Lappeenranta
University of Technology

Simo Nikula

HANDLING OF PLASTICITY IN PHYSICS ENGINE

Thesis for the degree of Doctor of Science (Technology) to be presented with
due permission for public examination and criticism in the Auditorium ...
at Lappeenranta University of Technology, Lappeenranta, Finland on the
..th of May, 2017, at noon.

Acta Universitatis
Lappeenrantaensis XXX

Supervisor Professor Timo Björk
Laboratory of Steel Structures
School of Energy Systems
Lappeenranta University of Technology
Finland

Reviewers Title Name
Department of
University of
Country

Title Name
Department of
University
Country

Opponent PhD Name
Institution
Country

ISBN 978-952-265-XXX-X
ISBN 978-952-265-XXX-X(PDF)
ISSN 1456-4491
Lappeenrannan teknillinen yliopisto
Yliopistopaino 2017

Abstract

Simo Nikula
Handling of plasticity in physics engine
Lappeenranta 2017
24 pages
Acta Universitatis Lappeenrantaensis XXXXXX
Diss. Lappeenranta University of Technology
ISBN 978-952-265-XXX-X, ISBN 978-952-265-XXX-X(PDF), ISSN 1456-4491

In this work suitable methods for handling of plasticity in physics engine are presented. Handling of plasticity basically means that objects involved may absorb energy and possibly break.
Methods are verified by extending Bullet Physics Library.
Keywords: Physics engine, simulation, dynamics, plasticity, inelastic, multibody dynamics, physical simulation, deformation.

Acknowledgements

This study was carried out in the School of Energy Systems at Lappeenranta University of Technology, Finland.
Thanks.

Simo Nikula
May 2017
Lappeenranta, Finland

To all of you,
use freely

Yours, Simo

Contents

1 Introduction

This work tries to invite structural engineers to take active part in development of more realistic physis engines. Current computer systems provide enough computing performance and efficient algorithms so collision detection and rigid body dynamics can be handled in real time in most scenarios. Soft body dynamics in which object are deformable is in many cases still too complicated for real time applications. Finite element method will probably replace other methods by providing general solution at some time but not in very near future. In this work focus is on method where rigid bodies are connected by constraints.

Figure 1.2 summarizes branches involved in this work.

