Recent Advances in ADAMS/Solver

Rajiv Rampalli
Director, Core Analysis Tools
Product Development

Mechanical Dynamics

ADAMS/Solver

- Expanded domain for virtual prototyping
  - Controls
  - Vibration
  - Durability
  - New Applications (Driveline, Aircraft…)
- Solver Requirements
  - Expanded modeling capabilities
  - More robust solutions
  - Improved accuracy
  - Faster simulation performance

Advances in ADAMS/Solver

V10
- Modeling: Joint Friction, 2D contacts
- Robustness: Modified corrector
- Accuracy: SI2, Enhanced GSTIFF
- Performance: Flex Invariants, Portable mnf, msc2mnf

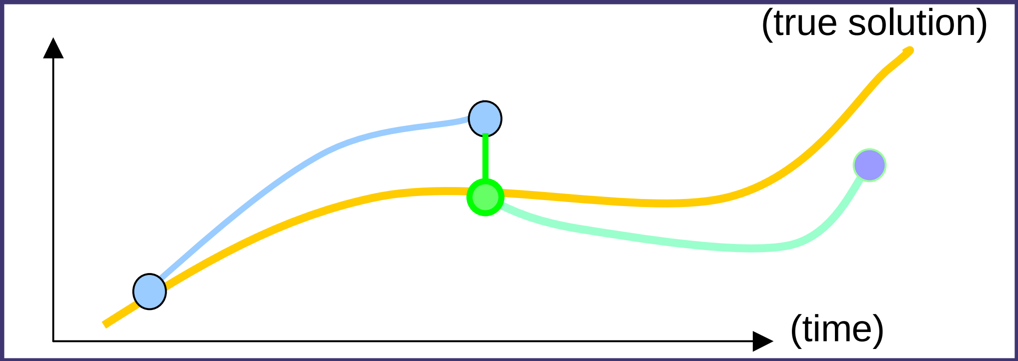
V11
- Modeling: Modal Loads, 3D contacts
- Robustness: Improved Statics
- Performance: Constant BDF

V12
- Modal state access
- Discrete GSE
- Large mnfs
- RKF-45
- SI1
- Improved Contacts
- CRATIO expressions

Mechanical Dynamics

The ADAMS Strategy

Simpler equations preferred to fewer but more complex set



Predictor	Corrector	Step/Order Control
Speed	Robustness	Accuracy
Improved extrapolation	Optimize convergence criteria Reduce index of equations	Reduce index of equations Improve modeling

Numerical Methods Improvements

ADAMS v10

Gstiff

- Enhanced corrector
- Modified corrector
- SI2 formulation

ADAMS v11

Constant BDF

- Enhanced corrector
- Modified corrector
- SI2 formulation

ADAMS v12

Integrators

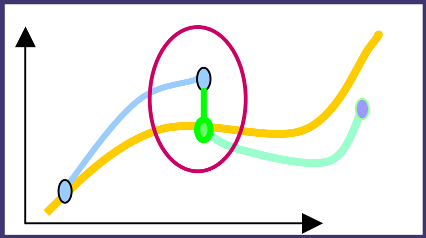
- Enhanced corrector
- Modified corrector
- SI2 formulation
- SI1 formulation

\- - - - - - - - - - - - - - - - -

- RKF 4-5

Enhanced GSTIFF Integrator

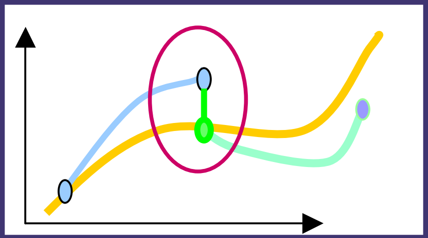
- Improved corrector
  - Optimize convergence criteria
  - Corrector convergence test is changed
    - Old Tolerance: $\varepsilon / 1000$
    - New Tolerance: $\varepsilon / 2 \bullet Neq \bullet (Order + 2)$
  - Relative Error Calculations changed
    - Old: Error = Δ / MAX (1, State)
    - New: Error = Δ / MAX (1, Max_State_so_far)



- Real-world benefits
  - Vehicle Dynamics
    - Average speed improvement of ~30%
    - No significant change in results
  - Durability
    - Accurate accelerations (reduction in spikes)

