RIGID-BODY SIMULATION USING GAMING ENGINES

A study on the usability of gaming industry software for simulating engineering problems.

J.H.H. de Jong

Master of Science Thesis

Delft University of Technology

Precision & Microsystems Engineering

RIGID-BODY SIMULATION USING GAMING ENGINES

A study on the usability of gaming industry software for simulating engineering problems.

Master of Science Thesis

For the degree of Master of Science in Mechanical Engineering at Delft University of Technology

J.H.H. de Jong

June 12, 2014

Faculty of Mechanical, Maritime and Materials Engineering (3mE) · Delft University of Technology

The work in this thesis has been caried out in cooperation with the European Space Agency.

PME - the Ultimate in Mechanical Engineering

Delft University of Technology
Department of
Precision & Microsystems Engineering (pme)

The undersigned hereby certify that they have read and recommend to the Faculty of Mechanical, Maritime and Materials Engineering (3mE) for acceptance a thesis entitled

RIGID-BODY SIMULATION USING GAMING ENGINES

by

J.H.H. de Jong

in partial fulfillment of the requirements for the degree of

Master of Science Mechanical Engineering

Dated: June 12, 2014

Supervisor(s):

ir. K. Wormnes

dr.ir. P. Tiso

Reader(s):

prof.dr.ir. A. van Keulen

dr.ir. P.T.L.M. van Woerkom

ir. M.V. van der Seijs

Table of Contents

Preface

As cliché as it might sound, one of my first memories is actually visiting the European Space Agency's on-site museum, Space Expo, with my grandmother Lia and cousin Han. I was truly captivated by the images of astronauts bobbing around in their clumsy but oh so shiny white space suits, the amazing colors of the numerous pictures of constellations, planets and the earth and the technological achievements of the engineers that made it all possible.

Apart from space exploration, classical mechanics has captivated me also, ever since I learned the basic principles in high school. What attracted me the most, is that it presented challenging puzzles describing the motion of objects, with very tangible outcomes.

Needless to say, I was of course overjoyed to get a position to do this work at ESA, combining two fields of interest of me.

I would not have been able to finish this work on my own, and therefore I would hereby like to acknowledge the people that have helped me greatly during this project. First I would like to acknowledge the people at ESA for coming up with the project and facilitating, guiding and helping me with experiments, especially Kjetil Wormnes, Gianfranco Visentin, Tim Wiese, Robin Nelen, Carlos Crespo (Automation & Robotics section, ESTEC Noordwijk) and Gianluigi Baldesi (Mechanisms section, ESTEC Noordwijk).

Second I would like thank my supervisor from the university, Paolo Tiso (Department of Precision & Microsystems Engineering, Faculty of 3mE, Delft University of Technology). He has helped me in giving direction to my work, and he supported and motivated me when the going got tough.

I thank Erwin Coumans (Google, Bullet Physics) and Kenny Erleben (Department of Computerscience, University Copenhagen). Their work has been the starting point for my research, and I thank them for personally taking the time to answer my questions.

And last but certainly not least, I would like to thank my friends, family and girlfriend for supporting me through this sometimes challenging period. I thank my parents in particular, for providing me with everything I needed to get to this point in my life.

Delft, University of Technology J.H.H. de Jong
June 12, 2014

Chapter 1

Conclusions & recommendations

In section 1-2 the research objectives were defined as follows.

- Identify the underlying mathematical methods of gaming engines and assess their accuracy.
- Determine what would be the advantages of using a gaming engines over traditional software.
- Define what applications would be well suited to be simulated with gaming engines.
- Propose a possible implementation for an engineering oriented simulator based on gaming engines.

This chapter will attempt to summarize our findings. The first section will cover the first three questions, and the second section discusses the fourth question.

1-1 Conclusions

In chapters 2 and 4 we have shown that the main difference between gaming engines and their engineering counterparts is their focus on real-time applications. This results in the need for simulations with a fixed integration time-step size, which cannot be combined with penalty-based contact models, the latter being a common method for engineering tools to handle rigid-body contact. Instead, gaming engines typically use the Stewart-Trinkle constraint-based method, for handling contacts.