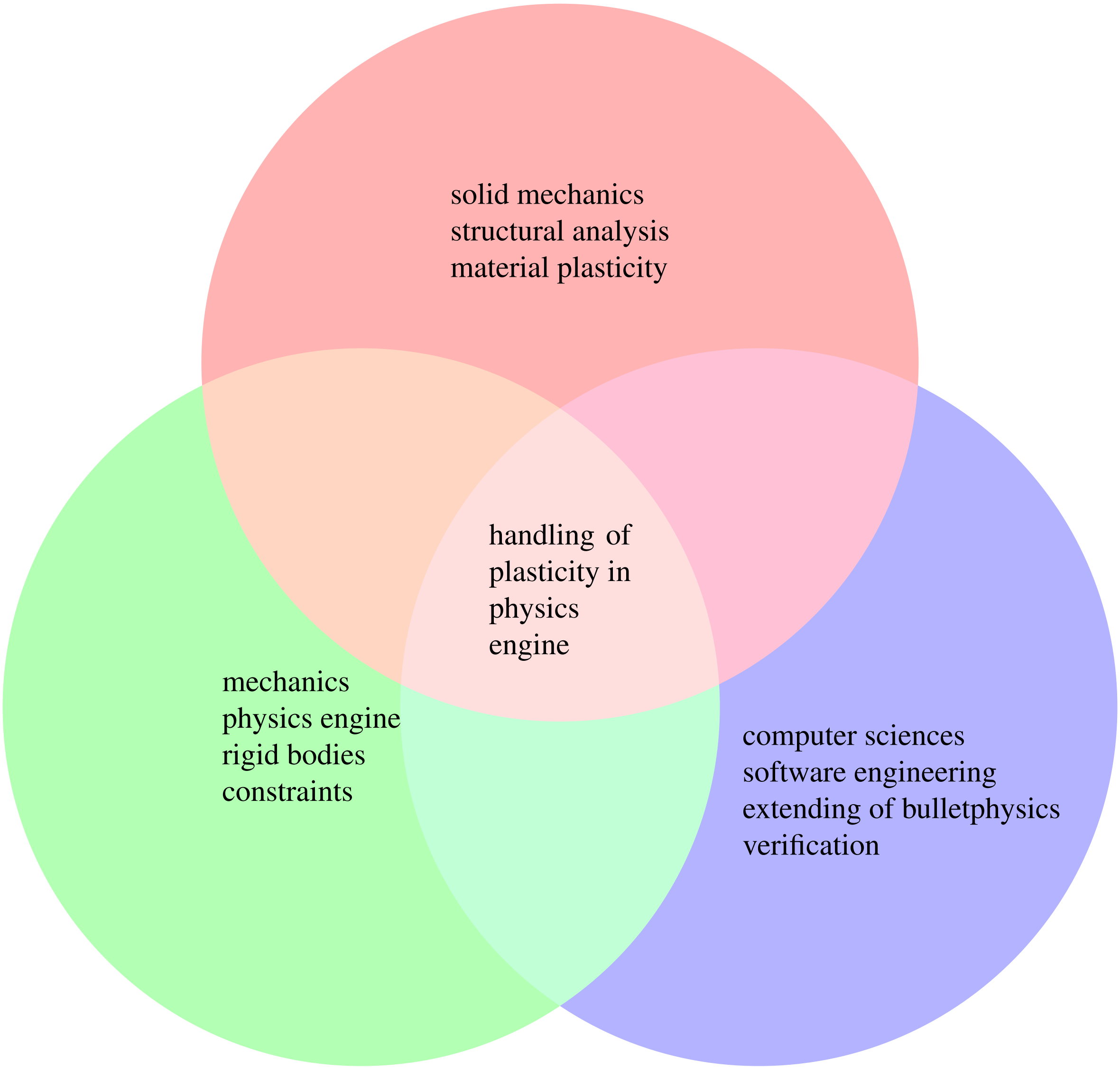

Figure 1.1: Multidisciplinary approach to plasticity in physics engine.

1.1 Structural analysis

Structural analysis for advanced structures is nowadays typically done using finite element method and analysis time is typically hours. Finite element method is already used in physics engines and usage will come more widespread in future as more computing capacity and memory will be available.
In this work all theories in area of structural analysis are quite old and e.g. Timoshenko (1940) is typical reference. Main focus is on elastic and linear plastic parts of stress-strain curve and elastic and plastic section modulus.
Typical stress strain curve of ductile steel is shown in 1.2. Stress-strain curve is not drawn to scale as elastic strain could not be seen as it is typically 0.001 to 0.005. Straing hardening is taken into account mainly by assuming that plasticity in bending expands. This can be seen e.g. by bending paperclip. It does not break at low angles but can take few full bends.

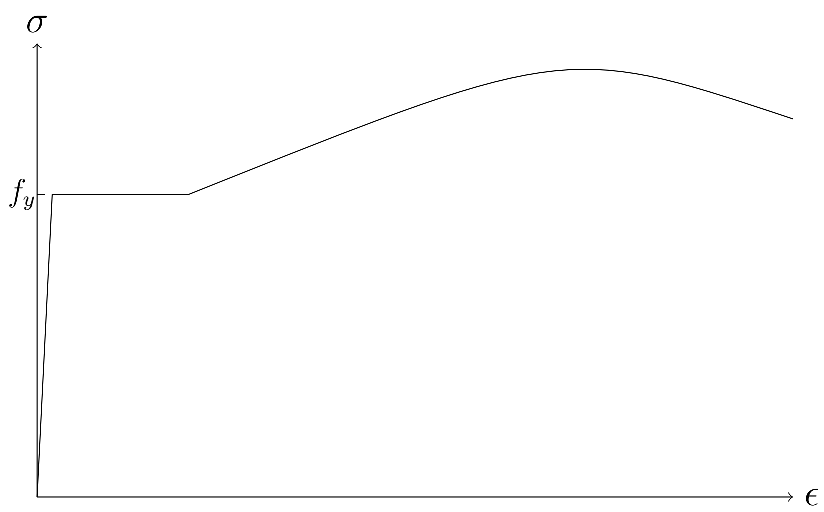

Figure 1.2: Stress-strain curve of ductile steel (not to scale).

