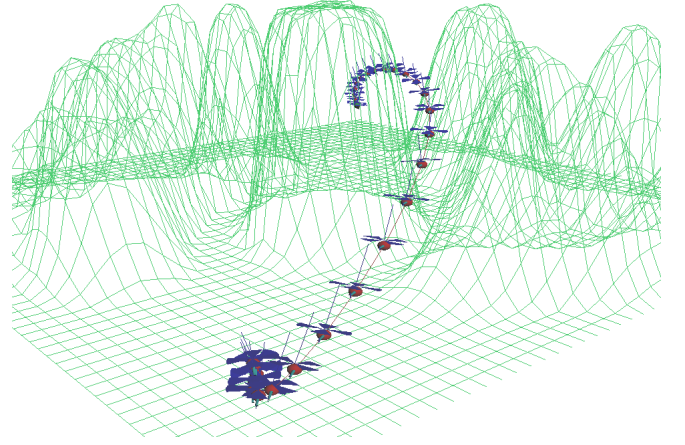


Fig. 2. Example of an optimized trajectory in a complex environment: a helicopter path through an outdoor canyon.

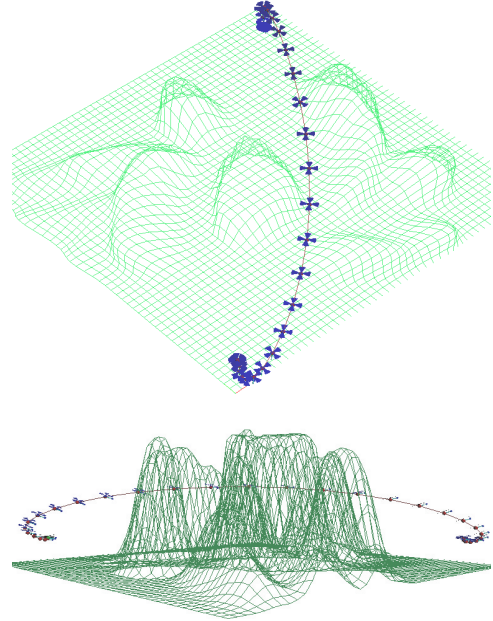


Fig. 3. Top and side views of the helicopter trajectory shown in Fig. 2

*Obstacles:* A robot can be required to stay away from obstacles by enforcing inequality constraints  $H_i(R, p) = \text{dist}(\mathcal{A}(R, p), \mathcal{O}_i) - d_s$ , where  $\mathcal{A} \subset \mathbb{R}^3$  is the region occupied by the robot,  $\mathcal{O}_i \subset \mathbb{R}^3$  represent the static obstacles, and  $d_s$  is some safety distance. The function  $\text{dist}$  computes the minimum distance between two rigid objects. In our implementation both the canyon and the helicopter are triangulated surfaces and we use the Proximity Query Package (PQP) to compute  $\text{dist}$ .

The optimization runs efficiently in the absence of obstacles or with simple (smooth and convex) obstacles (taking in the order of a few seconds for most tests). On the other hand, complex rough terrains can slow down the system significantly.

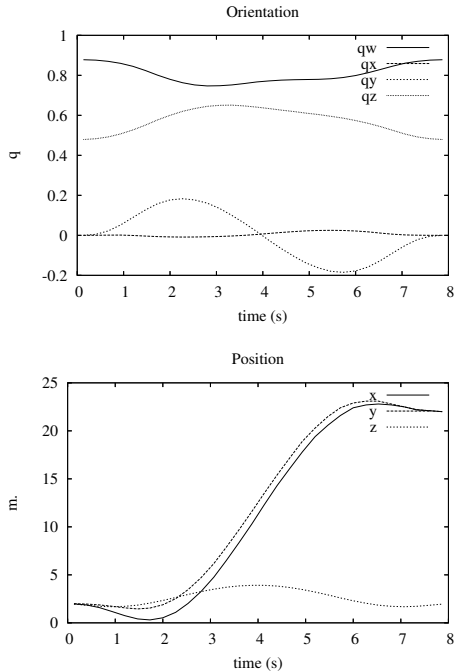


Fig. 4. The orientation and position as functions of time for the helicopter trajectory shown in Fig. 2.

Note also that our simple implementation faces the same robustness issues as many optimization procedures: a bad choice for the initial trajectory may lead to a local minima. One way to speedup convergence is to start with a good initial obstacle-free path computed, for example, using a local obstacle avoidance method (e.g. [26]). Nevertheless, as already pointed out by several similar works cited in our references, this approach should be used to produce small-scale local solutions that are combined by more *global* methods in a hierarchical fashion (e.g. see [8] regarding incremental and roadmap planners, as well as [27, 28] for planning using primitives). An obvious application is also the refinement of trajectories produced by discrete or sampling-based planners. Finally, in order to assess the exact numerical benefits of this method, a detailed performance analysis and comparison to related methods is needed and is currently under way.

