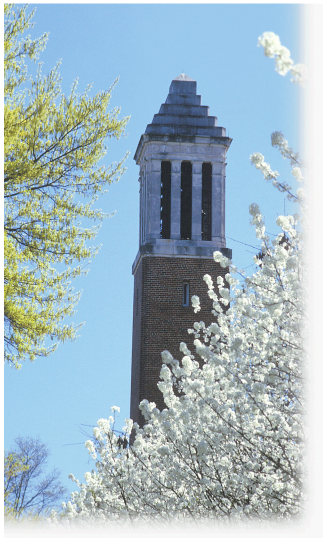
THE UNIVERSITY OF ALABAMA

Explicit\Implicit time Integration in MPM\GIMP

Abilash Nair and Samit Roy

University of Alabama, Tuscaloosa



Objectives

- Develop a Implicit algorithm for GIMP based on Implicit MPM*
- Benchmark the algorithm using exact solution to a dynamic problem
- Extend the algorithm for large deformation problems

*J.E.Guilkey and J.A.Weiss. Implicit time Integration for the material point method: Quantitative and algorithmic comparisons with the finite element method. Int. J. Numer. Meth. Engng. 2003; 57: 1323-1338

Implicit Algorithm for MPM: Review*

- Extrapolate mass, velocities, accelerations (from time t) and external forces (at time t+Δt) from material points to nodes (standard MPM). Initialize displacement of node for first iteration

$$\mathbf{u}_g^0(t+\Delta t)=\mathbf{0}$$

- Newmark approximations for displacement, velocity and accelerations of nodes at time “t+Δt”. For iteration k,

$$\mathbf{u}_g^k(t+\Delta t)=\frac{\Delta t}{2}\left(\mathbf{v}_g^k(t+\Delta t)+\mathbf{v}_g(t)\right)$$

$$\mathbf{v}_g^k(t+\Delta t)=\mathbf{v}_g(t)+\frac{\Delta t}{2}\left(\mathbf{a}_g^k(t+\Delta t)+\mathbf{a}_g(t)\right)$$

$$\mathbf{a}_g^k(t+\Delta t)=\frac{4}{\Delta t^2}\mathbf{u}_g^k(t+\Delta t)-\frac{4}{\Delta t}\mathbf{v}_g(t)-\mathbf{a}_g(t)$$

*J.E.Guilkey and J.A.Weiss. Implicit time Integration for the material point method: Quantitative and algorithmic comparisons with the finite element method. Int. J. Numer. Meth. Engng. 2003; 57: 1323-1338

Implicit Algorithm Review: continued

- Assemble internal forces and element stiffness matrix. The material points will act as integration points within each cell

$$\mathbf{F}^{k-1}_{\mathbf{int}g}(t+\Delta t) = \sum_e \int_{V_e} \mathbf{B}^T \boldsymbol{\sigma}_p dV_e \qquad \mathbf{K}^{k-1}_g(t+\Delta t) = \sum_e \int_{V_e} \mathbf{B}^T \mathbf{D}_p \mathbf{B} dV_e + \frac{4}{\Delta t^2}\mathbf{M}_g$$

- Solve for $\Delta\mathbf{u}_g$ and update incremental displacement for timestep "t+Δt"

$$\mathbf{K}^{k-1}_g(t+\Delta t)\Delta\mathbf{u}_g = \mathbf{F}_{\mathbf{ext}g} - \mathbf{F}^{k-1}_{\mathbf{int}g}(t+\Delta t) - \mathbf{M}_g\left(\frac{4}{\Delta t^2}\mathbf{u}^{k-1}_g(t+\Delta t) - \frac{4}{\Delta t}\mathbf{v}_g(t) - \mathbf{a}_g(t)\right)$$

$$\mathbf{u}^k_g(t+\Delta t) = \mathbf{u}^{k-1}_g(t+\Delta t) + \Delta\mathbf{u}_g$$

$$d\boldsymbol{\epsilon}_p = \frac{1}{2}\left(\boldsymbol{\nabla}\mathbf{u}^k_g(t+\Delta t) + (\boldsymbol{\nabla}\mathbf{u}^k_g(t+\Delta t))^T\right) \quad \text{where,} \boldsymbol{\nabla}\mathbf{u}^k_g(t+\Delta t) = \sum_g \mathbf{G}_{pg}\mathbf{u}^k_g(t+\Delta t)$$