Difference between elastic and plastic section modulus is shown in 1.3.
If stress is below yield limit, stress and strain are linear within cross section. If cross section is fully plastic, stress is assumed to be at yield level over whole cross section and so plastic section modulus is higher than elastic section modulus.
Elastic part is often ignored in this work as displacements due to to elastic deformation are small and related frequencies are high.
E.g. Camp (2015) is good example of university cource providing needed information about plasticity.

1.2 Physics engine

Physics engines is active research area on many research and industry areas. Especially gaming and movie industry are powerful drivers. Robotics is typical research areas that

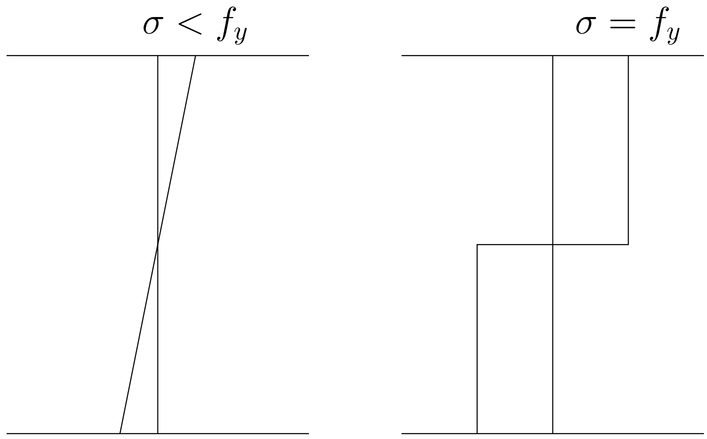

Figure 1.3: Stress distribution under elastic and plastic loads.

uses physics engines. In Lappeenranta University of Technologysimulation of working machines has been active research area, e.g. Moisio (2013).
Physics engines cannot used in serious structural analysis but using structural analysis in physics engines is not rare. E.g. O'Brien and Hodgins (1999) used finite element model to analyze crack initiation and propagation. Already Terzopoulos and Fleischer (1998) used quite similar principles as in this work to handle plasticity.
Erleben (2005) provides extensive introduction to physics engines. He uses term Physics-based Animation and does detailed analysis on many areas like

- preferring fast solutions rather than exact solutions
- various methods for solving differential equations
- processing of constraints

E.g. University (2015a) and University (2015b) are good examples of university cources about physics engines. They also provide additional resources on used technologies.
Few physics engines which have public source

- Bullet Physics- Coumans (2003-2016) which is used e.g. in Blender and Erwin Coumans has received Tehnical Academic Award for creation of the Bullet physics library, Oscar (2015)
- Box2D - Catto (2007-2015) which is used e.g. in Angry Birds
- RigsOfRods - Ricordel (2005-2015) which uses large amount of rod elements to simulate deformation and breaking of vehicles

1.3 Software engineering

Main requirements for software implementation can be summarized

- plastic deformation takes place if force or moment exceeds given limit
- plastic deformation absorbs enegry
- after predefined limit of displacement is exceeded constraint is deactivated
- results should be in same magnitude as theoretical and test results
- solution can be integrated to existing dynamics engines
- additional calculation time for each timestep may not exceed few milliseconds

Few decades ago software engineering and structural analysis were highly coupled and e.g. Bathe (1982) and Cook (1981) contain very detailed instructions on computer implementation. Those books provided good baseline for many computer standalone programs created in Lappeenranta University of Technologyfor for analysis of thin-walled structures Halme (1995); Nikula and Halme (1996a,b). Getting and compiling complicated software packages is currently quite easy. Understanding and getting enough knowledge to extend them requires a fair amount of hard work.
Using calculation methods that have initially planned to be hand calculated gives good starting point for getting fast processing. Lack of clean boundary conditions will require additional logic or accepting larger errors.

1.4 Motivation and objectives of the study

Target of this work is to find out suitable methods for handling of plasticity and verify them in commonly used physics engine Bullet Physics. Handling of plasticity basically means that objects involved may absorb kinetic energy and break.
There are two major areas of applications for these new features.
Possibility to provide simulated demostrations and verifications of new theories or exercises would be targeted for studies of mechanics and strength of material.
Methods presented in this work can be used in gaming industry to provide objects that behave more realistically than currently without significant extra work. Efficient simulation algorithms can also help in saving energy. Saving even few wats or percent for single user makes big difference as gaming PCs are estimated to have consumed 75 TWh/year of electricity globally in 2012, Mills and Mills (2015).

