Geometric, Variational Integrators for Computer Animation

L. Kharevych  Weiwei  Y. Tong  E. Kanso†  J. E. Marsden  P. Schröder  M. Desbrun
Caltech - USC

Abstract
We present a general-purpose numerical scheme for time integration of Lagrangian dynamical systems—an
important computational tool at the core of most physics-based animation techniques. Several features make this
particular time integrator highly desirable for computer animation: it numerically preserves important invariants,
such as linear and angular momenta; the symplectic nature of the integrator also guarantees a correct energy
behavior, even when dissipation and external forces are added; holonomic constraints can also be enforced quite
simply; finally, our simple methodology allows for the design of high-order accurate schemes if needed. Two key
properties set the method apart from earlier approaches. First, the nonlinear equations that must be solved during
an update step are replaced by a minimization of a novel functional, speeding up time stepping by more than a
factor of two in practice. Second, the formulation introduces additional variables that provide key flexibility in the
implementation of the method. These properties are achieved using a discrete form of a general variational prin-
ciple called the Pontryagin-Hamilton principle, expressing time integration in a geometric manner. We demonstrate
the applicability of our integrators to the simulation of non-linear elasticity with implementation details.

1. Introduction
Mathematical models of the evolution in time of dynamical systems (whether in biology, economics, or computer ani-
mation) generally involve systems of differential equations. Solving a physical system means figuring out how to move
the system forward in time from a set of initial conditions, allowing the computation of, for instance, the trajectory of
a ball (i.e., its position as a function of time) thrown up in the air. Although this example can easily be solved anal-
detically, direct solutions of the differential equations govern-
ing a system are generally hard or impossible—we need
to resort to numerical techniques to find a discrete tempo-
ral description of a motion. Consequently, there has been
a significant amount of research in applied mathematics on
how to deal with some of the most useful systems of equa-
tions, leading to a plethora of numerical schemes with vari-
ous properties, orders of accuracy, and levels of complexity
of implementation [PFTV92]. In Computer Animation, these
integrators are crucial computational tools at the core of
most physics-based animation techniques, and classical
methods (such as fourth-order Runge-Kutta, implicit Euler,
and more recently the Newmark scheme) have been meth-
ods of choice in practice [Par01]. Surprisingly, developing
better (i.e., faster and/or more reliable) integrators received
very little attention in our community, even if the few papers
dedicated to this goal showed encouraging results [HES03].

In this paper, we follow a geometric—and instead of a tradi-
tional numerical-analytic—approach to the problem of time
integration. Motivated by the success of discrete variational
approaches in geometric modeling and discrete differential
geometry, we will consider mechanics from a variational
point of view. The very essence of a mechanical system is
indeed characterized by its symmetries and invariants (e.g.,
momenta), thus preserving these geometric notions into the
discrete computational setting is of paramount importance if
one wants discrete time integration to properly capture the
underlying continuous motion. Consequently, we advocate
the use of discrete variational principles as a way to derive
simple, robust, and accurate time integrators. In particular,
we derive a novel, simple geometric integrator based on the
very general Hamilton-Pontryagin principle.

1.1. Background
Dynamics as a Variational Problem Considering mechan-
ics from a variational point of view goes back to Euler, La-
grange and Hamilton. The form of the variational principle
most important for continuous mechanics is due to Hamil-
ton, and is often called Hamilton’s principle or the least ac-
tion principle (as we will see later, this is a bit of a mis-
nomer: “stationary action principle” would be more correct):
it states that a dynamical system always finds an optimal
course from one position to another (a more formal defini-
tion will be presented in Section 2). One consequence is that
we can recast the traditional way of thinking about an object
accelerating in response to applied forces, into a geometric
viewpoint. There the path followed by the object has opti-
mal geometric properties—alike to the notion of geodesics
on curved surfaces. This point of view is equivalent to New-
ton’s laws in the context of classical mechanics, but is broad
enough to encompass areas ranging to E&M and quantum
mechanics.

Geometric Integrators are a class of numerical time-
stepping methods that exploit the geometric structure of me-
chanical systems [HLW02]. Of particular interest within this
class, variational integrators [MW01] discretize the varia-
tional formulation of mechanics we mentioned above, providing a solution for most ordinary and partial differential equations that arise in mechanics. While the idea of discretizing variational formulations of mechanics is standard for elliptic problems using Galerkin Finite Element methods for instance, only recently did it get used to derive variational time-stepping algorithms for mechanical systems. This approach allows the construction of integrators with any order of accuracy [Wes03, Lew03], that can handle constraints as well as external forcing. These integrators have been shown remarkably powerful for simulations of physical phenomena when compared to traditional numerical time stepping methods [KMOW00]. This discrete-geometric framework is thus versatile, powerful, and general. For example, the well-known symplectic variant of the Newmark scheme (velocity Verlet) can best be elucidated by writing it as a variational integrator [Wes03]. Of particular interest in computer animation, the simplest variational integrator can be implemented by taking two consecutive positions \( q_0 = q(t_0) \) and \( q_1 = q(t_0 + \Delta t) \) of the system to compute the next position \( q_2 \). Repeating this process calculates an entire discrete (in time) trajectory.

