

Compliant Reference

Maxime Tournier

October 17, 2013

Abstract

Quick notes about the mathematical framework used in Compliant. We review the time-stepping scheme, kinematic constraint handling, potential forces as compliant constraints, and KKT systems.

TODO

- finish sections :-)
- references
- wikipedia links
- introduction ?

1 Lagrangian Dynamics

Given an n -dimensional state space Q , we consider a Lagrangian \mathcal{L} defined as usual:

$$\mathcal{L}(q, v) = \frac{1}{2} v^T M v - V(q) \quad (1)$$

where $V(q)$ is the potential energy of the form:

$$V(q) = \frac{1}{2} \|g(q)\|_N^2 \quad (2)$$

for a matrix norm N in the suitable deformation space $Im(g)$. To be fully general, we include a Rayleigh dissipation function $\frac{1}{2} v^T D v$, where D is PSD. The Euler-Lagrange equations of motions are then:

$$\frac{d}{dt} \frac{\partial \mathcal{L}}{\partial v} = \frac{\partial \mathcal{L}}{\partial q} - \frac{\partial D}{\partial v} \quad (3)$$

For simplicity, we restrict to the case where M is constant, which yields:

$$M \dot{v} = -\nabla_q V - D v \quad (4)$$

2 Time Stepping

We now discretize time using a fixed time step h . We use forward differences for \dot{v} as follows:

$$h \dot{v}_{k+1} \approx v_{k+1} - v_k \quad (5)$$

Equation (4) becomes:

$$Mv_{k+1} = Mv_k - h(\nabla_q V_{k+1} + Dv_{k+1}) \quad (6)$$

Or, calling $p_k = Mv_k$ the momentum of the system and $f_k = -\nabla_q V_k$ the potential forces:

$$(M + hD)v_{k+1} = p_k + hf_{k+1} \quad (7)$$

In order to control how implicit our integrator is on forces, we introduced a blending parameter $\alpha \in [0, 1]$ as follows:

$$(M + h\alpha D)v_{k+1} = p_k - h(1 - \alpha)Dv_k + h((1 - \alpha)f_k + \alpha f_{k+1}) \quad (8)$$

We linearize the above using the Hessian of the potential energy (*i.e.* stiffness matrix):

$$\nabla_q V_{k+1} \approx \nabla_q V_k + \nabla_q^2 V_k (q_{k+1} - q_k) \quad (9)$$

Finally, we use the following position time-stepping scheme:

$$q_{k+1} = q_k + h((1 - \beta)v_k + \beta v_{k+1}) \quad (10)$$

where $\beta \in [0, 1]$ is again a blending parameter. Letting the stiffness matrix be $K = \nabla_q^2 V_k$ ¹ and $d_k = -Dv_k$ be the damping forces, we put everything together as:

$$(M + h\alpha D + h^2\alpha\beta K)v_{k+1} = p_k + h(f_k + (1 - \alpha)d_k) - \alpha(1 - \beta)h^2Kv_k \quad (11)$$

(and yes, this is very ugly). Fortunately, in a sane world everyone would use the much friendlier, fully implicit scheme $\alpha = \beta = 1$ as follows:

$$(M + hD + h^2K)v_{k+1} = p_k + hf_k \quad (12)$$

3 Response Matrix, Net Momentum

From now on, we will refer to the *response matrix* defined as follows:

$$W = (M + h\alpha D + h^2\alpha\beta K) \quad (13)$$

We will also refer to the *net momentum* of the system at time step k :

$$c_k = p_k + h(f_k + (1 - \alpha)d_k) - \alpha(1 - \beta)h^2Kv_k \quad (14)$$

The time-stepping scheme (11) thus involves solving the following linear system:

$$Wv_{k+1} = c_k \quad (15)$$

The numerical solvers for time-stepping will be described in section 11.

¹ This is the opposite of the stiffness matrix as defined in SOFA, *i.e.* $\nabla f(q) = -\nabla_q^2 V$. I find it easier to work with PSD matrices.

4 Constraints

We now introduce holonomic constraints of the form:

$$g(q) = 0 \quad (16)$$

where g again maps kinematic DOFs to a suitable deformation space $Im(g)$. Such constraints are satisfied by introducing constraint forces $J^T \lambda$, where $J = dg$ is the constraint Jacobian matrix, and λ are the *Lagrange multipliers*:

$$M\dot{v} = -\nabla_q V - Dv + J^T \lambda \quad (17)$$

$$g(q) = 0 \quad (18)$$

Again, we discretize time and \dot{v} as follows:

$$Mv_{k+1} = p_k + h(f_{k+1} + d_{k+1} + J_{k+1}^T \lambda_{k+1}) \quad (19)$$

$$g(q_{k+1}) = 0 \quad (20)$$

And again:

$$g(q_{k+1}) = g_k + hJ_k((1 - \beta)v_k + \beta v_{k+1}) \quad (21)$$

At this point we become lazy and approximate constraints as *affine* functions, meaning that $J_{k+1} \approx J_k$, otherwise computing the constraint Hessian d^2g would be too costly, and would result in a non-linear system to solve anyways (*i.e.* with terms involving $v_{k+1}^T d^2g_k \lambda_{k+1}$). As we will see later, such approximation is consistent with our treatment of potential forces as compliant constraints. The time-discrete system is then:

$$Mv_{k+1} = p_k + h(f_{k+1} + d_{k+1} + J_k^T \lambda_{k+1}) \quad (22)$$

$$J_k v_{k+1} = -\frac{g_k}{\beta h} - \frac{1 - \beta}{\beta} J_k v_k \quad (23)$$