1.5 Scope

In this work one solution to handling of plasticity is presented and verified with few examples.

1.6 Scientific contribution

This work opens wide area of further work of combining structural analysis and plasticity with physics engines. Few examples are shown in sections **??** and possible areas of further work are discussed in section 4.1.

2 Simulation of Charpy impact test using Bullet Physics

In following sections various methods for handling plasticity are described. As physics engines have not been traditional area of research for plasticity, well known Charpy impact test was selected to be used as primary scenario although it is against first two tips in Bullet PhysicsManual.

- Avoid very small and very larger collision shapes
- Avoid large mass ratios

2.1 Introduction to constraint processing in Bullet Physics

Constraint processing in Bullet Physicsis based on ODE, Smith (2001-2007). Mathematical background and detailed examples are available by Smith (2002). Equations 2.1, 2.2 and 2.3 are created for each constraint.

$$J_1 v_1 + \Omega_1 \omega_1 + J_2 v_2 + \Omega_2 \omega_2 = c + C\lambda \tag{2.1}$$

$$\lambda \geq l \tag{2.2}$$

$$\lambda \leq h \tag{2.3}$$

Main parameters and corresponding fields in Bullet Physics are described in table 2.1.

Parameter	**Description**	**btConstraintInfo2 pointer**
J_1, Ω_1 J_2, Ω_2	Jacobian	m_J1linearAxis, m_J1angularAxis m_J2linearAxis, m_J2angularAxis
v	linear velocity	
ω	angular velocity	
c	right side vector	m_constraintError
C	constraint force mixing	cfm
λ	constraint force	
l	low limit for constraint force	m_lowerLimit
h	high limit for constraint force	m_upperLimit

Table 2.1: Constraint parameters

2.2 Common data

2.2.1 Material

Material is steel. Density is 7800 $\frac{kg}{m^3}$. Young's modulus is 200 GPa. Yield stress is initially 400 MPa but it can be modified. Coefficient of restitution can be modified and has value of 0 (fully plastic) unless otherwise staten.

2.2.2 Coordinate system

X axis is horizontal, positive to direction where hammer comes from (left)
Y axis is vertical, positive up
Z axis is horizontal and Z=0 is symmetry plane

2.2.3 Specimen

Specimen dimensions can modified. Basic measures are 10x10x55 mm with 2 mm notch in middle which is taken into account in calculations. Specimen is positioned symmetrically around z=0, bottom at y=0.2 m and backside at x=0. Expected energy loss is product of plastic moment of section, hinge angle needed for specimen to go through supports and ultimate tensile strength of specimen. For hinge angle of 1.9 radians and yields stress of 400 MPa expected energy loss is 122 J. Mass is about 40 g and in most cases it is modelled as two 20 g items. In laboratory tests variation for similar specimens is roughly about 10 Specimen can be subdivided to multiple parts.

2.2.4 Support anvils

Support anvils initially have 40 mm open space between them. Their width is 40 mm. If specimen bends about 1.9 radians (108 degrees) it will go between anvils. Space between anvils can be changed.

2.2.5 Hammer

Hammer is 0.5 m wide and 0.25 m high, thickness is 0.02 m. Mass is 19.5 kg. Hammer and hinge are positioned so that impact is horizontal (global x-direction). Hammer arm is 40x40x500 mm. Hammer thickness can be changed. Setting it to zero causes hammer not to get added to model. Hammer has modifiable draft. Default value is 0.04 m. Arm and draft are not taken into account for mass and inertia calculation.

2.2.6 Energy calculation

Energy for whole system is calculated so that potential energy and kinetic energy are calculated in updateEnergy. When hammer is resting in low position, energy is 64 J.

2.2.7 Breaking of constraints between objects

For standard Bullet Physicsconstraints breakingImpulseThreshold (BITh) can be defined. If impulse is larger than set limit constraint is activated. This allows object breaking to be simulated with very cheap calculation. This method is too simplified for ductile material and developing more precise and realistic methods is main objective of this work.