**Accurate vs. Qualitative Integrators** While it is unavoidable to make approximations in numerical algorithms (i.e., to differ from the continuous equivalent), the matter becomes whether the numerics can provide satisfactory results. Qualitative reproduction of phenomena is often favored in computer animation over absolute accuracy. We argue in the following that one does not have to ask for either plausibility or accuracy. In fact, we seek a simple method robust enough to provide good, qualitative simulations that can also be easily rendered arbitrarily accurate. The simplistic character of variational integrators provides good foundations for the design of robust algorithms: this property guarantees good statistical predictability through accurate preservation of the geometric properties of the exact flow of the differential equations. As a consequence, symplecticity offers long-time energy preservation—a crucial property since large energy increase is often synonymous with numerical divergence while a large decrease dampens the motion, decreasing visual plausibility. A well-known example where this property is crucial is the simple pendulum (particularly relevant in robotic applications for articulated figures), for which other (even high-order) integrators can fail in keeping the amplitude of the oscillations (see Figure 1). With this in mind, we will pursue numerical schemes which offer qualitatively-correct as well as arbitrarily accurate solutions.

1.2. Contributions

We address the problem of discrete time integration as a discrete geometric problem where the dynamics is obtained from a (stationary action) Hamilton-Pontryagin principle, i.e., as the stationary point of a discrete action. Using the Hamilton-Pontryagin principle provides conceptual and algorithmic simplicity even for dissipative systems and in the presence of constraints. Computationally, our novel approach is more efficient (an improvement of at least a factor of two) since we can replace the usual non-linear multi-dimensional root finding time stepping techniques by a simpler minimization procedure (generalizing the idea of “minimum principle” [RO99]). The resulting new family of variational symplectic integrators also inherits key numerical properties: guaranteed momenta preservation and correct energy behavior. We demonstrate the robustness, simplicity, and efficiency of our time integration schemes by applying them to nonlinear elasticity and additionally describe a simple dissipation model.

2. Overview of Continuous Lagrangian Dynamics

Before presenting our contributions, we first give a description of the continuous Lagrangian principles of dynamical systems as they relate to the development of the discrete Hamilton-Pontryagin principle.

Consider a finite-dimensional dynamical system parameterized by the state variable \( q \) (i.e., the vector containing all degrees of freedom). The Lagrangian function of the system is given as a function of \( q \) and \( \dot{q} \). In the more restrictive case of basic elasticity, this Lagrangian function \( L \) is defined as the kinetic energy \( K \) minus the potential energy \( W \) of the system:

\[
L(q, \dot{q}) = K(\dot{q}) - W(q).
\]

The action functional is the integral of \( L \) along a path \( q(t) \), over time \( t \in [0, T] \). Hamilton’s principle now states that the correct path of motion of a dynamical system is such that its action has a stationary value, i.e., the integral along the correct path has the same value to within first-order infinitesimal perturbations. As an “integral principle” this description encompasses the entire motion of a system between two fixed times.

Computing variations of the action induced by variations \( \delta q \) of the path \( q(t) \) results in:

\[
\delta S(q) = \delta \int_0^T L(q(t), \dot{q}(t)) \, dt = \int_0^T \left[ \frac{\partial L}{\partial q} \delta q + \frac{\partial L}{\partial \dot{q}} \delta \dot{q} \right] \, dt
\]

\[
= \int_0^T \left[ \frac{\partial L}{\partial q} - \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}} \right) \right] \delta q \, dt + \left[ \frac{\partial L}{\partial \dot{q}} \delta \dot{q} \right]_{t_0}^T.
\]

where integration by parts is used in the last equality. When the endpoints of \( q(t) \) are held fixed with respect to all variations \( \delta q(t) \) (i.e., \( \delta q(0) = \delta q(T) = 0 \)), the rightmost term in the above equation vanishes. Therefore, the condition of

© The Eurographics Association 2006.
stationary action for arbitrary variations \( \delta q \) with fixed endpoints stated in Hamilton’s principle directly indicates that the remaining integrand in the previous equation must be zero for all time \( t \), yielding the well-known Euler-Lagrange equations:

\[
\frac{\partial L}{\partial q} - \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}} \right) = 0.
\]

**(1)**

**Standard Example**  Let \( K = \frac{1}{2} q^T M q \), where \( M \) is the mass Matrix. Then (1) simply states Newton’s law: \( M \ddot{q} = -\nabla W(q) \), i.e., mass times accelerations equals force. Here, the force is conservative (no damping occurs) since it is derived from a potential function.

**Forced Systems**  To account for non-conservative forces \( F \) (typically, dissipation), the least action principle is modified:

\[
\delta \int_0^T \left[ L(q(t), \dot{q}(t)) + \int F(q,t,\dot{q}(t)) \cdot \delta q \ dt \right] dt = 0,
\]

which is known as the Lagrange-d’Alembert principle.