## VI. CONCLUSION

There are many ways to solve optimal control problems, as confirmed by the rich literature on this subject. We have presented one approach that focuses on a proper discretization of the system dynamics through the extremization of a discrete action. An optimal control framework for systems with symmetries is then derived and applied to a simple robotic model. It will be useful to further optimize our approach using ideas from, e.g., [6, 7, 29]. A comparison with the closely related method [11] is also an obvious research direction. Since the optimization algorithm is inherently local (as are most gradient

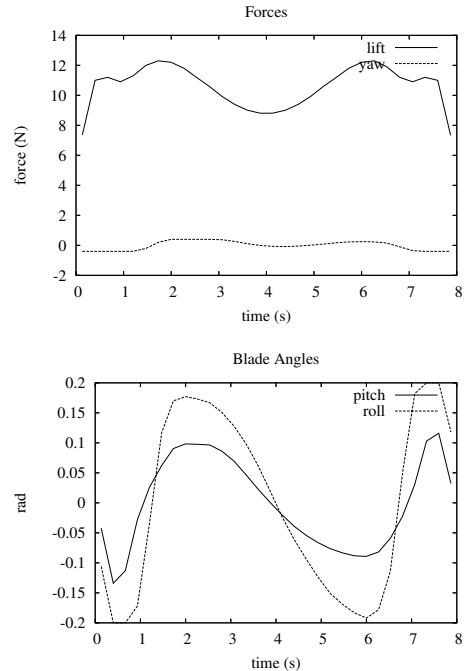


Fig. 5. The control forces and blade angles as functions of time for the helicopter trajectory shown in Fig. 2.

based methods) our method will be most effective in a global hierarchical optimization framework. This is a central issue related to the nature of discrete mechanics. In that respect, discrete mechanics can be used to perform provably correct coarse-to-fine discretization of time and space which is linked to global convergence of trajectories.

## Future Directions and Related Work

There are several promising directions of potential benefit to robotics, in which the discrete geometric approach is currently being extended. One of these is to further exploit the simplicity of the approach as well as its hierarchical benefits. Another is the discretization and optimal control of nonholonomic systems. Initial results with applications to Chaplygin systems and car-like robots can be found in [17]. Yet another direction is the inclusion of environment constraints such as obstacles or interaction constraints in groups of vehicles directly as part of the discrete optimization framework. Coarse-to-fine methods can then be used to jointly deform such constraints along with solution trajectories to produce more efficient methods with provable convergence properties. For example, multi-resolution extensions of the optimization methods presented in this paper already exhibit real-time performance when applied to rigid body systems. Finally, hierarchical and decentralized (e.g. [19]) DMOC methods are under development to address multi-vehicle or other complex systems in a scalable and robust manner.

As has been indicated, there are several important issues that need additional attention. One is a detailed comparison



to other optimization, both in terms of accuracy and in terms of hierarchical and coarse-to-fine benefits. A preliminary and encouraging study was done in [21], but this needs systematic and detailed analysis.

#### ACKNOWLEDGMENT

We are grateful to Eva Kanso, Nawaf Bou-Rabee, Sina Ober-Blöbaum, and Sigrid Leyendecker for their interest and helpful comments. This work was supported in part by NSF (CCR-0120778, CCR-0503786, IIS-0133947 and ITR DMS-0453145), DOE (DE-FG02-04ER25657), the Caltech Center for Mathematics of Information and AFOSR Contract FA9550-05-1-0343.