2.2.8 Timestep

Usually bullet simulations are done using fixed time step of 1/60 s i.e. 16.67 ms. For this case 17 ms is too large timestep even to keep system stable. For impact time much smaller timestep is needed. Real time simulation was tried as option but was removed from code as it did not provide good results. Default time step was selected to be 5 ms outside impact time and 0.1 ms during impact. Automatic time stepping routine changes timestep so that at angles higher than 0.2 5 ms time step is used and adjusts it linearly to selected time step between angles of 0.2 and 0.05. If specimen is no longer at anvil larger timestep is also used.

2.3 Single rigid body

This is basic reference case without plasticity. Specimen should stop hammer.

2.4 Constraint with zero limits

In this case btGeneric6DofConstraint is used with high and low limits set to zero. Constraint is made breakable by setting breakingImpulseThreshold. This is common technique to provide breakable objects in games and provides visually accetable results for brittle materials.

2.5 SpringConstraint

This shows how btGeneric6DofSpringConstraints act in case like this. Specimen can absorb very little elastic energy so rotational spring constants are calculated using plastic state i.e. $W = bh^2/4$ where h=b-0.002m and yield stress is used instead of Young's modulus for rotational springs.

2.6 Spring2Constraint

This basically same as SpringConstraint but uses newer btGeneric6DofSpring2Constraint which has recently appeared in Bullet Physics. Automatic stiffness limitation helps to avoid instability but may make constraint too soft.

2.7 Hinge constraint with motors

In this case btHingeConstraint is used. Angular motor is enabled with target velocity of 0 and maximum motor impulse is set to be plastic moment multiplied by timestep so that motor resists any rotation.

2.8 PlasticHingeConstraint

In this case new btPlasticHingeConstraint is used. It is modification of btHingeConstraint and was is developed during this work. PlasticHingeConstraint has additional fields for plasticMoment and previousHingeAngle and getAbsorbedEnergy method which returns product of plasticMoment and hingeAngle change.

2.8.1 Maximum plastic rotation

Maximum plastic rotation is new term which was introduced to allow efficient way to express ultimate strain in bending. Implemented constraint has additional fields for maxPlasticRotation and currentPlasticRotation. After each step in which rotation angle changes, absolute value of angle change is accumulated to currentPlasticRotation. As currentPlasticRotation reaches maxPlasticRotation constraint is inactivated and specimen breaks. For ductile behaviour maxPlasticRotation could be set to e.g. 1-6 (radians) and for brittle behaviour to e.g.0.01-0.1 radians. Maxium plastic rotation is basically rotational stretch under bending moment. Basic sample can be tried using paperclip. It can typically handle two bend overs before breaking. Sewing machine needle or tooth pick which has same diameter is much harder to bend and it often breaks before plastic deformation takes place.

2.8.2 Limiting maximum moment in constraint

btPlasticHingeConstraint::getInfo2InternalUsingFrameOffset is modified so that in case of lostop=histop, plasticMoment*timeStep is used as upper and lower limit instead of SIMD_INFINITY.

2.9 ElasticPlasticConstraint

In this case new bt6DofElasticPlasticConstraint is used. It is based on SpringConstraint with some modifications

- internalUpdateSprings is modified so that nonlinear relation between displacements and forces can be defined. If force is smaller than maximum force or moment for given degree of freedom, elastic behavior is handled in similar way as spring works. If force is larger, maximum force is used.
- constraint breaks if active forces are high enough and plastic reserve has been used
- frequency ratio of lowest mode and integration is used to limit spring functionality to avoid stability issues. Ratio defines how many steps are needed for one period. If number of integration steps is too low, spring is modified to ignore elastic part.

2.10 ElasticPlastic2Constraint

This is basically same as elasticPlasticConstraint but uses new bt6DofElasticPlastic2Constraint which is based on btGeneric6DofSpring2Constraint. Get_limit_motor_info2, setAngularLimits and setLinearLimits which are called by getInfo2 are modified. SIMD_INFINITY values are replaced by maximum force values which are transformed to corresponding impulsed by multiplying forces by integration interval. btGeneric6DofSpring2Constraint has option limitIfNeeded for stiffness and damping to avoid issues. Spring is softened so that 4 steps are used for one step. This is not realistic for typical steel structures. In bt6DofElasticPlastic2Constraint plasticity code is activated if frequency is too high for stable solution. If plasticity is activated due to large spring force or due to high frequency forces constraint error is calculated in similar way as for case where upper limit equals lower limit, (m_currentLimit==3).