$$\boldsymbol{\epsilon}^k_p(t+\Delta t) = \boldsymbol{\epsilon}_p(t) + d\boldsymbol{\epsilon}_p$$

$$\boldsymbol{\sigma}^k_p = \mathbf{D}_p\boldsymbol{\varepsilon}^k_p$$

Implicit Algorithm Review: continued

- Iterate until residuals are minimized (recommended error norms: displacement and energy)
- Interpolate displacement and acceleration from the grid to material point. Update position, velocity and acceleration of MP and proceed to next time step

$$\mathbf{u}_p(t+\Delta t) = \sum_i S_{ip}\mathbf{u}_i^k(t+\Delta t)$$

$$\mathbf{a}_p(t+\Delta t) = \sum_i S_{ip}\mathbf{a}_i^k(t+\Delta t)$$

$$\mathbf{x}_p(t+\Delta t) = \mathbf{x}_p(t) + \mathbf{u}_p(t+\Delta t)$$

$$\mathbf{v}_p(t+\Delta t) = \mathbf{v}_p(t) + \frac{\Delta t}{2}\left(\mathbf{a}_p(t) + \mathbf{a}_p(t+\Delta t)\right)$$

Equations for Implicit Generalized Interpolation Material Point Method (GIMP)

- In GIMP, any continuous data f(**x**), can be represented as

$$\mathrm{f}(\mathbf{x}) = \sum_p \mathrm{f}_p \chi_p(\mathbf{x})$$

- Consider the Integral:

$$\begin{aligned}
\mathbf{K}_g &= \sum_e \int_{\Omega_e} \mathbf{B}^T \mathbf{D}(\mathbf{x}) \mathbf{B} d\Omega_e \\
&= \sum_e \int_{\Omega_e} \mathbf{B}^T \left(\sum_p \mathbf{D}_p \chi_p(\mathbf{x}) \right) \mathbf{B} d\Omega_e \\
&= \sum_e \sum_p \mathbf{D}_p \iiint_{\Omega_p} \mathbf{B}^T \mathbf{B} d\Omega_p
\end{aligned}$$

If N_i and N_j are Interpolation functions to node i and node j, respectively (S_i is the grid shape function)

$$\begin{aligned}
N_i(x, y, z) &= S_i^x \cdot S_i^y \cdot S_i^z \\
N_j(x, y, z) &= S_j^x \cdot S_j^y \cdot S_j^z
\end{aligned}$$

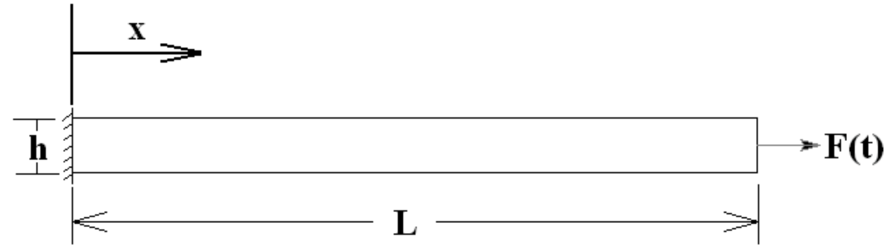
$$S_i^x = \begin{cases} 0 & x \leq x_i - L^x \\ 1 + \frac{x - x_i}{L^x} & x_i - L^x < x \leq x_i \\ 1 - \frac{x - x_i}{L^x} & x_i < x \leq x_i + L^x \\ 0 & x > x_i + L^x \end{cases}$$

Then we have for example for a two-dimensional problem,

$$\begin{aligned}
&\iint_p \frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial y} t_p dx dy \\
&= t_p \iint_p \left(\frac{dS_i^x}{dx} \cdot S_i^y \right) \left(S_j^x \cdot \frac{dS_j^y}{dy} \right) dx dy
\end{aligned}$$