At this point, we *could* introduce an implicit blending between λ_k and λ_{k+1} , but this would result in unneeded complication as λ_{k+1} would need to be computed anyways. The blended force would then be:

$$J_k^T((1 - \alpha)\lambda_k + \alpha\lambda_{k+1})$$

which simply offsets λ_{k+1} with respect to λ_k . We will thus happily ignore such blending. The final linear system to solve (for v_{k+1} and λ_{k+1}) becomes:

$$Wv_{k+1} - J_k^T \lambda_{k+1} = c_k \quad (24)$$

$$J_k v_{k+1} = -b_k \quad (25)$$

where $b_k = \frac{g_k}{\beta h} + \frac{1 - \beta}{\beta} J_k v_k$. Before leaving, note the sneaky accounting of h inside λ_{k+1} .

5 Compliance

We will now establish a connection between elastic forces and constraint forces through a compliance matrix. Remember that we defined potential energy as:

$$V(q) = \frac{1}{2} \|g(q)\|_N^2 \quad (26)$$

where g maps kinematic DOFs to an appropriate deformation space $Im(g)$. This means the potential forces are:

$$f = -\nabla_q V = J^T N g(q) \quad (27)$$

Now, the Hessian or stiffness matrix is :

$$K = \nabla_q^2 V = dJ^T N g(q) + J^T N J \quad (28)$$

As in section 4, we are too lazy for computing second derivatives and approximate deformation mappings as affine maps, meaning that $dJ \approx 0$. We are left with the following response matrix (13):

$$W = (M + h\alpha D + h^2\alpha\beta J_k^T N J_k) \quad (29)$$

The net momentum (14) becomes:

$$c_k = p_k + h(1 - \alpha)d_k - hJ_k^T N (g_k + h\alpha(1 - \beta)J_k v_k) \quad (30)$$

If we write the potential force as $J_k^T \lambda_{k+1}$, we have:

$$\lambda_{k+1} = -N (h^2\alpha\beta J_k v_{k+1} + h g_k + h^2\alpha(1 - \beta)J_k v_k) \quad (31)$$

It turns out that this system can be rewritten as a *larger* system:

$$\begin{pmatrix} M + h\alpha D & -J_k \\ J_k & \frac{N^{-1}}{h^2\alpha\beta} \end{pmatrix} \begin{pmatrix} v_{k+1} \\ \lambda_{k+1} \end{pmatrix} = \begin{pmatrix} p_k + h(1 - \alpha)d_k \\ -\frac{g_k}{h\alpha\beta} - \frac{(1-\beta)}{\beta} J_k v_k \end{pmatrix} \quad (32)$$

(pffew !) One can notice that this system is *almost* exactly the one we obtained with kinematic constraints in (24), with $\alpha = 1$, with the exception of the N^{-1} term in the (2, 2) block. We call matrix $C = N^{-1}$ the *compliance* matrix. We see that kinematic constraints simply correspond to a zero compliance matrix, *i.e.* infinite stiffness, as one would intuitively expect. It can be checked that taking α into account for constraint forces produces the same system with $C = 0$. (TODO) We now recall the main results for the unified elastic/constraint treatment:

Linear System

$$\begin{pmatrix} W & -J \\ -J & -\frac{C}{h^2\alpha\beta} \end{pmatrix} \begin{pmatrix} v \\ \lambda \end{pmatrix} = \begin{pmatrix} c_k \\ b_k \end{pmatrix} \quad (33)$$

Response Matrix

$$W = M + h\alpha D \quad (34)$$

Net Momentum

$$c_k = p_k + h(1 - \alpha)d_k \quad (35)$$

Constraint Value

$$b_k = \frac{g_k}{h\alpha\beta} + \frac{(1 - \beta)}{\beta} J_k v_k \quad (36)$$

TODO pourquoi c'est cool: transition seamless entre contraintes et elasticite, traitement unifi contraintes/elasticit, conditionnement pour les sytemes trs raides. par contre ca fait des systemes plus gros.