Constraint-based simulation requires either the velocity or the position at the next time-step to be explicit in terms of the forces acting on the bodies. Therefore it is often combined with the semi-implicit Euler integration scheme. This is a symplectic integrator, meaning that it more or less conserves energy. It is shown that the error arising from numerically solving the Newton-Euler equations using this method, combined with a fixed time-step, scales with $\mathcal{O}(h)$ for a free falling object. For a simple harmonic oscillator the error in the amplitude and period scale with $\mathcal{O}(h^2)$, as long as the time step is smaller than a value related to the natural frequency of the oscillator.

The Stewart-Trinkle method is based on a maximal coordinate formulation. Combined with constraints and numerical integration this formulation generally suffers from stability errors such as joint-drift. This can be solved by adding a constraint stabilization method. Several methods are present, of which Baumgarte is the most simple. However, combining Baumgarte with Stewart-Trinkle results in energy being created in collisions. Post-stabilization seems to be a better alternative, because decreasing the time-step size will in this case yield convergence.

Different levels of friction approximation exist, each with their own deficiencies. When the linearized friction model presented in section 4-4-1 is applied, the exact solution of Coulomb's friction model can be approached by selecting a large number of vectors to span the friction plane. The decoupled friction model presented in section 4-4-2 suffers from overestimation of the static friction force.

The type of friction model that is selected may limit the types of solver that can be used to solve the resulting complementarity problem. Different types of solvers exist, each with their own convergence and computational efficiency characteristics. However, no matter how accurate the friction model and the solution to the complementarity problem, uniqueness is never guaranteed, due to the statistical undetermined nature of systems of rigid-bodies.

Based on all this, we can conclude that depending on the combination of methods that is used, gaming engines can provide accurate solutions to problems involving classical mechanics and rigid-bodies. Generally selecting more accurate methods will yield a more computationally expensive simulation, but this is not true for all cases.

The main advantages of using gaming engines for particular problems are their stability and computational efficiency in handling contact between multiple rigid-bodies. In chapter 10 it was shown that Bullet as a showcase for gaming engines performs several orders of magnitude faster than MSC/Adams when simulating a simple simulation with several rigid-bodies in contact. However, stiff systems and in particular damped systems are not handled with the same ease as is the case in MSC/Adams. A possible explanation for this is that Bullet does not vary the time-step based on the stiffness and harmonic behavior of the system.

In their current form, gaming engines are particular useful when real-time operation is required. The best known example of this is a simulator. Furthermore gaming engines are very useful when simulating a large number of contact rigid-bodies, or simulations where there is a lot of static contact (i.e. objects lying on each other for a large part of the simulation).

1-2 Recommendations

Based on this research we can state that the simulation of space applications can significantly benefit from the use of gaming engines. However, creating a purpose built rigid-body simulator from scratch would require a considerable amount of effort, and the same goes for maintaining it and keeping it up to date.

Bullet & Blender already have a lot of functionality and would be good candidates for adaptation by the engineering community. The reason for this is that they are open-source and therefore flexible and have a strong ongoing development, and hence are constantly being updated with latest advancements in rigid-body simulation.

However, in its current integration in Blender, Bullet shows unphysical behavior in some cases (e.g. friction modeling, elastic collisions). According to Bullet's development team, most of these shortcomings have either already been fixed but are not accessible from Blender, or will be fixed in newer versions of Bullet.

If an engineer were to use Bullet or any other gaming engine, control over the types of algorithms used is preferred. For instance, some simulations might require more accurate friction modeling, which would lead to the linearized friction model and a complementarity problem solver based on the Newton method. However, if the focus is more on speed, and friction accuracy is less stringent, one might prefer using the decoupled friction model with a Projected Gauss-Seidel (PGS) solver. Therefore it is the author's opinion that a tool that allows more control over the gaming engine would be a great asset.