**Lagrangian vs. Hamiltonian Mechanics**  Lagrangian mechanics is not the only existing formalism available. In fact, Hamiltonian mechanics provides an alternative, closely related formulation. For later use we point out that Hamiltonian mechanics is described in phase space, i.e., the current state of a dynamical system is given as a pair \((q,p)\), where \( q \) is the state variable, while \( p \) is the momentum, defined as \( p = \partial L/\partial \dot{q} \).

**Discrete Lagrangian Mechanics**  The least action principle stated above can be used as a guiding principle to derive discrete integrators. In fact, West [Wes03] proposed a direct discretization of the integral of the Lagrangian to construct a proper and simple discrete action function. In this approach the integrals are replaced with quadrature rules, i.e., linear combinations of discrete evaluations of the Lagrangian, over each elementary time step. Time stepping is then realized by taking the variation of the discrete action between two positions \( q(t + dt) \) and \( q(t + 2dt) \) of a dynamical system. This class of approaches respects the variational nature of time evolution in the discrete realm. The resulting discrete Euler-Lagrange (DEL) equations provide the update rule to advance in time: given two consecutive (in time) states of the system, the next state (at the end of the current time step) can be computed through a non-linear solve of the DEL equations. For more details on the DEL equations, we refer the reader to an introductory text on discrete mechanics [SD06].

3. **Fully Variational Integrators**

We will now present a novel family of variational integrators based on a more general principle known as the Hamilton-Pontryagin principle (a.k.a. Livins’ principle). In this approach the velocity \( v \) is, a priori, an additional free variable. We will show how a discrete version of this principle will lead to integrators sharing the exact same numerical benefits as the best integrators known so far and allow us to express time-stepping as a simple minimization instead of a computationally more expensive multi-dimensional root finding problem.

3.1. **Continuous Hamilton-Pontryagin Principle**

The Hamilton-Pontryagin principle (deeply rooted in the control of dynamical systems) states that the equations of mechanics are given by the critical points of the Hamilton-Pontryagin action:

\[
\delta \int_0^T \left[ p(\dot{q} - v) + L(q,v) \right] dt = 0,
\]

where the configuration variable \( q \), the velocity \( v \) and the momentum \( p \) are all viewed as independent variables. (See [YM06] for an exposition and history.) That is, \( q(t), v(t), p(t) \) are varied independently (with end-point conditions on \( q(t) \)). Notice the similarity with Hamilton’s principle: \( p \) can be interpreted as a Lagrange multiplier to enforce the equality between \( \dot{q} \) and \( v \). The Hamilton-Pontryagin principle yields equations equivalent to the Euler-Lagrange equations (1), since, for the respective variations \( \delta p(t), \delta q(t) \) and \( \delta v(t) \) over the three independent variables, we get:

\[
v = \dot{q}, \quad \frac{dp}{dt} = \frac{\partial L(q,v)}{\partial q}, \quad p = \frac{\partial L(q,v)}{\partial v}. \]

(2)

We stress the important feature this different variational approach brings and that points to the generality of this principle: with the addition of the new variables, these equations can be understood from a Lagrangian and Hamiltonian point of view since the formulation involves both phase-space variables \( q \) and \( p \) within the action. A more thorough discussion on this connection to Hamiltonian mechanics can be found in [LW06].

3.2. **Set-Up and Discrete Formulation**

**Time Discretization** A motion \( q(t) \), for \( t \in [0,T] \), of the mesh is replaced by a discrete sequence of poses \( q_k \), with \( k = 0, \ldots, N \in \mathbb{N} \), at discrete times: \( \{ t_0 = 0, \ldots, t_{k-1}, t_k, t_{k+1}, \ldots, t_N = T \} \). We will call \( h_k \) the time step between time \( t_k \) and \( t_{k+1} \). Note that the time step can be adjusted throughout the computation based on standard time step control ideas if necessary. We similarly discretize \( v(t) \) and \( p(t) \) by the sets \( \{ v_k \}_{k=0}^N \) and \( \{ p_k \}_{k=0}^N \). Velocities \( v_k \) and momenta \( p_k \) are viewed as approximations within the interval \( [t_{k-1}, t_k] \), i.e., staggered with respect to the positions \( q_k \).

**Quadrature-based Discrete Action** We will remain agnostic as to the Lagrangian used in this section: the case of non-linear elasticity will be addressed in Section 5, but our explanations are valid for any continuous Lagrangian \( L(q,v) \). For a given choice of Lagrangian, one can easily derive a discrete action through quadrature. Computationally very attractive are one-point quadrature rules to turn the continuous action (i.e., the integral in time of the Lagrangian) into a discrete Lagrangian \( L^d(q_k,v_{k+1}) \) through:

\[
L^d(q_k,v_{k+1}) = L(q_k + \alpha h_k v_{k+1}, v_{k+1}) h_k \simeq \int_k^{k+1} L(q,v) dt.
\]

(3)

\( L^d \) is a time integral of the Lagrangian that we refer to as a discrete Lagrangian. This is not unlike the use of the term “discrete curvature” in CG which refers to a small, local integral of a continuous curvature. Notice that this quadrature has quadratic accuracy for \( \alpha = 1/2 \) and linear accuracy for...
all other $\alpha \in [0,1]$. More accurate quadrature rules (be they of Newton-Cotes or Gaussian type [PFTV92], for example) can be employed to increase the approximation order if necessary. Without loss of generality, we will solely use Eq. (3) in the remainder of this paper for simplicity.

