Project Chrono:

Modules, Solvers, and Integrators

July 7, 2023

Reference Material

Related to general Project Chrono (existing distribution)

- Project Chrono Reference Manual: https://api.projectchrono.org/manual_root.html
- Project Chrono White Papers: https://projectchrono.org/whitepapers/
- Project Chrono FAQs: https://projectchrono.org/faq/
- Project Chrono API: https://api.projectchrono.org/group__chrono.html

Related to new developments for 3D-printing and concrete

- Chrono-Concrete Readme
- https://github.com/Concrete-Chrono-Development/chrono-concrete#readme

Related to FreeCAD user interface

- Chrono-Preprocessor Readme
- https://github.com/Concrete-Chrono-Development/chrono-preprocessor#readme

Places to Ask Questions

Related to general Project Chrono (existing distribution)

- Project Chono Forum
- https://groups.google.com/g/projectchrono

Related to new developments for 3D-printing and concrete

- Chrono-Concrete Discussions
- https://github.com/Concrete-Chrono-Development/chrono-concrete/discussions

Related to FreeCAD user interface

- Chrono-Preprocessor Discussions
- https://github.com/Concrete-Chrono-Development/chrono-preprocessor/discussions

Modules to Use

FEA, CSL, LDPM

- Chrono core module
- CPU implementation

DEM

- DEM-Engine
- GPU implementation

SPH

- FSI
- GPU implementation

Categories of Solvers

chrono::ChDirectSolverLS

- Base class for sparse direct linear solvers.
- Sparse linear direct solvers. Cannot handle VI and complementarity problems, so it cannot be used with NSC formulations.

chrono::ChIterativeSolverLS

- Base class for Chrono solvers based on Eigen iterative linear solvers.
- Iterative linear solvers. Cannot handle VI and complementarity problems, so they cannot be used with NSC formulations.

chrono::ChIterativeSolverVI

- Base class for iterative solvers aimed at solving complementarity problems arising from QP optimization problems.

LS: Linear Solver
VI: Variational Inequality

All Solvers

chrono::ChDirectSolverLS

• Mumps	MUMPS parallel sparse direct solver
• ParadisoMKL and ParadisoProject	Intel MKL Pardiso or Paradiso Project sparse direct solver
• SparseLU and SparseQR	Eigen SparseLU or Eigen SparseQR direct solver

chrono::ChIterativeSolverLS

• BiCGSTAB	Bi-conjugate gradient stabilized algorithm
• GMRES	Generalized Minimal Residual Algorithm
• MINRES	Minimum Residual Method

chrono::ChIterativeSolverVI

• MulticoreNSC and MulticoreSMC	NSC – Non-smooth dynamics, SMC – Smooth dynamics
• ADMM	Alternating Direction Method of Multipliers
• APGD	Nesterov's Projected Gradient Descent
• BB	Modified Krylov iteration w/ Barzilai-Borwein
• Pjacobi	Projective fixed point method (projected Jacobi)
• PMINRES	Modified Krylov iteration of MINRES w/ gradient projections
• PSOR and PSSOR	Projective fixed point method and symmetric PSOR

Chrono-Recommended Solvers

SOR

- Low precision: convergence might stall, especially with odd mass ratios
- Supports Differential Variational Inequalites (DVI) (hard contacts, with complementarity)
- Used most often for small problems, solution accuracy is not particularly important

APGD

- Very good convergence, used most often for simulations in which high accuracy in results is desired
- Supports DVI (hard contacts, with complementarity)

BARZILAIBORWEIN

- Good convergence
- Supports DVI (hard contacts, with complementarity)
- Similar to APGD, might be more robust when using large mass ratios

MINRES

- Good convergence
- Supports FEA problems
- Does nor support DVI (hard contacts, with complementarity) for the moment.

Categories of Time Integrators

chrono::ChTimestepperIIorder

- Base class for 2nd order timesteppers

chrono::ChTimestepperIorder

- Base class for 1st order timesteppers.

All Time Integrators

chrono::ChTimestepperIIorder

- ChTimestepperEulerExplIIorder: Euler explicit for 2nd order
- ChTimestepperEulerImplicit: Euler implicit
- ChTimestepperEulerImplicitLinearized: Euler implicit w/ Anitescu/Stewart/Trinkle single-iteration
- ChTimestepperEulerImplicitProjected: Semi implicit Euler w/o constraint stabilization; projection
- ChTimestepperEulerSemiImplicit: Typical Euler semi implicit
- ChTimestepperHHT: Hilber-Hughes-Taylor (HHT) implicit integrator
- ChTimestepperLeapfrog: Leapfrog explicit integrator (symplectic w/ 2nd order accu.)
- ChTimestepperNewmark: Newmark constrained implicit for 2nd order DAE
- ChTimestepperTrapezoidal: Trapezoidal implicit
- ChTimestepperTrapezoidalLinearized: Trapazoidal implicit linearized
- ChTimestepperTrapezoidalLinearized2: **Do not use**

chrono::ChTimestepperIorder

- ChTimestepperEulerExpl: Euler explicit
- ChTimestepperHeun: Heun explicit
- ChTimestepperRungeKuttaExpl: 4th order explicit Runge-Kutta