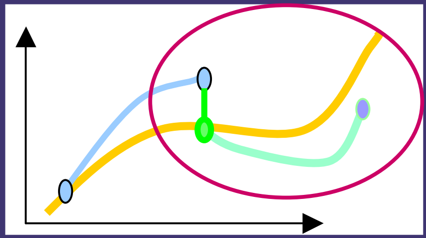
Modified Corrector

- Improved corrector
  - Optimize convergence criteria
  - Corrector monitors errors only on:
    - displacements
    - user defined dynamic states
  - Corrector does not monitor error on:
    - Velocities
    - Lagrange Multipliers
    - Forces
- Real-world benefits
  - Contact
    - 200% faster
    - Fewer corrector convergence failures
  - Friction
  - Durability



SI2/SI1

- Improved Corrector
  - Optimize state selection
- Improved Step/Order Control
  - Reduce index of DAE
- Real-world benefits
  - Nibble (wheel imbalance test)
    - 200% reduction in simulation time
    - Improved accuracy
  - Durability
    - Improved forces
    - Improved accelerations



Index 3 Formulation

$$\dot{p} - \frac{\partial L}{\partial q} + \Phi_q^T \lambda - \sum_{k=1}^{na} F_k \bullet \frac{\partial r_k}{\partial q} = 0$$

$$p - \frac{\partial L}{\partial u} = 0$$

$$u - \dot{q} = 0$$

$$\Phi(q,t) = 0$$

L = Lagrangian = T - V
Φ = Constraint Equations
r = Application point of force

p = Momenta
u = Velocity
q = Displacement
λ = Lagrange Multiplier
F = Externally appplied force

SI2 Formulation

$$\dot{p} - \frac{\partial L}{\partial q} + \Phi_q^T \lambda - \sum_{k=1}^{na} F_k \bullet \frac{\partial r_k}{\partial q} = 0$$

$$p - \frac{\partial L}{\partial u} = 0$$

$$u - \dot{q} + \Phi_q^T \mu = 0$$

$$\Phi(q,t) = 0$$

$$\dot{\Phi}(q,u,t) = 0$$

L = Lagrangian = T - V

Φ = Constraint Equations

r = Application point of force

p = Momenta

u = Velocity

q = Displacement

λ = Lagrange Multiplier

F = Externally appplied force

μ = Lagrange Multiplier

Mechanical Dynamics

SI1 Formulation

$$\dot{p} - \frac{\partial L}{\partial q} + \Phi_q^T \dot{\varsigma} - \sum_{k=1}^{na} F_k \bullet \frac{\partial r_k}{\partial q} = 0$$

$$p - \frac{\partial L}{\partial u} = 0$$

$$u - \dot{q} + \Phi_q^T \dot{\kappa} = 0$$

$$\Phi(q,t) = 0$$

$$\dot{\Phi}(q,u,t) = 0$$

L = Lagrangian = T - V

Φ = Constraint Equations

r = Application point of force

p = Momenta

u = Velocity

q = Displacement

$\dot{\varsigma}$ = Lagrange Multiplier

F = Externally appplied force

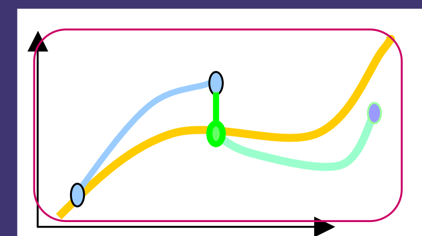
$\dot{\kappa}$ = Lagrange Multiplier

Index 3 - SI2 - SI1 Formulations

	Index-3	SI2	SI1
Accuracy	Q: High V, A: Conditional	Q, V, A: High	Q, V, A, λ : High
Robustness	Medium	High	High
Tolerance	High	Medium	Low
Speed	High	Medium	Medium
Track high frequencies	Low-Medium	High	Best

Constant_BDF

- A new fixed-step, stiff integrator
  - User must specify HMAX
- Primarily constant step size
  - Corrector ensures solution satisfies EOM
  - Step size controls error
  - Integrator runs at KMAX(6) to minimize truncation error