3 Charpy impact test results using Bullet Physics

Results are quite sensitive and obtained in February 2016 using following processes

- single precision debug build is used
- sequential impulse constraint solver with 10 iterations is used
- energy loss values are rounded to nearest 10 joules

3.1 Results for single rigid body

Timestep of 0.1 ms provides expeced results and specimen bounces hammer back as seen in figure 3.1.

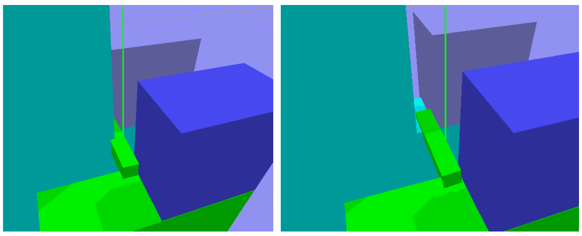

Figure 3.1: Specimen bounces hammer back with timestep of 0.1 ms

Larger timesteps produce various kinds of unrealistic results seen in figure 3.2. In these cases timestep is 17 ms. Specimen penetrates hammer and anvils. Without draft specimen stays on anvil.

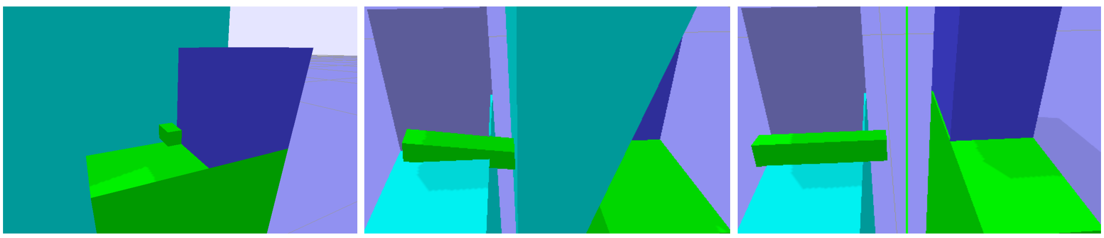

Figure 3.2: Results with timestep of 17 ms

3.2 Results for constraint with zero limits

This technique provides effective and stable way to simulate breaking of brittle material but does not provide realistic results for ductile material as can be seen in figure 3.4. Required energy to break constraint can be controlled using breakingImpulseThreshold.

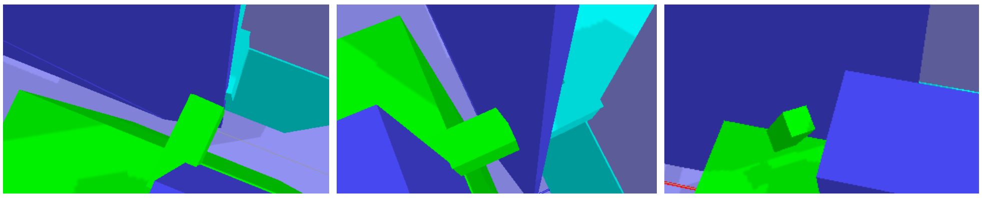

Figure 3.3: Specimen breaks in brittle way

3.3 Results for springConstraint

Energy loss is about 50 J. Visual feedback is not realistic as specimen gets back to initial shape as seen in figure 3.4.

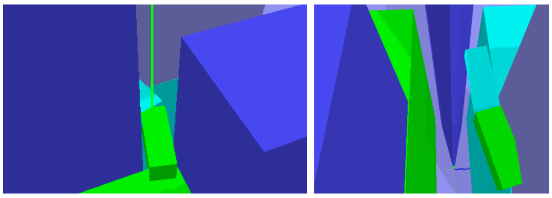

Figure 3.4: Spring constraint brings specimen back to initial shape

Unstable simulation can be seen by setting yield stress to e.g. 800 MPa. Few frames of such simulation can be seen in figure 3.5. Unstable behaviour starts immediately at start of simulation and is triggered without impact from hammer.

