FAST AND STABLE SIMULATION OF GRANULAR MATTER AND MACHINES

Claude Lacoursière[1], Martin Servin[1], and Anders Backman[2]

1. Umeå University. 2. Algoryx Simulation.

Fast and stable simulation of granular matter and machines is achieved through a unified model for particles, fluids and rigid bodies based on constraint multibody systems and using dynamic resolution.

INTRODUCTION

We present mathematical models and numerical techniques designed to address the challenges of real-time simulator training for earth moving equipment involving heavy vehicles and granular matter. The same techniques have also been applied to mix granular matter with a nearly incompressible SPH model of water. The central elements in our technique are a unified model based on constrained multibody systems including point particles and rigid bodies, a dynamic resolution technique based on merging and splitting elementary bodies. The numerical integration is based on a time-discrete variational formulation of analytic mechanics, which is closely related to the *Rattle* and *Shake* [3] solvers. We have modified these by introducing stabilized linear approximations to avoid solving the nonlinear equations exactly [5]. Frictional contact forces are modeled using a linear complementarity (LCP) formulations similar to those already well-known in this field [8]. Our deviation from these models is mainly in the solution technique, which mixes a direct LCP solver for computing normal forces and a Gauss-Seidel (GS) process to solve for the frictional ones. The SPH model includes kinematic

constraints for incompressibility and for the boundary conditions. The latter are formulated as non-penetration conditions producing also buoyancy for immersed bodies. Another novelty of our overall method is the introduction of dynamic resolution using a merge-split technique based on local analysis of contact forces and complementarity conditions. A significant result is the real-time simulation of a tractor, which can shovel granular matter consisting of moderately small elements integrated in an interactive 3D application used for operator training. The conventional computational techniques for granular materials are based either on modeling the system as distinct particles [7] or using continuum mechanics with specific constitutive laws for granular matter and discretization using finite elements or mesh-free methods. In turn, the discrete element techniques are phrased either as penalty methods, e.g., the Hertz contact model, requiring very small time steps. Complementarity formulations of contact models solved using Gauss-Seidel iterations [9] can use larger time steps, but have slow, linear convergence. By contrast, interactive applications put strong constraints on speed and time steps. There are usually roughly 10 milliseconds to compute a 1/60 second update. With this budget, there is usually only time for a single step with size $h = 1/60$ s. In turn the stable simulation of machines demands precise solutions to maintain stability. One additional reason for the choice of our integration method is that it is solidly anchored in discrete time mechanics and is in fact much more stable than higher order explicit methods. It is also necessary to explicitly compute the Lagrange multipliers with good precision, using a *direct* method for the most part, or with a good preconditioner. Our experience has shown that neither explicit penalty methods nor GS based solver can be tuned to provide sufficient stability.

THEORY AND METHODS

Our starting point is the descriptor form of multibody dynamics. We write M for the system mass matrix, G for the constraint Jacobians, x for the generalized position variables v for the velocities, and λ for the Lagrange multipliers. The constraint indicators are $g(x)$ and these can be either equality or inequality conditions.

After discretization and linearization, our time stepping method requires the solution of the following *Mixed Linear Complementarity Problem* (*MLCP*)

$$\begin{aligned} &Hz + b = w_+ - w_- \\ &0 \le z - l \perp w_+ \ge 0 \\ &0 \le w - z \perp w_- \ge 0 \end{aligned}, \qquad \text{with } H = \begin{bmatrix} M & -G^T \\ G & \Sigma \end{bmatrix} \text{ and } z = \begin{bmatrix} v \\ \lambda \end{bmatrix}. \qquad (1)$$

Vectors l and u are lower and upper bounds, and vector b contains stabilization terms to avoid numerical drift away from the in constraint surface. The slack variables w_+, w_- are discarded once the solution is computed. The inequalities and orthogonality should be understood component wise. Σ is a non-negative diagonal matrix which protects against constraint degeneracy and introduces a small amount of elasticity. We typically use perturbations which are eight orders of magnitudes less than the masses. Details of this model are presented elsewhere, along with a description of our direct solver which is based on a non-smooth Newton method [5,4]. It is still impractical to solve system (1) directly for very large matrices. Reasonable approximations of Coulomb frictional forces also prove impractical to solve directly and therefore, we have resolved to a split iterative method in which normal forces are computed directly and frictional ones are processed using a GS process. For SPH fluids, we have used both GS and preconditioned Conjugate Gradient solvers with success [1] to produce very near incompressibility as well as mixed simulation involving boundary conditions and buoyancy.