Integrator/Formulation options

Release 12.0

Integrator /

$$\left\{\begin{matrix} Gstiff \\ Wstiff \\ Constant\_BDF \end{matrix}\right\} , \left\{\begin{matrix} I3 \\ SI2 \\ SI1 \end{matrix}\right\} , corrector = \left\{\begin{matrix} original \\ modified \end{matrix}\right\}$$

$$RKF45$$

$$ABAM$$

Contact Modeling Improvements

ADAMS v10

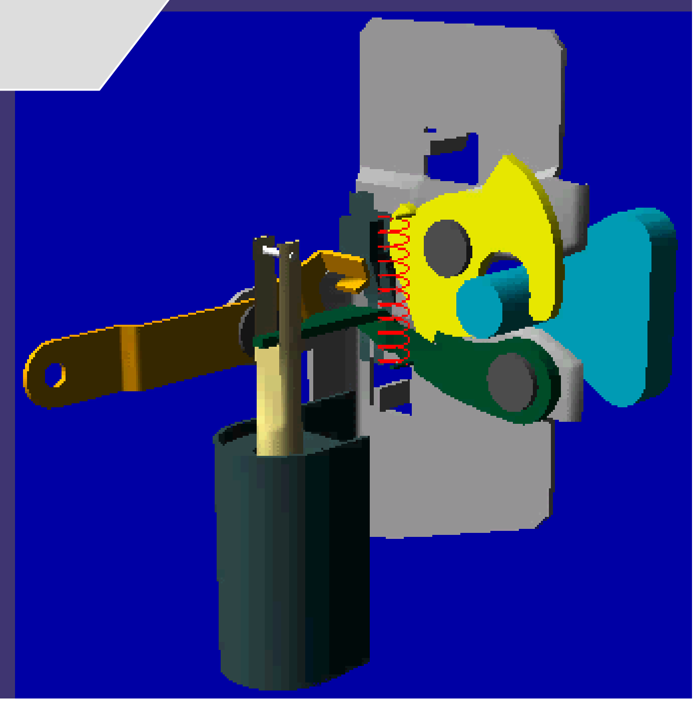
2D Contact
- •Point
- •Line
- •Curve
- •Friction

ADAMS v11

3D Contact
- •Sphere
- •Cylinder
- •Box
- •Solid

ADAMS v12

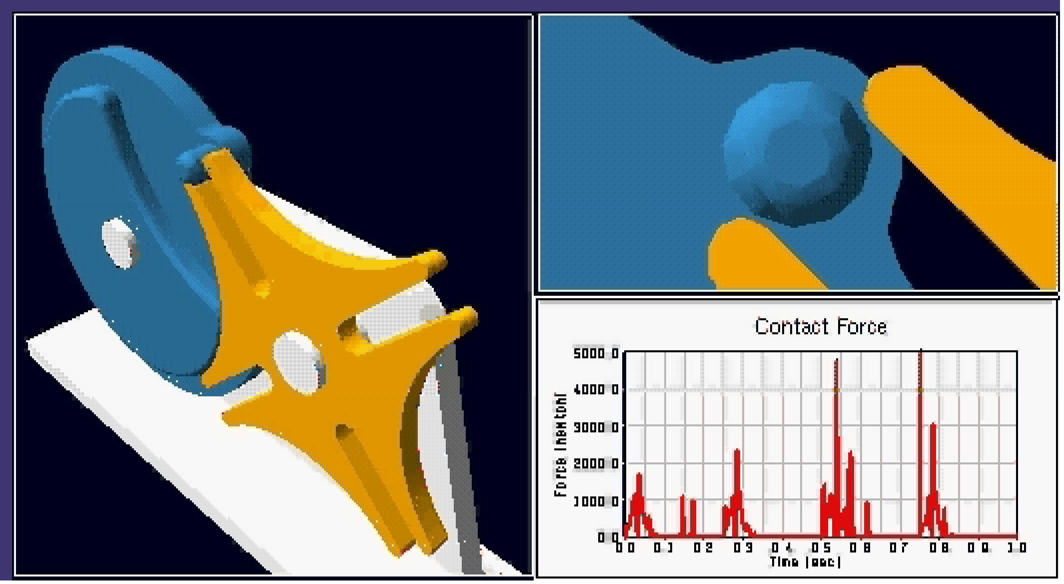
Performance
- •Robustness
- •Speed
- •Contact Function