Also additional functionality could be included through extensions of Blender. Blender lacks any post-processing environment. Although everything can be exported to external data files, a more user friendly way to view simulation results would be a great addition.

A module that tracks the total energy of the system could be of use, just as having way to extract reaction forces from the simulation. However, one must always be careful when interpreting these, especially with unilateral constraints. Finally a function to control and read the error in the complementarity problem would be a big improvement.

1-3 Further research

One thing that, to the author's best knowledge, has not been investigated much, is the use of adaptive time-stepping for stiff systems in combination with the Stewart-Trinkle method. If this is possible, it might considerably improve the handling of stiff and damped systems by the semi-implicit Euler integration, and could lead to a tool that combines the best of both worlds.

Also, the experimental verification of the net and string should be taken further. More complex, asymmetric excitations of the string were not represented correctly in the simulation. This is perhaps due to the fact that the model does not account for any bending stiffness in the string, which in reality is present. The chaotic nature of the net experiment setup made it very hard to truly verify the accuracy of the simulation method. A better experiment, with a more simple and consistent geometric arrangement would improve the correlation between simulation and experiment.

Bibliography

[1] SIMPACK AG, "Simpack multi-body simulation software." http://www.simpack.com.

[2] K. Wormnes, J. de Jong, and G. Visentin, "Throw-nets and tethers for robust space debris capture," in *64^{th} International Astronautical Congress, Beijing, China*, 2013.

[3] M. Software, "Adams." http://www.mscsoftware.com/product/adams.

[4] E. Coumans, *Bullet 2.80 Physics SDK Manual*, 2012.

[5] B. Nguyen and Jef, "dvc3d: a three dimensional physical simulation tool for rigid bodies with contacts and coulomb friction," in *The 1^{st} Joint International Conference on Multibody System Dynamics, Lappeenranta, Finland*, 2010.

[6] A. Price, "Introduction to rigid-body simulations."
http://www.blenderguru.com/videos/quick-tutorial-make-a-wrecking-ball-with-rigid-body-physics/comment-page-1/. Accessed on March 2014.

[7] NVidia, "Popular physics engines comparison."
www.physxinfo.com, December 2009.

[8] G. Hippmann, "An algorithm for compliant contact between complexly shaped surfaces in multibody dynamics," in *Multibody dynamics* (J. A. Ambrósio, ed.), (Lisbon), IDMEC/IST, July 2003.

[9] NVidia, "Physx." www.physicsinfo.com.

[10] Intel, "Havok physics." http://www.havok.com/products/physics.

[11] R. Smith, "Open dynamics engine." http://www.ode.org/.

[12] J. Jerez and A. Suero, "Newton game dynamics." http://www.newtondynamics.com/.

[13] N. Koeng and J. Polo, "Gazebo, 3d multiple robot simulator with dynamics." http://gazebosim.org/.

[14] Cyberbotics, “Webots 7.” http://www.cyberbotics.com/.

[15] K. Kapellos, “Planetary exploration missions simulation using 3drov,” in *Euromech colloquium on "Nonsmooth con- tact and impact laws in mechanics", Grenoble, France*, 2011.

[16] J. Bender, K. Erleben, J. Trinkle, and E. Coumans, “Interactive simulation of rigid body dynamics in computer graphics,” in *Eurographics* (M. Cani and F. Ganovelli, eds.), (Cagliari, Italy), May 2012.

[17] P. Lötstedt, “A numerical method for the simulation of mechanical systems with unilateral constraints,” *Department of Numerical Analysis and Computing Science, the Royal Institute of Technology, Stockholm, Sweden*, 1979.

[18] P. Lötstedt, “Numerical simulation of time-dependent contact and friction problems in rigid body dynamics,” *SIAM Journal of Scientific Statistical Computing*, vol. 5, pp. 370–393, 1984.

[19] D. Baraff, “Fast contact force computation for nonpenetrating rigid bodies,” *Computer Graphics*, vol. 28, pp. 23–24, 1994.