3.3. Discrete Hamilton-Pontryagin Principle

Once a discrete Lagrangian is given, a discrete Hamilton-Pontryagin principle can be expressed through:

$$\delta \sum_{k=0}^{N} \left[ p_{k+1} \left( \frac{q_{k+1} - q_k}{h_k} - v_{k+1} \right) + L^d \left( q_k, v_{k+1} \right) \right] = 0.$$  

Discrete Variational Equations

The discrete Hamilton-Pontryagin principle yields, upon taking discrete variations with respect to each state variable with fixed endpoints:

$$\delta p : \quad q_{k+1} - q_k = h_k v_{k+1} \quad \text{(4)}$$
$$\delta v : \quad p_k - p_{k-1} = D_1 L^d (q_k, v_{k+1}) \quad \text{(5)}$$
$$\delta v : \quad h_k p_{k+1} - p_k = D_2 L^d (q_k, v_{k+1}) \quad \text{(6)}$$

where $D_1$ and $D_2$ denote the differentiation with respect to the first ($q_k$) and second ($v_{k+1}$) arguments of $L^d$.

Natural Update Procedure

Given a point in the discrete Pontryagin-state space $(q_k, v_k, p_k)$, the above equations are to be solved for $(q_{k+1}, v_{k+1}, p_{k+1})$ in the following way:

- Plug (5) into (6) so that $p_{k+1}$ is replaced by a function of $p_k$ and $D_1 L^d (q_k, v_{k+1})$.

- The resulting equation:

$$D_2 L^d (q_k, v_{k+1}) - h_k p_k - h_k D_1 L^d (q_k, v_{k+1}) = 0 \quad \text{(7)}$$

now can be solved for $v_{k+1}$ with any non-linear solver.

- $q_{k+1}$ and $p_k$ are found with (4) and (6) respectively.

Equivalence with DEL Equations

One can readily verify (using the chain rule) that the integration procedure (4-6) obtained from the discrete Hamilton-Pontryagin principle is mathematically equivalent to the variational integrator described in [Wes03]. Thus, both schemes share the same numerical benefits such as the conservation of discrete momenta and energy, as we will discuss further in Section 4.3.

3.4. Discrete Pontryagin-d’Alembert Principle

For non-conservative systems, the (continuous) Pontryagin-d’Alembert principle is given by:

$$\delta \int_0^T \left[ L(q, v) + p(\dot{q} - v) \right] dt + \int_0^T F_v(q, v) \cdot \delta q dt = 0$$

where $F_v(q,v)$ is an arbitrary (external) non-conservative force function. The discrete Pontryagin-d’Alembert principle can thus be defined as:

$$\delta \sum_{k=0}^{N} p_{k+1} (q_{k+1} - q_k - h_k v_{k+1}) + L^d (q_k, v_{k+1}) + \sum_{k=0}^{N} \left( F^d^-(q_k, v_{k+1}) \cdot \delta q_k + F^d^+(q_k, v_{k+1}) \cdot \delta v_{k+1} \right) = 0,$$

where $F^d^-$ and $F^d^+$ approximate the total forcing over a time step (see schematic figure below) through:

$$F^d^-(q_k, v_{k+1}) \delta q_k + F^d^+(q_k, v_{k+1}) \delta v_{k+1} \approx \int_{t_k}^{t_{k+1}} F(q, \dot{q}) \delta \dot{q} dt.$$
function” (the squared norm of the residual) to monitor the progress made towards reaching the zero [NW99]. Significant computational gain could thus be achieved by having a scalar function to minimize instead, with lower order and complexity than the merit function. In fact, this idea is very much responsible for the success of the well-known Conjugate Gradient method to solve a linear system like $Ax = b$. Its foundations come from a minimization technique applied to the function $f(x) = \frac{1}{2}x^T A x - b x$. If one were to use the residual $\|Ax - b\|^2$ instead, the “merit function” has a term in $x^T (A^T A) x$, resulting in a much worse condition number.

When non-linear equations are to be solved, the gain can be even greater. Thus, we propose in this section a more general derivation of variational integrators, and in particular, of our discrete Pontryagin-Hamilton integrator, for which the time stepping will be performed through a minimization.

