A dimensional reduction algorithm and software for acyclically dependent constraints

P. Areias[1]
[1] *University of Évora and CERIS (IST), pmaa@uevora.pt*

For discrete equations of motion with acyclically dependent equality constraints and within the context of the null-space method, a novel Algorithm is introduced. By topologically ordering the degrees-of-freedom in the constraint gradient matrix, the saddle point problem can be solved with a sparse triangular system for the constraint equations. Given n unconstrained equations and m equality constraints (nonlinear), the original square sparse $(n+m)^2$ system is replaced by a sparse system $(n-m)^2$ and a sparse triangular solve with m^2 coefficients and $n-m$ right-hand-sides. This triangular solve, since it involves three sparse matrices (in existing literature only two out of three matrices are sparse), is here discussed in detail. Seven sparse operations are addressed (five standard and two non-standard) and are required in addition to some specific (*ad-hoc*) operations. Source code is available in GitHub [1].

We introduce $t \in \mathbb{R}_0^+$ as the time and $\boldsymbol{q} \in \mathbb{R}^n$ as the unconstrained degree-of-freedom vector. We consider a $n-$dimensional dynamic system subject to a set of m equality constraints:

$$\begin{aligned} \delta\boldsymbol{q}\cdot\boldsymbol{r}\left(\boldsymbol{q},\dot{\boldsymbol{q}},\ddot{\boldsymbol{q}},t\right) &= 0 \\ \boldsymbol{g}\left(\boldsymbol{q},\dot{\boldsymbol{q}},t\right) &= \boldsymbol{0} \end{aligned} \tag{1}$$

with $\delta\boldsymbol{q}$ belonging to the null space of the constraint gradients $\nabla\boldsymbol{g}\cdot\delta\boldsymbol{q} = \boldsymbol{0}$ where ∇ is the gradient with respect to $\boldsymbol{q}$ *including* the dependence on $\dot{\boldsymbol{q}}$ and $\ddot{\boldsymbol{q}}$.

We now make use of a *general* form of integration for a given time step n such that a variation $\delta\dot{\boldsymbol{q}} \equiv c_{\dot{q}}\delta\boldsymbol{q}$ and $\delta\ddot{\boldsymbol{q}} = c_{\ddot{q}}\delta\boldsymbol{q}$. For completeness we also introduce $\delta\boldsymbol{q} = c_q\delta\boldsymbol{q}$. Using a set of Lagrange multipliers $\boldsymbol{\lambda} \in \mathbb{R}^m$, and omitting the arguments of $\boldsymbol{r}$ and $\boldsymbol{g}$, Newton iteration provides (after a small exercise to move the Lagrange multiplier vector $\boldsymbol{\lambda}$ to the left-hand-side):

$$\left[\begin{array}{c|c} \boldsymbol{K} & -\nabla\boldsymbol{g}^T \\ \hline -\nabla\boldsymbol{g} & \boldsymbol{0} \end{array}\right]\left\{\begin{array}{c} \Delta\boldsymbol{q} \\ \boldsymbol{\lambda} \end{array}\right\} = \left\{\begin{array}{c} \boldsymbol{f} \\ \boldsymbol{g} \end{array}\right\} \tag{2}$$

where $\boldsymbol{K} = \nabla\boldsymbol{r} - \boldsymbol{\lambda}\cdot\nabla^2\boldsymbol{g}$, $\boldsymbol{B} = -\nabla\boldsymbol{g}$ and $\boldsymbol{f} = -\boldsymbol{r}$.

For rank$[\nabla\boldsymbol{g}] = m$, we use a basis for the null space of $\nabla\boldsymbol{g}$, $\boldsymbol{Z} \in \mathbb{R}^{n\times(n-m)}$ where columns of $\boldsymbol{Z}$ are basis vectors. Introducing $\boldsymbol{Y} \in \mathbb{R}^{n\times m}$ such that $[\boldsymbol{Z}\,|\,\boldsymbol{Y}]$ spans $\mathbb{R}^n$, we partition $\Delta\boldsymbol{q}$ as the sum of a particular solution $\Delta\hat{\boldsymbol{q}}$ and a term $\Delta\check{\boldsymbol{q}}$ depending on free parameters:

$$\Delta\boldsymbol{q} = \Delta\hat{\boldsymbol{q}} + \Delta\check{\boldsymbol{q}} \tag{3}$$

with $\Delta\hat{\boldsymbol{q}} = \boldsymbol{Y}\Delta\boldsymbol{q}_1$ and $\Delta\check{\boldsymbol{q}} = \boldsymbol{Z}\Delta\boldsymbol{q}_2$. Using the property $\nabla\boldsymbol{g}\boldsymbol{Z} = \boldsymbol{0}$ we obtain the solution [2] as:

1. Determine $\Delta\boldsymbol{q}_1$ by solving the system $\boldsymbol{B}\boldsymbol{Y}\Delta\boldsymbol{q}_1 = \boldsymbol{g}$;
2. Determine $\Delta\boldsymbol{q}_2$ by solving the system $\boldsymbol{Z}^T\boldsymbol{K}\boldsymbol{Z}\Delta\boldsymbol{q}_2 = \boldsymbol{Z}^T\left(\boldsymbol{f} - \boldsymbol{K}\boldsymbol{Y}\Delta\boldsymbol{q}_1\right)$;
3. Determine the complete set of unknowns $\Delta\boldsymbol{q} = \boldsymbol{Y}\Delta\boldsymbol{q}_1 + \boldsymbol{Z}\Delta\boldsymbol{q}_2$;
4. Determine $\Delta\boldsymbol{\lambda}$ by solving the constraint equation $\boldsymbol{Y}^T\boldsymbol{B}^T\boldsymbol{\lambda} = \boldsymbol{Y}^T\left(\boldsymbol{f} - \boldsymbol{K}\Delta\boldsymbol{q}\right)$;

Obtaining matrices $\boldsymbol{Y}$ and $\boldsymbol{Z}$ is a task that can be accomplished by partitioning of degrees of freedom corresponding to specific columns of $\boldsymbol{B}$. This is called *direct elimination* by Fletcher and Johnson [3]. By partitioning (by permutation) $\boldsymbol{B}$ in two sub-matrices: a non-singular $\boldsymbol{B}_1 \in \mathbb{R}^{m\times m}$ and $\boldsymbol{B}_2 \in \mathbb{R}^{m\times(n-m)}$: $\boldsymbol{B} = [\boldsymbol{B}_1 \,|\, \boldsymbol{B}_2]$ we obtain the *fundamental* basis $\boldsymbol{Z}$ and the matrix $\boldsymbol{Y}$:

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{B}_1^{-1} \\ \boldsymbol{0} \end{bmatrix} \in \mathbb{R}^{[m+(n-m)]\times m} \tag{4}$$

$$\boldsymbol{Z} = \begin{bmatrix} -\boldsymbol{B}_1^{-1}\boldsymbol{B}_2 \\ \boldsymbol{I} \end{bmatrix} \in \mathbb{R}^{[m+(n-m)]\times(n-m)} \tag{5}$$

With the choice for $\boldsymbol{Y}$ (4), $\Delta\boldsymbol{q}_1 = \boldsymbol{g}$, see Benzi, Golub and Liesen [4]. Various alternatives for solution of this problem are presented by Rees and Scott [2]. We find a lower-triangular form of $\boldsymbol{B}_1$ (and therefore $\boldsymbol{B}_1^{-1}$) by identifying the pivot order a-priori[1] and performing a topological ordering of degrees-of-freedom and therefore of constraints (by the a-priori pivots) in the matrix $\boldsymbol{B}$. The directed graph of a triangular matrix is acyclic. In full (dense) format, the topologically ordered $\boldsymbol{B}$ becomes

$$\boldsymbol{B}_p = -[\nabla\boldsymbol{g}_1 \,|\, \nabla\boldsymbol{g}_2] = [\boldsymbol{L}_1 \,|\, \boldsymbol{B}_2] \tag{6}$$

$$= \begin{bmatrix} B_{1,1} & 0 & \cdots & 0 & B_{1,m+1} & \cdots & B_{1,n} \\ B_{2,1} & B_{2,2} & \cdots & 0 & B_{2,m+1} & \cdots & B_{2,n} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ B_{m,1} & B_{m,2} & \cdots & B_{m,m} & B_{m,m+1} & \cdots & B_{m,n} \end{bmatrix} \tag{7}$$

where both constraint permutations and degree-of-freedom permutation of $\boldsymbol{B}$ were performed. We have $\boldsymbol{Y}$ and $\boldsymbol{Z}$ defined as:

$$\boldsymbol{Y} = \begin{bmatrix} \boldsymbol{L}_1^{-1} \\ \boldsymbol{0} \end{bmatrix} \tag{8}$$

$$\boldsymbol{Z} = \begin{bmatrix} \boldsymbol{Z}_t \\ \boldsymbol{I} \end{bmatrix} \tag{9}$$

where $\boldsymbol{Z}_t$ is determined by solving the triangular system with $n-m$ right-hand-sides:

$$\boldsymbol{L}_1\boldsymbol{Z}_t = -\boldsymbol{B}_2 \tag{10}$$

It is possible to directly solve for $\boldsymbol{Z}$ by changing $\boldsymbol{L}_1$ so that a $(n-m)^2$ identity matrix appears beneath $\boldsymbol{Z}_t$:

$$\boldsymbol{L}_1^{\star}\boldsymbol{Z} = -\boldsymbol{B}_2 \tag{11}$$

The explicit form (8) of $\boldsymbol{Y}$ is not required in our Algorithm.

References

[1] P. Areias. Sparse acyclic library. `https://github.com/PedroAreias/sparseacyclic`, 2018.

[2] T. Rees and J. Scott. A comparative study of null-space factorizations for sparse symmetric saddle point systems. *Numerical Linear Algebra with Applications*, pages e2103–n/a, 2017. Early View.

[3] R. Fletcher and T. Johnson. On the stability of null-space methods for KKT systems. *SIAM J. Matrix Anal. Appl.*, 18(4):938–958, 1997.

[4] M. Benzi, G.H. Golub, and J. Liesen. Numerical solution of saddle point problems. *Acta Numerica*, 14:1–137, 2005.

[1] First active column for each row of $\boldsymbol{B}$ is the selected pivot degree-of-freedom.