[20] D. Stewart and J. Trinkle, “An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction,” *International Journal of Numerical Methods in Engineering*, 1996.

[21] M. Anitescu and F. Potra, “Formulating dynamic multi-rigid-body contact problems with friction as solvable linear complementary problems.,” *Nonlinear Dynamics*, vol. 14, pp. 231–247, 1997.

[22] K. Erleben, *Stable, Robust, and Versatile Multibody Dynamics Animation.* PhD thesis, University of Copenhagen, Denmark, 2005.

[23] K. Erleben, J. Sporring, K. Henriksen, and H. Dohlmann, *Physics based animation.* Boston, Massachusetts: Cengage Learning, 2005.

[24] J. Sauer and E. Schömer, “A constraint-based approach to rigid body dynamics for virtual reality applications,” *ACM Symposium on Virtual Reality Software and Technology*, pp. 153–161, 1998.

[25] J. Baumgarte, “Stabilization of constraints and integrals of motion in dynamical systems,” *Computer Methods in Applied Mechanics and Engineering*, vol. 1, pp. 1–16, 1972.

[26] M. B. Cline and D. K. Pai, “Post-stabilization for rigid body simulation with contact and constraints,” in *IEEE Intl. Conf. on Robotics & Automation*, vol. 3, (Vancouver, BC), 2003.

[27] A. Cromer, “Stable solutions using the euler approximation,” *American Journal of Physics*, vol. 49, pp. 455–459, 1981.

[28] D. Donnely and E. Rogers, “Symplectic integrators: An introduction,” *American Journal of Physics*, vol. 73, pp. 938–945, 2005.

[29] J. Damkjaer and K. Erleben, "Gpu accelerated tandem traversal of blocked bounding volume hierarchy collision detection for multibody dynamics," in *6th Workshop on Virtual Reality Interactions and Physical Simulations*, vol. 1, (Karlsruhe, Germany), pp. 115–124, 2009.

[30] J. L. Morales, J. Nocedal, and M. Smelyanskiy, "An algorithm for the fast solution of symmetric linear complementarity problems," *Numerische Mathematik*, vol. 111, pp. 251–266, 2008.

[31] M. Silcowitz, S. Niebe, and K. Erleben, "Interactive rigid body dynamics using a projected gauss-seidel subspace minimization method," *Computer Vision, Imaging and Computer Graphics. Theory and applications*, pp. 218–229, 2011.

[32] C. G. F. Pfeiffer, *Multibody dynamics with unilateral contacts.* New York: Wiley&Sons, 1996.

[33] R. Featherstone, *Robot Dynamics Algorithms.* Kluwer Academic Publishers, Boston/-Dordrecht/Lancaster, 1987.

[34] A. Hanson, "Visualizing quaternions," in *SIGGRAPH*, 2005.

[35] R. Cottle, J.-S. Pang, and R. E. Stone, *The Linear Complementarity Problem.* Boston, Massachusetts: Academic Press, 1992.

[36] C. Lemke, "Bimatrix equilibrium points and mathematical programming," *Management Science*, vol. 11, pp. 681–689, 1965.

[37] M. Silcowitz, S. Niebe, and K. Erleben, "Nonsmooth newton method for fischer function reformulation of contact force problems for interactive rigid body simulation," *The Visual Computer*, vol. 26, pp. 893–901, 2010.

[38] R. van der Linde and A. Schwab, "Multibody dynamics B." Lecture Notes, Delft University of Technology, 2013.

[39] O. Mangasarian, "Linear complementarity problems solvable by a single linear program," *Mathematical Programming*, vol. 10, pp. 263–270, 1976.

[40] Sony. http://pro.sony.com/bbsc/ssr/product-NEXFS700UK/. Accessed on April 2014.

[41] F. Alam, S. Watkins, and A. Subic, "The aerodynamic forces on a series of tennis balls," in *15th Australasian Fluid Mechanics Conference, The University of Sydney, Sydney, Australia*, December 2004.