#### REFERENCES

- [1] S. Kelly and R. Murray, "Geometric phases and robotic locomotion," *Journal of Robotic Systems*, vol. 12, no. 6, pp. 417–431, 1995.
- [2] J. Ostrowski, "Computing reduced equations for robotic systems with constraints and symmetries," *IEEE Transactions on Robotics and Automation*, pp. 111–123, 1999.
- [3] E. A. Shamma, H. Choset, and A. A. Rizzi, "Motion planning for dynamic variable inertia mechanical systems with non-holonomic constraints," in *International Workshop on the Algorithmic Foundations of Robotics*, 2006.
- [4] J. E. Marsden and T. S. Ratiu, *Introduction to Mechanics and Symmetry*. Springer, 1999.
- [5] F. Bullo and A. Lewis, *Geometric Control of Mechanical Systems*. Springer, 2004.
- [6] J. P. Ostrowski, J. P. Desai, and V. Kumar, "Optimal gait selection for nonholonomic locomotion systems," *The International Journal of Robotics Research*, vol. 19, no. 3, pp. 225–237, 2000.
- [7] J. Cortes, S. Martinez, J. P. Ostrowski, and K. A. McIsaac, "Optimal gaits for dynamic robotic locomotion," *The International Journal of Robotics Research*, vol. 20, no. 9, pp. 707–728, 2001.
- [8] S. M. LaValle, *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006.
- [9] O. Junge, J. Marsden, and S. Ober-Blöbaum, "Discrete mechanics and optimal control," in *Proceedings of the 16th IFAC World Congress*, 2005.
- [10] E. Kanso and J. Marsden, "Optimal motion of an articulated body in a perfect fluid," in *IEEE Conference on Decision and Control*, 2005, pp. 2511–2516.
- [11] T. Lee, N. McClamroch, and M. Leok, "Optimal control of a rigid body using geometrically exact computations on SE(3)," in *Proc. IEEE Conf. on Decision and Control*, 2006.
- [12] J. Marsden and M. West, "Discrete mechanics and variational integrators," *Acta Numerica*, vol. 10, pp. 357–514, 2001.
- [13] J. P. Desai and V. Kumar, "Motion planning for cooperating mobile manipulators," *Journal of Robotic Systems*, vol. 16, no. 10, pp. 557–579, 1999.
- [14] L. Kharevych, Weiwei, Y. Tong, E. Kanso, J.E. Marsden, P. Schroder, and M.Desbrun, "Geometric, variational integrators for computer animation," in *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, 2006, pp. 43–51.
- [15] N. Bou-Rabee and J. E. Marsden, "Reduced Hamilton-Pontryagin variational integrators," preprint.
- [16] H. Yoshimura and J. Marsden, "Dirac structures in Lagrangian mechanics part ii: Variational structures," *Journal of Geometry and Physics*, vol. 57, pp. 209–250, dec 2006.
- [17] M. Kobilarov, M. Desbrun, J. Marsden, and G. S. Sukhatme, "Optimal control using d'alembert-pontryagin nonholonomic integrators," Center for Robotics and Embedded Systems, University of Southern California, Tech. Rep., 2007.
- [18] R. Fetecau, J. Marsden, M. Ortiz, and M. West, "Nonsmooth Lagrangian mechanics and variational collision integrators," *SIAM Journal on Applied Dynamical Systems*, vol. 2, no. 3, pp. 381–416, 2003.
- [19] O. Junge, J. Marsden, and S. Ober-Blöbaum, "Optimal reconfiguration of formation flying spacecraft - a decentralized approach," in *45th IEEE Conference on Decision and Control*, 2006, pp. 5210–5215.
- [20] S. Ross, "Optimal flapping strokes for self-propulsion in a perfect fluid," in *American Control Conference*, 2006.
- [21] M. Kobilarov and G. S. Sukhatme, "Optimal control using nonholonomic integrators," in *IEEE International Conference on Robotics and Automation*, Apr 2007, pp. 1832–1837.
- [22] E. Celledoni and B. Owren, "Lie group methods for rigid body dynamics and time integration on manifolds," *Comput. meth. in Appl. Mech. and Eng.*, vol. 19, no. 3,4, pp. 421–438, 2003.
- [23] J.E. Marsden and S. Pekarsky and S. Shkoller, "Discrete Euler-Poincaré and Lie-Poisson equations," *Nonlinearity*, vol. 12, p. 1647–1662, 1999.
- [24] E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration*, ser. Springer Series in Computational Mathematics. Springer-Verlag, 2006, no. 31.
- [25] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM J. on Optimization*, vol. 12, no. 4, pp. 979–1006, 2002.
- [26] D. E. Chang, S. Shadden, J. E. Marsden, and R. Olfati-Saber, "Collision avoidance for multiple agent systems," in *IEEE Conference on Decision and Control*, vol. 42, 2003, pp. 539–543.
- [27] E. Frazzoli, M. A. Dahleh, and E. Feron, "Maneuver-based motion planning for nonlinear systems with symmetries," *IEEE Transactions on Robotics*, vol. 21, no. 6, pp. 1077–1091, dec 2005.
- [28] C. Dever, B. Mettler, E. Feron, and J. Popovic, "Nonlinear trajectory generation for autonomous vehicles via parameterized maneuver classes," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 2, pp. 289–302, 2006.
- [29] M. B. Milam, K. Mushambi, and R. M. Murray, "A new computational approach to real-time trajectory generation for constrained mechanical systems," in *IEEE Conference on Decision and Control*, vol. 1, 2000, pp. 845–851.