### 4.2. Variational Update

The time integrator that is based on (4-6) can be replaced by a variational update procedure done via minimization of an energy-like function given that the dynamical system satisfies certain integrability conditions as discussed below. This technique extends an idea of Radovitzky and Ortiz [RO09], where Verlet’s integrator was shown to satisfy a minimum principle—a surprising fact given the extremum nature of Hamilton’s principle. Our construction extends this property to a whole family of arbitrarily high order schemes that we call fully-variational integrators as a variational principle is not only used for their derivation, but also for numerical computations.

### Variational Integrability Assumption

We consider the class of dynamical systems whose discrete Lagrangian $L^d$ has the property:

$$D_1 L^d(q_k,v_{k+1}) = D_2 P(q_k,v_{k+1})$$

(8)

for some function $P(q_k,v_{k+1})$. The property (8) will be referred to as the variational integrability property. One can view this property as a design criterion that some (exceptionally nice) variational integrators might have, and in fact this condition is strictly equivalent to another formulation given in Section 2.8 of [Lew03]. However, this particular property is not as restrictive as indicated in this reference: in fact, most current models used in Computer Animation satisfy it. Indeed, this property is valid for any quadrature-based discretization of a Lagrangian describing an arbitrary elastic model (we will provide a concrete example of discrete Lagrangian for non-linear elasticity in Section 5).

Thus, our assumption is general, and can directly be used to design higher-order accurate schemes (through higher order quadrature rules which map continuous integrals to discrete sums [MW01]) still satisfying this integrability criterion.

### Fully-Variational Update

Now, start again with the variational equations (4-6). Clearly, (6) can be rewritten as:

$$\frac{\partial}{\partial v_{k+1}}[-h_k p_{k+1} v_{k+1} + L^d(q_k,v_{k+1})] = -h_k p_{k+1} + D_2 L^d(q_k,v_{k+1}) = 0$$

We can substitute (5) in the above equation to get:

$$-h_k p_{k+1} - h_k D_1 L^d(q_k,v_{k+1}) + D_2 L^d(q_k,v_{k+1}) = 0$$

Thanks to the variational integrability property, this last equation can be rewritten as:

$$\frac{\partial}{\partial v_{k+1}}[-h_k p_{k+1} v_{k+1} - h_k P(q_k,v_{k+1}) + L^d(q_k,v_{k+1})] = 0.$$  

(9)

The quantity inside the bracket is an energy-like function of $q_k, p_k$ and $v_{k+1}$ and will be referred to hereafter as the Lilyan function $E$:

$$E(v_{k+1}) = -h_k p_{k+1} v_{k+1} - h_k P(q_k,v_{k+1}) + L^d(q_k,v_{k+1}).$$

(10)

The value of $v_{k+1}$ can then be found as a critical point of the Lilyan. We can now state the following result:

**Suppose that the variational integrability property (8) holds. Given the triplet $(q_k,p_k,v_k)$, we can find $v_{k+1}$ by minimizing the Lilyan defined by (10), while $q_{k+1}$ and $p_{k+1}$ are then explicitly computed using (4) and (5). The resulting triplet $(q_{k+1},p_{k+1},v_{k+1})$ satisfies (4), (5), and (6), giving us a fully variational integration scheme. In particular, this procedure defines a (symplectic) update map $(q_k,p_k) \mapsto (q_{k+1},p_{k+1})$.**

**Proof:** Of course (4) and (5) are satisfied by construction. We need to check that (6) holds when minimizing (10) with respect to $v_{k+1}$. However, this is a simple calculation:

$$h_k p_{k+1} = h_k p_k + h_k D_1 L^d(q_k,v_{k+1}) \quad \text{definition of } p_{k+1}$$

$$= h_k p_k + h_k D_2 P(q_k,v_{k+1}) \quad \text{eq. (8)}$$

$$= \frac{\partial}{\partial v_{k+1}}[h_k p_{k+1} v_{k+1} + h_k P(q_k,v_{k+1})] \quad \text{obvious}$$

$$= \frac{\partial}{\partial v_{k+1}}L^d(q_k,v_{k+1}) \quad \text{assumed eq. (9)}$$

$$= D_2 L^d(q_k,v_{k+1}),$$

which is the desired equation (6). The last statement of our claim holds because this update map is equivalent to the position momentum form of the DEL equation mentioned in [Wes03].

### Numerical Behavior of the Lilyan

A closer look shows that if $h_k$ is small, the Lilyan $E$ is quadratic in $v_{k+1}$: the terms depending on the potential energy are of order $h_k^2$, leaving only $p_k v_{k+1}$ and the kinetic energy as terms of order $h_k$—and those form a quadratic function of $v_{k+1}$. Thus, for small enough time steps, one can always find $v_{k+1}$ as the value that globally minimizes the Lilyan for the current values of $q_k$ and $p_k$.