Benchmark Problem #1: Traveling Wave*

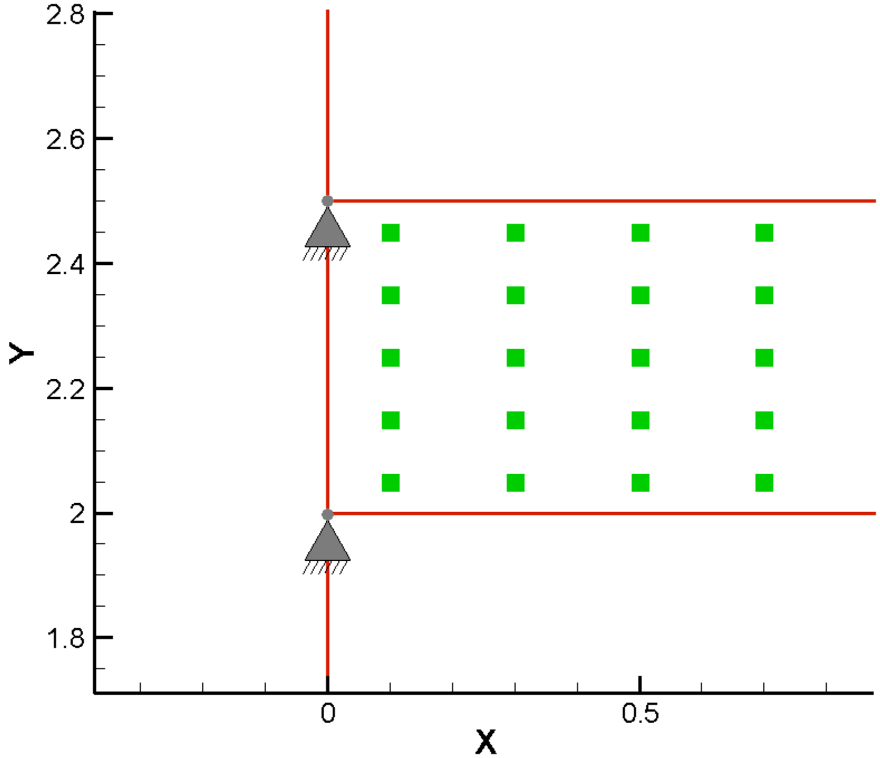
$F(t) = F_0\, u(t)$ [***u*****(t)** is unit step function]
L = 30 mm (for infinite span beam)
b (thickness) = 1 mm
h = 0.5 mm
E = 200GPa, v=0.3, ρ=7.8 g/cc
F_0 =1N



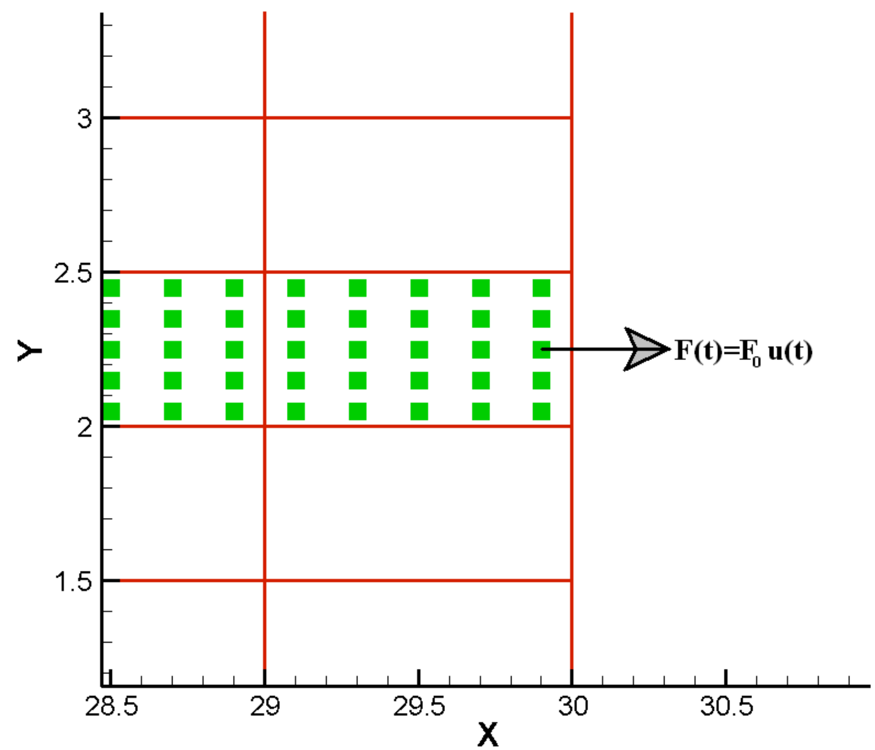
$$u(x,t) = \frac{8F_0L}{\pi^2 EA}\sum_{r=1}^{\infty}\frac{(-1)^{r-1}}{(2r-1)^2}\sin\frac{(2r-1)\pi x}{2L}\left[1-\cos\frac{(2r-1)\pi}{2}\sqrt{\frac{EA}{mL^2}}t\right]$$