The H matrix for coupled system has the form of nested dissection shown in Fig. 1 where the blocks, from left to right, “fluid”, “rocks” and “vehicle” each have the matrix form shown in Eqn. (1). For the latter, examples of the sparsity pattern of the H matrix are shown in Fig. 1 for each of the subsystems (20.000 fluid particles, 30 rocks and 12 machine parts). The “contact” blocks involve pairwise constraints couplings between elements of the subsystems and consist of relatively few equations. The fluid matrix in Fig. 1 scales linearly with the number of particles. This is because each particle brings in a fixed number of element per row, around 25 for the SPH kernel we use. For the rigid rock model, each body brings a number of contacts (at worst 20) related to the shape of the geometry. A good di-

rect solver such as SuperLU [6] scales linearly for such problems. Note that both the rock pile and the loader are overdetermined.

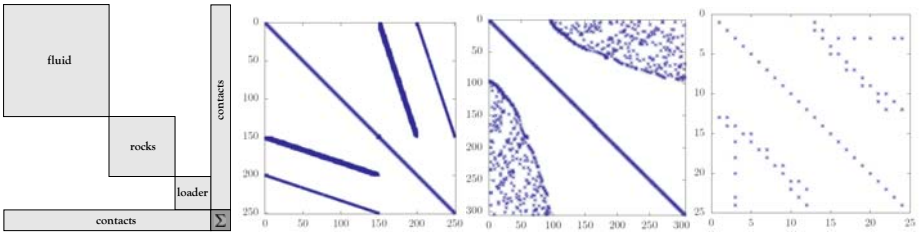

Fig 1. Schematic illustration of the system matrix *H* and its sparsity patterns.

The dynamic resolution algorithm is illustrated in Fig. 2. Contacting bodies are merged into rigid *super-bodies*, the filled geometries, if the relative velocity at the contacts are below a given threshold and contact forces are all sufficiently positive. A merge event is marked in the figure with a dashed line. Conversely, any geometry exposed to an impacting, sliding or separating contact is split from the super-body it belongs to. Split and merge events are propagated using GS iterations ordered as a breadth first traversal starting at the transition points. Splits and merge operations are recursive so that meta bodies can be agglomerated and split. Elementary bodies are atomic and cannot be fractured. The merge-split operations preserve the important invariants, namely, total mass, inertia, momentum, angular momentum. The energy is preserved up to the accuracy implicitly set by the thresholds.

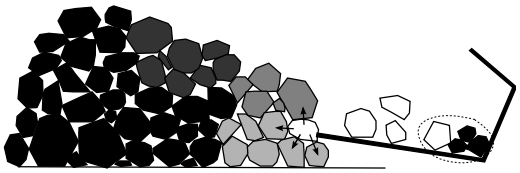

Fig 2. Illustration of the dynamic resolution technique.

THEORY AND METHODS

Still frames from our earth moving training simulator are shown in Fig. 3. On a laptop with 64 bit 3.03 GHz processor and 4 Gb RAM a single simulation with a pile of 500 rocks and a tractor made of 12 bodies and 15 joints, the computational

time was between 2-3.5 s per 1 s of simulation time in single thread mode. The variation depends on the amount of interaction between the vehicle and the pile. The time step is 1/60 ≈ 0.017 s, and the rocks are represented as composite spheres with mean size of 0.6 m, and realistic mass density. The numerical solver takes roughly 75% of the time and collision detection dominates the remaining time. Using dynamic resolution, the computational time per simulated second is reduced to 0.1 s when the pile is at rest, 0.8 when running the load bucket into the pile and interacting with 150 bodies and 1.6, or when driving over the rocks with the vehicle. For these examples, the split-merge procedures take up to 5% of the computational time while it improves the speed by 100-2000%, depending on the amount of interaction.