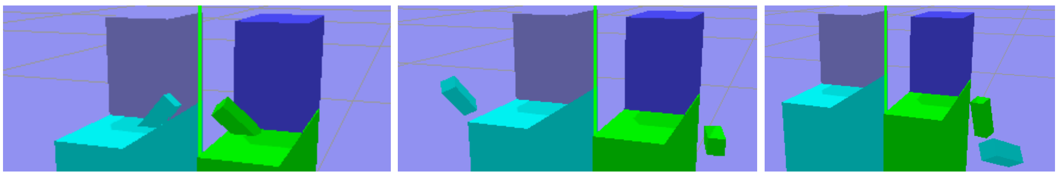

Figure 3.5: Unstable simulation with stiff spring contstaints

3.4 Results for spring2Constraint

Results are quite similar as for springConstraint but automatic parameter tuning prevents unstable simulations. Energy loss is about 40 J.

3.5 Results for hinge constraint with motors

Energy loss is about 110 J. Visual feedback is realistic as specimen keeps deformed shape as seen in figure 3.6.

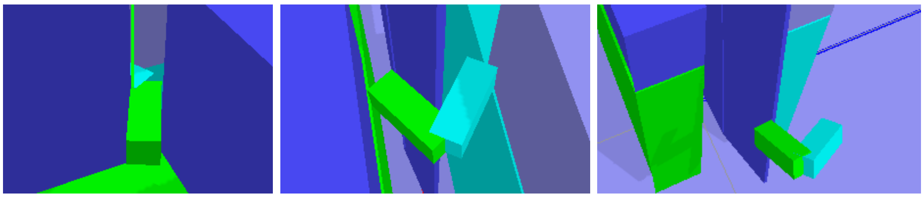

Figure 3.6: Hinge constraint with motors

3.6 Results for plasticHingeConstraint

Results are quite similar as for hinge constraint with motors. Breaking point can be configured based on rotation angle which makes material behaviour more realistic. In figure 3.7 maximum plastic rotation is limited to 1.5 and energy loss is about 95 J.

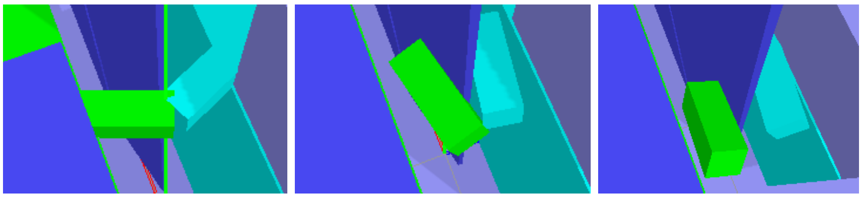

Figure 3.7: PlasticHingeConstraint with maximum plastic rotation set to 1.5

3.7 Results for elasticPlasticConstraint

Results are quite similar as for plasticHingeConstraint as this case is dominated by plastic bending. Additional logic helps to avoid stablity issues.

3.8 Results for elasticPlastic2Constraint

Results are quite similar as for elasticPlasticConstraint and this constraint seems to best candidate for further development. Table 3.1 shows energy loss for few timesteps as example to demonstrate sensitivity of solution.

3.9 Summary and evaluation of results

Suggested methods for handling plasticity are well suited for cases where simulation of ductile failure is desired. It requires only slightly more computational effort than methods that are used for simulation of brittle failures. Hinge constraint with motors method can be used without introducing any additional code to most existing dynamic engine as

Timestep[ms]	Energy loss [J]
0.1001	100
0.1	110
0.099	90
0.08	130
0.05	150

Table 3.1: Energy loss for few timesteps

it is based on widely used motor-codes. Additional features are quite simple to implement and make spring based constraints even more usable than currently by allowing stiff constraints to be used without instability.
Suggested methods cannot at least currently be used if exact results are needed but are well suited for educational and entertainment scenarios.

4 Discussion

4.1 Future work

4.1.1 Implementation using GPU

Bullet Physics can already utilize GPU for many operations. Analysis on possible issues in used interfaces should be done before integrations.

4.1.2 Acceptance to current physics engines

Required work depends on selected dynamics engine. Availability of these features makes similar implementation possible.

- generic spring or motor constraint is already available
- possibility to update constraints after each simulation step

Current implementation is done based on Bullet Physics but it still needs refactoring and testing before it can be accepted to main stream. Interfaces should be reafctored so that handling of additional features like large displacements and material nonlinearity can be implemented without further changes.

