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:

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

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

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

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:

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

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

$$M\dot{v} = -\nabla_q V - Dv \tag{4}$$

2 Time Stepping

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

$$h\,\dot{v}_{k+1} \approx v_{k+1} - v_k \tag{5}$$

Equation (4) becomes:

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

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

$$\left(M + h\,D\right) v_{k+1} = p_k + hf_{k+1} \tag{7}$$

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

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

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

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

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

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

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

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

(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:

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

3 Response Matrix, Net Momentum

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

$$W = \left(M + h\alpha D + h^2\alpha\beta K\right) \tag{13}$$

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

$$c_k = p_k + h\left(f_k + (1-\alpha)d_k\right) - \alpha(1-\beta)h^2 K v_k \tag{14}$$

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

$$Wv_{k+1} = c_k \tag{15}$$

The numerical solvers for time-stepping will be described in section 12.

4 Constraints

We now introduce holonomic constraints of the form:

$$g(q) = 0 \tag{16}$$

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*:

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

$$g(q) = 0 \tag{18}$$

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

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

$$g\left(q_{k+1}\right) = 0 \tag{20}$$

And again:

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

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 $v_{k+1}^T d^2 g_k \lambda_{k+1}$). As we will see later, such approximation is consistent with our treatment of potential forces as compliant constraints. The time-discrete system is then:

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

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

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:

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

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:

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

$$J_k v_{k+1} = -b_k \tag{25}$$

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 KKT Systems and Schur Complement

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:

$$\begin{pmatrix} Q & -A^T \\ -A & 0 \end{pmatrix} \begin{pmatrix} x \\ \lambda \end{pmatrix} = \begin{pmatrix} c \\ b \end{pmatrix} \tag{26}$$

KKT systems typically arise from the Karush-Kuhn-Tucker (hence the name) optimality conditions of constrained optimization problems. For instance, the KKT system (26) corresponds to the following Quadratic Program (QP):

$$\underset{Ax+b=0}{\text{argmin}} \quad \frac{1}{2} x^T Q x - c^T x \tag{27}$$

The alternate name *saddle-point* comes from the fact that the system matrix is symmetric *indefinite*, and the system solution corresponds to a saddle point of the corresponding quadratic form. As one can see, the constrained integrator (24) is an example of a KKT system:

$$\begin{pmatrix} W & -J \\ -J & 0 \end{pmatrix} \begin{pmatrix} v \\ \lambda \end{pmatrix} = \begin{pmatrix} c_k \\ b_k \end{pmatrix} \tag{28}$$

where we dropped subscripts for readability. As symmetric indefinite systems, they tend to be more difficult to solve than SPD 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:

$$x = Q^{-1} \left(c - A^T \lambda \right) \tag{29}$$

$$-Ax = AQ^{-1} A^T \lambda - AQ^{-1} c = b \tag{30}$$

$$AQ^{-1} A^T \lambda = b - AQ^{-1} c \tag{31}$$

This smaller system (31) is known as the *Schur* complement of the KKT system. It is always PSD as long as A^T is full column-rank, but requires to invert Q which might be costly in practice. The Schur form also corresponds to an (unconstrained) optimization problem, which can provide insight for compliant systems as we will see later. Under the most general form, KKT systems include unilateral and complementarity constraints. We will develop these in more details in section 9.

6 Compliance

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

$$V(q) = \frac{1}{2} \left|\left| g(q) \right|\right|_N^2 \tag{32}$$

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

$$f = -\nabla_q V = J^T N g(q) \tag{33}$$

Now, the Hessian or stiffness matrix is :

$$K = \nabla_q^2 V = dJ^T N g(q) + J^T N J \tag{34}$$

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):

$$W = \left(M + h\alpha D + h^2\alpha\beta J_k^T N J_k\right) \tag{35}$$

The net momentum (14) becomes:

$$c_k = p_k + h(1-\alpha)d_k - hJ_k^T N\left(g_k + h\alpha(1-\beta)J_k v_k\right) \tag{36}$$

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

$$\lambda_{k+1} = -N\left(h^2\alpha\beta J_k v_{k+1} + hg_k + h^2\alpha(1-\beta)J_k v_k\right) \tag{37}$$

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

$$\begin{pmatrix} M + h\alpha D & -J_k \\ J_k & \frac{N^{-1}}{h^2\alpha\beta} \end{pmatrix} \begin{pmatrix} v_{k+1} \\ \lambda_{k+1} \end{pmatrix} = \begin{pmatrix} p_k + h(1-\alpha)d_k \\ -\frac{g_k}{h\alpha\beta} - \frac{(1-\beta)}{\beta} J_k v_k \end{pmatrix} \tag{38}$$

(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:

KKT system

$$\begin{pmatrix} W & -J \\ -J & -\frac{C}{h^2\alpha\beta} \end{pmatrix} \begin{pmatrix} v \\ \lambda \end{pmatrix} = \begin{pmatrix} c_k \\ b_k \end{pmatrix} \tag{39}$$

Response Matrix

$$W = M + h\alpha D \tag{40}$$

Net Momentum

$$c_k = p_k + h(1-\alpha)d_k \tag{41}$$

Constraint Value

$$b_k = \frac{g_k}{h\alpha\beta} + \frac{(1-\beta)}{\beta} J_k v_k \tag{42}$$

7 Damped KKT Systems

TODO: effect of C on the schur complement, lagrangian relaxation

8 Geometric Stiffness

TODO: what is left when we approximate g as an affine function, how to recover it

9 Unilateral Constraints

TODO: general KKT systems

10 Stabilization

TODO

11 Restitution

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

12 Numerical Solvers

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