6 KKT Systems, Compliance and Relaxation

At this point, it is probably a good idea to introduce a few notions on saddle-point (or KKT) systems. Such systems are (for now, linear) systems of the form:

$$\begin{pmatrix} Q & -A^T \\ -A & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} c \\ b \end{pmatrix} \quad (37)$$

KKT systems typically arise from the Karush-Kuhn-Tucker (hence the name) optimality conditions of constrained optimization problems. For instance, the KKT system (37) corresponds to the following equality-constrained Quadratic Program (QP):

$$\underset{Ax+b=0}{\operatorname{argmin}} \quad \frac{1}{2}x^T Qx - c^T x \quad (38)$$

The KKT system summarizes the optimality conditions for the following unconstrained function (also called the *Lagrangian*, this time in the context of optimization):

$$\mathcal{L}(x, \lambda) = \frac{1}{2}x^T Qx - c^T x - (Ax + b)^T \lambda \quad (39)$$

and whose critical points (in fact, saddle-points) solve the constrained problem (38). As one can see, the constrained integrator (24) is an example of such KKT systems:

$$\begin{pmatrix} W & -J \\ -J & 0 \end{pmatrix} \begin{pmatrix} v \\ \lambda \end{pmatrix} = \begin{pmatrix} c_k \\ b_k \end{pmatrix} \quad (40)$$

As symmetric *indefinite* systems, they tend to be more difficult to solve than positive definite ones: CG and Cholesky LDL^T factorizations may breakdown

on such systems. However, when the $(1, 1)$ block (here, matrix Q) is invertible, one can use *pivoting* to obtain the following smaller system:

$$x = Q^{-1} (c - A^T \lambda) \quad (41)$$

$$-Ax = AQ^{-1}A^T \lambda - AQ^{-1}c = b \quad (42)$$

$$AQ^{-1}A^T \lambda = b - AQ^{-1}c \quad (43)$$

This smaller system (43) is known as the *Schur* complement of the KKT system. It is always SPD as long as A^T is full column-rank, but requires to invert Q which might be costly in practice. The Schur complement system also corresponds to an (unconstrained) optimization problem:

$$\operatorname{argmin}_{\lambda} \quad \frac{1}{2} \lambda^T S \lambda - s^T \lambda \quad (44)$$

where $S = AQ^{-1}A^T$ is the Schur complement, and $s = b + AQ^{-1}c$. The minimized quantity has a physical interpretation: typically, holonomic constraints minimize the total kinetic energy. If we now consider systems with compliance, such as:

$$\begin{pmatrix} Q & -A^T \\ -A & -D \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} c \\ b \end{pmatrix} \quad (45)$$

we see that they correspond to the following Lagrangian:

$$\mathcal{L}(x, \lambda) = \frac{1}{2} x^T Q x - c^T x - (Ax + b)^T \lambda - \frac{1}{2} \lambda^T D \lambda \quad (46)$$

One can check that the Schur complement becomes $S + D$, and corresponds to:

$$\operatorname{argmin}_{\lambda} \quad \frac{1}{2} \lambda^T S \lambda - s^T \lambda + \frac{1}{2} \lambda^T D \lambda \quad (47)$$

We see that D acts as a form of numerical damping on the resolution of constraints, by biasing the solution of the constrained system towards zero. It is a form of *constraint relaxation*. Physically, elastic forces minimize a mix between the kinetic energy of the system and the D -norm of the forces λ . Under their most general form, KKT systems include unilateral and complementarity constraints. We will develop these in more details in section 8.

7 Geometric Stiffness

For both constraint and elastic forces, we happily neglected second-order derivatives of the deformation mapping g . In the case of elastic forces, this resulted in a nice stiffness matrix of the form:

$$\nabla_q^2 V(q) \approx J^T N J = K \quad (48)$$

which enabled us to treat elastic forces as a compliant kinematic constraints. But what about the second order terms $\tilde{K} = (dJ)^T N g(q)$? First of all, when the configuration space is a vector space Q and g is \mathcal{C}^2 -continuous, the Schwarz' theorem ensures that $\nabla_q^2 V$ is symmetric, hence so is \tilde{K} . We call \tilde{K} the *geometric stiffness*, as it is induced by the variation of the deformation mapping g . We do not know whether in the general case, it is possible to factor *both* K and \tilde{K} as:

$$\nabla_q^2 V(q) = K + \tilde{K} \stackrel{?}{=} J^T (N + \tilde{N}) J \quad (49)$$

even though some specific examples exist (*e.g.* mass spring systems). Of course, it is always possible to apply a Cholesky factorization directly on $\nabla_q^2 V(q)$ as:

$$\nabla_q^2 V(q) = LDL^T \quad (50)$$

and get back on our feet, but this would be highly inefficient in practice. Therefore, unless an ad-hoc derivation provides the needed factorization, we are left with only one alternative: treat the geometric stiffness as a regular stiffness instead of compliance.

8 Unilateral Constraints

TODO: general KKT systems

9 Stabilization

TODO

10 Restitution

TODO: velocity constraints for contact with restitution (rigid), mention Generalized Reflections

11 Numerical Solvers

TODO: CG, Cholesky, Minres, GS, PGS, Sequential Impulses