4.1.3 Large displacements

Large displacements are typical if plasticity is involved and large displacements can also take place due to rigid body motion and should be taken into account.
Update of constraints can be done in many ways but using callbacks allows clean integration i.e. no changes to solver core are needed. In Bullet Physics there are already three per step callbacks.

- updateActions
- updateActivationState
- btInternalTickCallback

DemolisherDemoalready uses updateActions and that seems good way to do also other updates. Special care should be taken to avoid unnecessary updates.

4.1.4 Material nonlinearity

Update of constraint properties can be done in callbacks in similar way as for large displacements. Material definition should be defined so that stress-strain curve can be defined as array of pairs. This is probably not significant feature for simulations in gaming area but interesting feature for research and teaching area.

4.2 Object subdivision just in time

Just in time activation could avoid extra overhead from multiple objects and constraints. Impacts causing plasticity are usually quite brief and they happen at distinct times. Simulation could allocate processing units for handling these and release them after impact has been processed.

References

Bathe, K.J. (1982). *Finite Element Procedures in Engineering Analysis*. Prentice-Hall Inc. ISBN 0-13-317305-4.

Camp, C. (2015). *CIVL 1101 - Introduction to Strength of Materials*. url: `http://www.ce.memphis.edu/1101/notes/concrete/section_1_strength_of_mat`

Catto, E. (2007-2015). *Box2D A 2D Physics Engine for Games*. url: `//box2d.org`.

Cook, R.D. (1981). *Concepts and Application of Finite Element Analysis*. John Wiley & Sons. ISBN 0-471-03050-3.

Coumans, E. (2003-2016). *Bullet Physics Library*. url: `//bulletphysics.org/`.

Erleben, K. (2005). *Stable, Robust, and Versatile Multibody Dynamics Animation*. Ph.D. thesis. University of Copenhagen.

Halme, T. (1995). *AGIFAP (Advanced Graphical Interactive Fram Analysis Program)*.

Mills, N. and Mills, E. (2015). Taming the energy use of gaming computers. *Energy Efficiency*, pp. 1–18. ISSN 1570-646X, doi:10.1007/s12053-015-9371-1, url: `http://dx.doi.org/10.1007/s12053-015-9371-1`.

Moisio, S. (2013). *A Soft Contact Collision Method For Real-Time Simulation of Triangularized Geometries in Multibody Dynamics*. Ph.D. thesis. Lappeenranta University of Technology.

Nikula, S. and Halme, T. (1996a). *FINSTRIP, User's Manual*.

Nikula, S. and Halme, T. (1996b). *VTB-Program, User's Manual*.

O'Brien, J.F. and Hodgins, J.K. (1999). Graphical Modeling and Animation of Brittle Fracture. In: *Proceedings of ACM SIGGRAPH 1999*, pp. 137–146. ACM Press/Addison-Wesley Publishing Co. url: `http://graphics.cs.berkeley.edu/papers/Obrien-GMA-1999-08/`.

Oscar (2015). *TECHNICAL ACHIEVEMENT AWARDS (ACADEMY CERTIFICATES)*. url: `//www.oscars.org/news/21-scientific-and-technical-achievements-be-h`

Ricordel, P.M. (2005-2015). *Rigs of Rods Soft Body Physics Simulator*. url: `//rigsofrods.github.io`.

Smith, R. (2001-2007). *Open Dynamics Engine*. url: `//ode.org`.

Smith, R. (2002). *How to make new joints in ODE*. url: `//ode.org/joints.pdf`.

Terzopoulos, D. and Fleischer, K. (1998). Modeling Inelastic Deformation: Viscoelasticity, Plasticity, Fracture. *Computer Graphics*, pp. 269–278. doi:10.1145/54852.378522, url: `http://web.cs.ucla.edu/d̃t/papers/siggraph88/siggraph88.pdf`.

Timoshenko, S. (1940). *Strength of Materials*. D. Van Nostrand Company, inc.

University, C. (2015a). *Introduction to Computer Game Development*. url: `//www.cs.cornell.edu/courses/cs3152/`.

University, C. (2015b). *Physically Based Animation for Computer Graphics*. url: `//www.cs.cornell.edu/courses/cs5643`.