Mechanical Dynamics

Contact Modeling

- Normal Force Models (Impact or Impulse)
- Simple Friction Model (No Stiction)
- Special Integrator Handling
  - Contact Sensing
  - Step-size and order control
- Extra Contact Output Steps



Flex Body Improvements

ADAMS v10

- User defined damping
- Improved NASTRAN Support
- Portable MNF's
- Invariant control

ADAMS v11

- Modal Force Loads
- Improved Abaqus Interface
- Preloaded Bodies

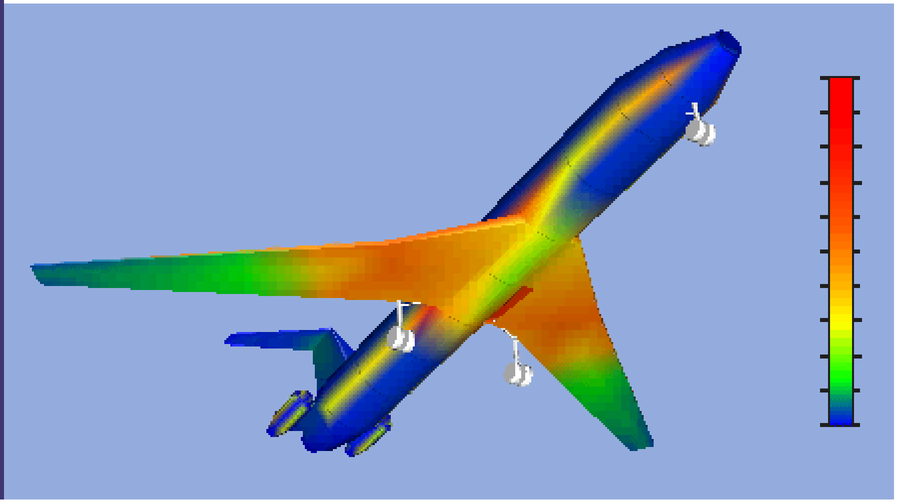
ADAMS v12

- Modal damping
- Large MNF (> 2GB)
- Modal State Access
- C++ Solver Support
- Mforce contour plots

A/Flex Improvements
Release 12.0

- Control damping ratio using an expression
- Large MNF file support (> 2GB)
- Modal State access from user subroutines
- Flex Bodies in C++ Solver
- Contour plots for MFORCE in PPT

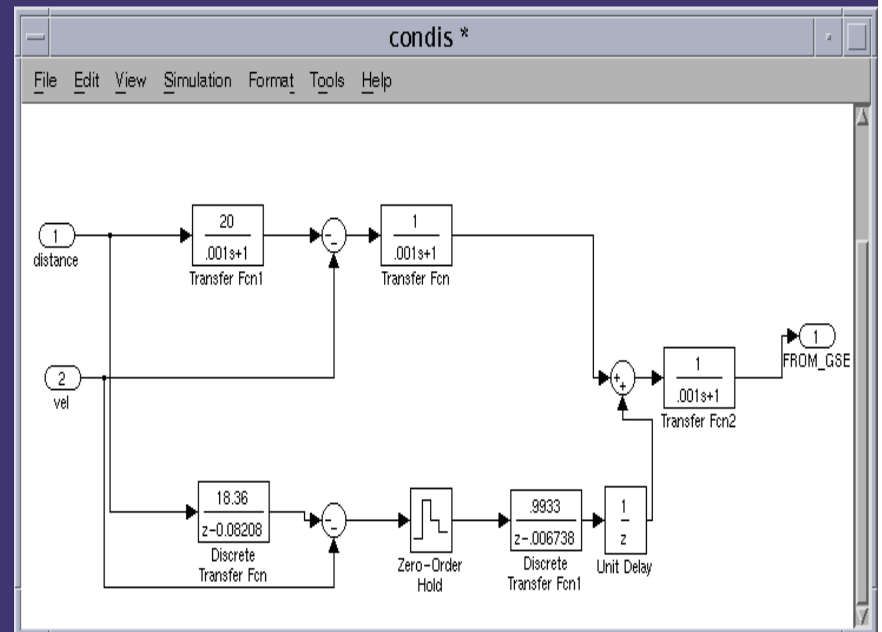
Modal Force Countours



Enhanced GSE

Release 12.0

- Support for:
  - Continuous systems
  - Discrete systems
  - Combined (Sampled) systems
- New user subroutine interface
- Subroutine interface similar to Matlab/RTW
- RTW interface available with ADAMS/Controls