### 4.3. Numerical Advantages of Fully-Variational Updates

#### Accuracy

Our particular choice of one point-quadrature for the discrete Lagrangian renders the accuracy of integration linear (for $\alpha \neq 1/2$) or quadratic (for $\alpha = 1/2$). Although this level of accuracy is enough for most applications in graphics, one can devise higher-order schemes by providing more accurate quadratures, at the price of a higher computational...
cost. As we will detail in Section 5, the scheme we introduced is also quite versatile, as the value \( \alpha = 0 \) provides a fully explicit integration, which is very efficient, still linear accurate and continues to preserve momenta. Note also that the step sizes \( h_k \) can be adjusted locally to control accuracy.

**Conservation Laws** A nice feature of our discrete variational framework is that the relationship between the symmetry and conserved quantities matches the continuous theory of mechanics. More precisely, the invariance of the (continuous) Lagrangian under a given set of transformations of its variables defines its symmetries. Clearly these leave the action integral invariant as well. Thus symmetries give rise to conserved quantities, as stated in Noether’s theorem. For example, the invariance of \( L(q(t), \dot{q}(t)) \) under translations and rotations results in the conservation of linear and angular momenta, respectively. One of the most attractive features of the variational integrators is that they conserve discrete quantities associated with discrete symmetries of the discrete Lagrangian [Lew03, Wes03]. We argued in Sections 3.2 and 4.2 that the variational scheme in (4-5) and (9) is mathematically equivalent to existing discrete Lagrangian-based integrators under certain integrability conditions and, hence, share the same numerical conservation properties (see Figure 2): momenta associated with symmetries of the Lagrangian are preserved exactly and automatically, for any order of accuracy. Note that the resulting update rules are not more complicated than standard integrators: we simply enforce conservation laws at no extra cost by a proper discretization of the geometric principle behind the dynamics.

**Energy** The symplectic nature of our scheme also guarantees a good energy behavior. For conservative systems, the integration shows a nice energy preservation as demonstrated in Figure 2. The proper treatment of forced systems handles energy dissipation gracefully as well (see Figure 3). Note that the energy dissipation in more traditional integrators is often a mix of user-prescribed damping and uncontrollable numerical viscosity (depending on the time step size). In sharp contrast, our algorithm allows a precise control of the amount of damping introduced in the simulation independent of the time step used for simulation—a particularly desirable property to better control the behavior of physics-based models such as cloths where adaptive timestepping is often necessary.

**A Word of Caution** The reader may be misled into thinking that our scheme does not require the typical Courant-Friedrichs-Levy (CFL) condition (or equivalent) on the time step size [PFTV92]. This is, of course, untrue: the same theoretical limitations in the explicit case \( (\alpha = 0) \) are still valid for our scheme. Other values of \( \alpha \)—leading to implicit schemes—do not share this particular limitation, generally allowing for much larger time steps. In the non-linear setting, time step sizes are often constrained by the non-linearity of the system. This too is no different in our setting, with the notable exception that numerical energy minimizers (applied to the Lifyan) are notably less sensitive to this constraint than multi-dimensional root finders.

### 5. Application to Non-Linear Elasticity

In this section we put our theory to work by applying it to the simulation of the motion of an elastic body under the influence of external forces.

#### 5.1. Set-Up

An elastic body \( B \) can undergo reversible deformations (changes in shape) due to applied forces. These may be body forces per unit volume or surface traction per unit area. Deformation typically depends on the material, size and geometry of the body as well as the applied forces. A motion is a one-parameter (time) family of deformations and can be described by \( x(X,t) \), where \( X \) denotes the position of a material particle of \( B \) in the reference configuration and \( t \) is time. That is, \( x \) is the particle position in the deformed or current configuration.

The kinetic energy of the body is given by:

\[
K = \frac{1}{2} \int_B \rho \mathbf{v} \cdot \mathbf{v} \, dV,
\]

where \( \rho \) is the mass density, \( \mathbf{v} \) is the velocity (function of material particle \( X \) and time \( t \)), and \( dV \) is a volume element. Further, in the pure mechanical theory of elasticity, there exists a strain (or stored) energy density function \( w \) per unit volume whose change represents the change in the internal energy due to mechanical deformations, which means the potential energy (excluding gravity) is written as

\[
W = \int_B w \, dV.
\]

The functional dependence of the internal energy \( w \) on the deformation is through the Cauchy strain \( C \), defined as:

\[
C = \left( \frac{\partial \mathbf{X}}{\partial X} \right)^T \left( \frac{\partial \mathbf{X}}{\partial X} \right).
\]

More specifically, \( w \) can only depend on the three invariants \( I_1, I_2 \) and \( I_3 \) of the tensor \( C \):

\[
I_1 = \text{tr}(C), \quad I_2 = \text{tr}(C^2) - \text{tr}(C)^2, \quad I_3 = \det(C).
\]

The function \( w(I_1, I_2, I_3) \) varies depending on the material type, for instance, for Mooney-Rivlin materials, \( w = a_1(I_1 - 3) + b_1(I_2 - 3) \), and for neo-Hookean materials \( w = a_1(I_1 - 3) + b_1/(\sqrt{I_1 - 1})^2 \). We will use a modified neo-Hookean model [BB98], but any other model results in a similar implementation.