Glossary

List of Acronyms

PGS	Projected Gauss-Seidel

List of Symbols

α	Baumgarte stabilization parameter
$\boldsymbol{\omega}$	Rotational velocity vector
$\boldsymbol{\sigma}$	Slack variable with no physical meaning
$\delta_{ij}\bullet$	Entry-wise subtraction of two vectors related to masses i and j
ϵ	Coefficient of restitution or strain
η	Number of vectors used to approximate the the friction cone
γ	Slack variable that can be interpreted as approximation of sliding velocity
λ_c	Lagrange multiplier for inequality constraints
λ_e	Lagrange multiplier for equality constraints
$\lambda_{f,i}$	Lagrange multiplier for friction impulse
μ	Friction coefficient
ω	Natural frequency of harmonic oscillator
ω_e	Element natural frequency
$\phi(\boldsymbol{s})$	Bilateral/equality constraint
$\psi(\boldsymbol{s})$	Unilateral constraint
ρ	Radius
ρ_{air}	Density of air
ρ_{lin}	Cord linear density
σ	Normal stress
ζ	Critical damping ratio
ζ	Slack variable with no physical meaning

$\boldsymbol{A}$	Complementarity problem matrix
$\boldsymbol{b}$	Complementarity problem vector
$\boldsymbol{D}$	Friction cone linearization matrix or diagonal matrix
$\boldsymbol{E}$	All-ones matrix used for friction model
$\boldsymbol{f}_{\text{contact}}(\boldsymbol{s},\boldsymbol{u})$	Contact force vector
$\boldsymbol{f}_{\text{ext}}(t,\boldsymbol{s},\boldsymbol{u})$	External force vector
$\boldsymbol{F}_{d,ij}$	Spring force between masses i and j
$\boldsymbol{F}_{D,i}$	Aerodynamic drag force acting on mass i
$\boldsymbol{F}_{s,ij}$	Spring force between masses i and j
$\boldsymbol{G}$	Constraint Jacobian without kinematic map
$\boldsymbol{I}$	Inertia tensor
$\boldsymbol{J}_c$	Jacobian of inequality constraint
$\boldsymbol{J}_e$	Jacobian of equality constraints
$\boldsymbol{L}$	Strictly lower diagonal matrix
$\boldsymbol{M}$	Generalized mass matrix
$\boldsymbol{p}_c$	Contact impulse
$\boldsymbol{p}_e$	Equality constraint impulse
$\boldsymbol{p}_f$	Friction impulse
$\boldsymbol{q}$	Unit quaternion
$\boldsymbol{r}$	Position vector
$\boldsymbol{s}$	Generalized position vector
$\boldsymbol{S}(\boldsymbol{s})$	Kinematic map
$\boldsymbol{U}$	Strictly upper diagonal matrix
$\boldsymbol{u}$	Generalized velocity vector
$\boldsymbol{v}$	Translational velocity vector
$\boldsymbol{w}$	Complementarity problem residue vector
$\boldsymbol{x}$	Complementarity problem solution vector
$\boldsymbol{x}_l$	Complementarity problem lower boundary
$\boldsymbol{x}_u$	Complementarity problem upper boundary
$\mathcal{T}$	Kinetic energy
$\mathcal{V}$	Potential energy
a	Relative velocity orthogonal to contact plane
A_h	Cord cross-sectional area perpendicular to horizontal direction
b_B	Complementarity problem vector term related to Baumgarte stabilization
b_e	Complementarity problem vector term related to elastic collisions
C_D	Drag coefficient
c_e	Element damping coefficient
d	Cord diameter
d_{tb}	Tennis ball diameter
E	Young's modulus

g	Gravitational acceleration
h	Time-step size
k	Number of contact points
k_{erp}	Error reduction parameter
k_e	Element stiffness
l_e	Element natural length
l_n	Cord natural length
l_t	Cord length under pretension
m	Number of equality constraints
m_e	Element mass
m_{tb}	Tennis ball mass
N	Number of mass elements
$\bullet'$	Denotes parameters related to the probing experiment
$\hat{\bullet}$	Denotes unit vector