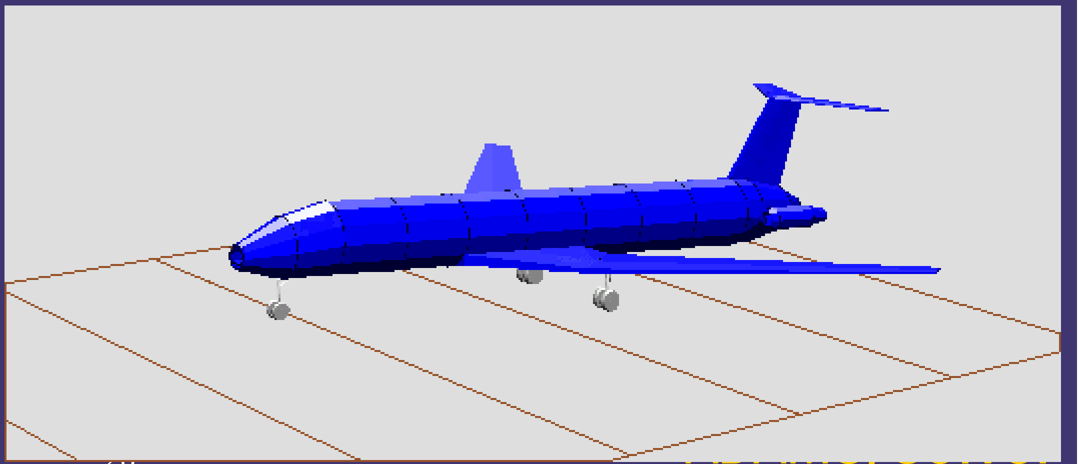
$$\dot{x}_c = f_c(x_c, u, t), \qquad x_c(t_o) = x_{co}$$

$$x_{d+1} = f_d(x_d, u, t), \quad x_d(t_o) = x_{do}$$

$$y = g(x_c, x_d, u, t)$$

ADAMS/Durability Support
Release 12.0

- Enhanced FEMDATA statement
  - von Mises stress and strain
  - Loads & Nodal Deformation
    - ABAQUS
    - ANSYS
    - NASTRAN
  - Modal deformation
    - NASTRAN
    - ANSYS



Mechanical Dynamics

The C++ Solver
Release 12.0

- Next generation ADAMS Solver
  - Object oriented structure
  - Compact and efficient
  - Facilitates encapsulation of data and methods
  - Set of modeling elements is user extensible
  - Reduced development time and maintenance costs
- Commercially available for 4 years
  - Solver for embedded products (CAD)
  - Larger user base than FullSim!
- Well-defined Interface (API)
  - Embeddable in client programs
    - CAD, Vertical products, User “main” programs

The C++ Solver
Release 12.0

- Optionally available to users since version 10
- User subroutine support (FORTRAN, C, C++)
- I/O file support
- All modeling elements except
  - Forces: NFORCE, MFORCE, FRICTION
  - Bodies: POINT_MASS
  - Integrators: ABAM, SI1, CONSTANT_BDF
  - Other: SENSOR
- Most simulation commands supported

The C++ Solver

Release 12.0

Why should I use the C++ Solver ?

By exercising the new Solver and communicating your experience you can help MDI provide you a better solver

Benefits of the C++ Solver

- Superior Modeling
  - API for Class Library - User extensibility
  - Superior implementation for Flexible Bodies
  - 2D Part
  - Generalized ACTIVATE/DEACTIVATE
  - User defined constraints
- Accuracy and Robustness
  - Analytical differentiation of expressions & user subroutines

Release 13.0

- Product migration to C++ based Solver
  - Full support for adm, acf, user subroutines
  - Default solver for selected ADAMS Vehicle Dynamics and Chassis events
- Continued improvements to the C++ Solver
  - Speed
  - Compatibility
- Improved Contact modeling, performance and post-processing

Questions?