**Space Discretization** In order to properly derive the equations of motion, we discretize space from the onset by approximating the elastic body using a finite dimensional simplicial mesh as routinely done in linear Finite Element methods, i.e., using linear basis functions \( \mathcal{N} \) associated to each vertex of the mesh. The position of the mesh vertices is described by the state variable \( q \), and a motion of the mesh is represented by a time-dependent function \( q(t) \). The spatially-discrete kinetic energy is formulated as

\[
K_d = \frac{1}{2} \mathbf{q}^T M \ddot{\mathbf{q}},
\]

where \( M \) is the lumped mass matrix, i.e., \( M_{kk} \) is the mass inside the dual cell of vertex \( k \) (be it the barycentric, or Voronoi dual cell) and \( M_{kl} = 0 \). The discrete potential energy, associated with the total stored energy (excluding gravity), is denoted by \( W \).

© The Eurographics Association 2006.
Time Discretization Using the same discrete setup as in Section 3.2, the time-discrete Lagrangian $L^d$ can now be written as:

$$L^d(q_k, v_{k+1}, h_k) = h_k \left[ \frac{1}{2} v_{k+1}^T M v_{k+1} - W(q_k + \alpha h_k v_{k+1}) \right]$$

where we used a one point-quadrature (with $\alpha \in [0, 1]$), midpoint for $\alpha = 1/2$ for the integration of $W$. Consequently, the partial derivatives are easily expressed as:

$$D_1 L^d(q_k, v_{k+1}, h_k) = -h_k \nabla W(q_k + \alpha h_k v_{k+1})$$

$$D_2 L^d(q_k, v_{k+1}, h_k) = h_k \left[ M v_{k+1} - \alpha h_k \nabla W(q_k + \alpha h_k v_{k+1}) \right]$$

5.2. Damping

For damping, we propose an extension of the constraint-based damping model of Baraff and Witkin [BW98]. Our idea is to use the strain energy function to “measure” and damp the amount of deformation happening in one step, tantamount to a generalized Rayleigh damping. As discussed in the previous paragraph, the strain energy $W$ is a function of the Cauchy tensor $C$, which itself is a function of the initial configuration $\hat{q}$ and the deformed configuration $q$: $W = W(C(\hat{q}, q))$. Thus we propose to simply compute the discrete damping force term as

$$F_{\text{damp}}(q_k, v_{k+1}) = -k_D \nabla W(C(q_k, q_k + h_k v_{k+1})).$$

Implementation of these damping forces is simple: for explicit integration, damping is added to $F^{d+}$; while for implicit integration, it is added to $F^{d-}$ (improving the conditioning of the non-linear problem as it dampens the dynamics). Notice that the strain energy function depends on the gradient of the deformation field, so our damping model depends on the tensor $\nabla \tau$. In particular, when the stored energy function of a spring is used, our model boils down to the traditional $-k_D \tau$ force. Similarly, for quadratic constraints, it becomes equivalent to the model proposed in [BW98]. Numerical experiments demonstrating the quality of this damping model (in particular, the fact that it does not reduce either linear or angular momentum) are described in Figure 3.

5.3. Numerical Integration

The particular choice of quadrature rule we have made thus far was designed purposely, for two distinct reasons. First, this quadrature allows fast time integration since finding the next position of a system only uses the state variables of the previous position. Second, despite its simplicity, the resulting scheme allows first and second order accuracy, the typical type of accuracy used in graphics. Finally, it also allows a choice for the user to go with a fast, explicit integration, or an implicit integration. We now describe the distinct integration schemes obtained depending on the value of $\alpha$ when an elastic object is simulated with external forces $F_{\text{ext}}$ and using our damping model.

Explicit Time Integration The choice $\alpha = 0$ leads to a fully explicit, linear-accurate integration scheme: no minimization is needed. In particular, one can bootstrap the integration by setting $q_0 = \hat{q}$ (initial position), $p_0 = \dot{q}_0 = 0$ (object at rest), then performing the following updates:

$$v_{k+1} = M^{-1} \left[ p_k - h_k \nabla W(q_k) - k_D \nabla W(C(q_{k-1}, q_k)) + h_k F_{\text{ext}}(q_k) \right]$$

$$p_{k+1} = M v_{k+1}$$

$$q_{k+1} = q_k + h_k v_{k+1}.$$  

Notice that we handled the dissipating term in an explicit manner to keep the overall procedure fully explicit.

Implicit time integration For all other $\alpha \in [0, 1]$, our integrator starts by finding $v_{k+1}$ that minimizes the Lilyan $E$:

$$E(v_{k+1}) = \frac{h_k}{2} v_{k+1}^T M v_{k+1} + h_k \left( 1 - \frac{\alpha}{\alpha} \right) W(q_k + \alpha h_k v_{k+1})$$

$$- h_k \left( \frac{\alpha}{\alpha} \right) E_{\text{ext}}(q_k + \alpha h_k v_{k+1})$$

$$+ k_D W(C(q_k, q_k + h_k v_{k+1})) - h_k p_k v_{k+1}.$$  

where $E_{\text{ext}}$ is the integral of the external force $F_{\text{ext}}$ with respect to $v_{k+1}$. When non-integrable external forces are applied, the forced terms mentioned in Section 3.4 can be used instead. Other variables are then updated directly via the fol-
Figure 3: Damping is added to the same setup as in Fig. 2. The energy plot shows a smooth decrease over time, while momenta are still exactly preserved, even after 2 million time steps (explicit integration was used, with a constant time step of 0.004 s).