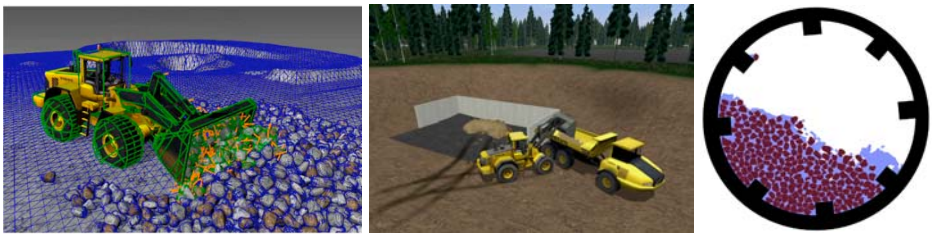

Fig 3. Image from prototype system, final wheel loader simulator and samples from simulation involving a mix of rigid particles and SPH fluid.

The implementation has room for further optimization and acceleration. It is possible for instance to use graphical processing units (GPGPU) to perform some of the iterative computations, part of work in progress. A detailed analysis of GPGPU for these systems and different combinations of direct and iterative solvers will be presented elsewhere. A two-dimensional example of coupling between our SPH based nearly incompressible fluid and immersed pebbles is shown in Fig. 3. For a 3D configuration, not shown here, our GS based implementation on the laptop described above can handle 1000 particles in real-time [1]. Preliminary results show roughly a factor 100 speedup when applying a parallel conjugate gradient solver on GPGPU, i.e., 100.000 particles in realtime. The particular 2D example in Fig. 3 has 222 rigid particles, 861 non penetration contacts and 1235 fluid elements was run with time step 10 ms and with computational time 100 ms per time step, still on the same laptop. With iterative solvers and parallelization on streaming processors the computational speeds for constraint based

methods are steadily improving. For instance, Tasora and Negrut [9] report 1000 bodies at 0.43 s per 0.01 s time step in the serial version and a speed-up by a factor seven to 0.06 s per time step when running on GPGPU.

ACKNOWLEDGMENTS

This research was supported by HPC2N, SSF grant A3 02:128, UMIT Lab and ProcessIT. Contributions were also made by N. Melin at Algoryx and K. Bodin.

REFERENCES

1. K. Bodin, C. Lacoursière, and M. Servin, *Constraint fluids*, submitted to IEEE Transactions of Visualization and Computer Graphics, (2010).
2. R. W. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem*, Computer Science and Scientific Computing, Ac. Press, New York, 1992.
3. E. Hairer, C. Lubich, and G. Wanner, *Geometric Numerical Integration*, vol. 31 of Spring Series in Computational Mathematics, Springer-Verlag, 2001.
4. C. Lacoursière, *Ghosts and Machines: Regularized Variational Methods for Interactive Simulations of Multibodies with Dry Frictional Contacts*, PhD thesis, Dep. of Computing Science, Umeå University, Sweden, June 2007.
5. C. Lacoursière, *Regularized, stabilized, variational methods for multibodies*, in The 48th Scandinavian Conference on Simulation and Modeling, P. Bunus and C. Führer, eds., Linköping University Electronic Press, 2007, pp. 40–48.
6. X. S. Li, *An overview of superlu: Algorithms, implementation, and user interface*, ACM Trans. Math. Softw., 31 (2005), pp. 302–325.
7. T. Pöschel and T. Schwager, *Computational Granular Dynamics, Models and Algorithms*, Springer-Verlag, 2005.
8. A. Tasora and M. Anitescu, *A convex complementarity approach for simulating large granular flows*, Journal of Comp. and Nonlinear Dynamics, (2010).
9. A. Tasora and D. Negrut, *Parallel algorithm for solving complex multibody problems with streaming processors*, Int. J. Comp. Vis. and Biomech., (2009).