Following rules:

\[
p_{k+1} = M v_{k+1} - h_k \alpha \nabla W(q_k + \alpha h_k v_{k+1}) + h_k \alpha F_{\text{ext}}(q_k + \alpha h_k v_{k+1}),
\]

\[
q_{k+1} = q_k + h_k v_{k+1}.
\]

Here, note that we included the dissipative terms directly inside the Ljuyan function as it does not change the implicit nature of this choice of integrator. Note finally that this scheme is linear accurate, except for \( \alpha = 1/2 \) where the quadrature becomes quadratic accurate—thus, so is the scheme. This scheme was used to produce the animation of the bunny model in Figure 4.

5.4. Comparisons of Numerical Methods

In order to assess the computational gain that our update via minimization confers, we ran the test presented in Fig. 2, but this time with a timestep size of \( h = 0.01 \) s, and at various spatial resolutions. We employed the widely-used TAO/PETSc solvers [BMMS04] as neutral numerical tools instead of relying on our in-house solvers. All our tests were run on a 3GHZ XeonHT PC with 2.50GB of RAM. For a very low-res bar (2K tets, 330 vertices), the speed-up of minimization (using tao_nls, implementing Newton’s method with line search for unconstrained minimization) vs. non-linear root finding (PETSc’s line-search SNES nonlinear solver) is only 20%. However, as soon as the number of nodes increases, results show a clear superiority of the minimization procedure; for a bar with 12.5K tets (2000 vertices), the speed-up brought by our minimization update is already 2.6, while the same bar with now 24K tets (3784 vertices) yields a speed-up of 3.

We also experimented with larger simulations to test both the robustness and practicality of our family of integrators. We concluded that the correct energy behavior and momenta preservation with and without damping (demonstrated in Figs. 2 and 3) are indeed important qualities that most other integrators (non-symplectic Newmark, implicit Euler, etc) do not have. In particular, being able to define damping in a manner fairly independent of the time step size is a significant advantage when trying to design a particular animation: the behavior of a physically-based object will be consistent throughout a wide range of time step sizes, making previews (i.e., coarse simulations) not noticeably different from final simulations.

6. Conclusions

We have presented an approach to derive general-purpose, fully variational time integrators for a wide class of mechanical systems using a discrete Hamilton-Pontryagin principle. Our approach has the following salient features:

- a minimization procedure replaces the traditional update rules which otherwise require computation-intensive multidimensional root-finding;
- the updates in time can be done explicitly or implicitly, and we demonstrated linear and quadratic accuracy;
- the time integrator is symplectic, and therefore preserves fundamental invariants while demonstrating excellent energy behavior;
- non-conservative forces or mechanical systems with (possibly non-linear) constraints can be handled easily and robustly;
- a novel damping model is easily added to our scheme.

The design of time integrators has not received much attention in our community despite their widespread use. Given the importance of qualitatively correct behavior in computer animation the geometric view is particularly pertinent as it ensures conservation of important quantities even for lower accuracy/higher speed simulations. Because of the general nature of our approach it no less admits high accuracy simulations when called for. An innovative aspect of our work is the introduction of the variational integrability condition which allows us to solve the non-linear problem at each time step (when using implicit integration) through a minimization. Together with the use of velocity/momentum/position variables it promises to play an important role in motion control. For instance, we believe that the optimization scheme

© The Eurographics Association 2006.
proposed in [JMOB05] where the constraints are based on the discrete Hamilton variational principle could significantly benefit from our minimization-based integrators, as it can render the global optimization more scalable. Furthermore, we believe that the discrete Hamilton-Pontryagin principle that we introduce here and the ability to control \( v, q, \) and \( p \) should provide fertile grounds for various control tools (e.g., trajectory planning) as one can alter these quantities during integration to influence the motion accordingly. In particular, variational collision handling along the lines of [FMOW03] could be made much more robust. Finally, we wish to study whether model reduction [KLM00, BJ05] can benefit from the discrete variational integrator framework.

Acknowledgments. We thank Rasmus Tamstorf, Eitan Grinspun, Matt West, Hiroaki Yoshimura, and Michael Ortiz for helpful comments. This research was partially supported by NSF (ACI-0204932, DMS-0453145, CCF-0503786 & 0528101,CCR-0133983), DOE (W-7405-ENG-48/B341492 & DE-FG02-04ER25657), Caltech Center for Mathematics of Information, nVidia, Autodesk, and Pixar.

References