***L.Meirovitch.** *Fundamentals of Vibrations*

Boundary Conditions in MPM for Problem #1

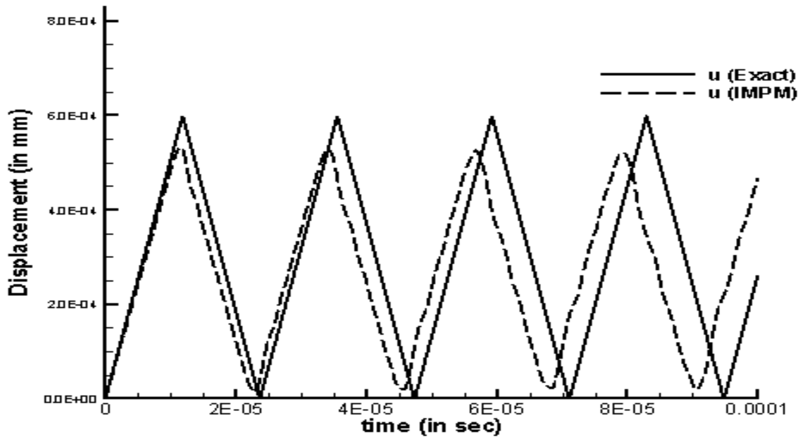


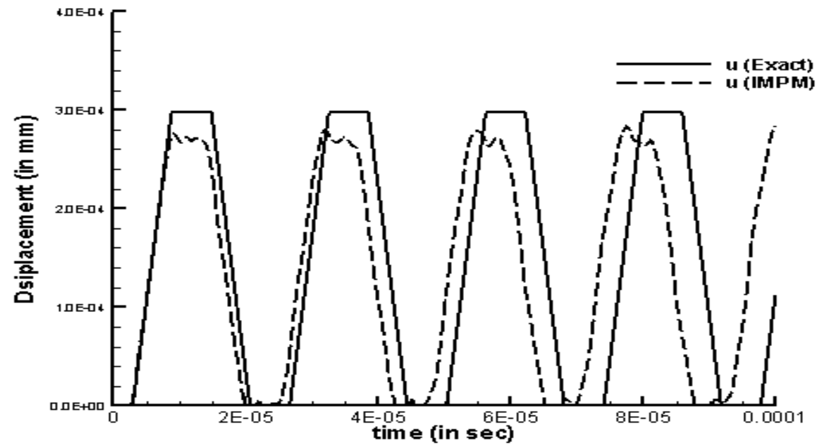
Displacement BC

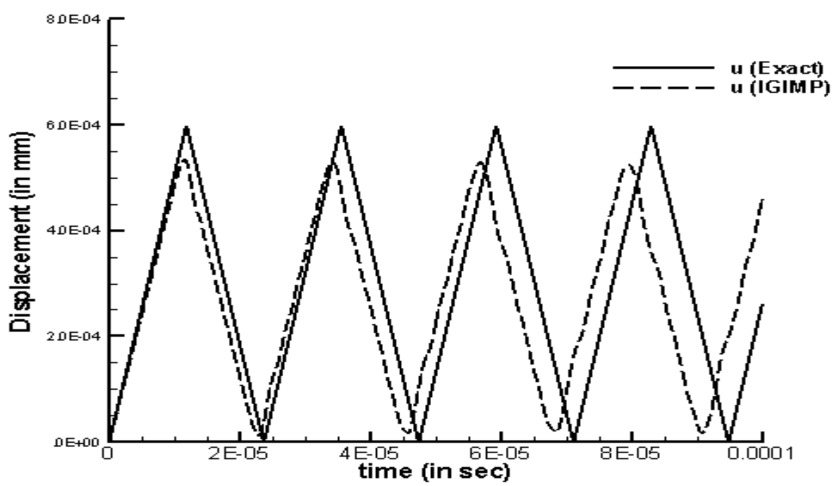


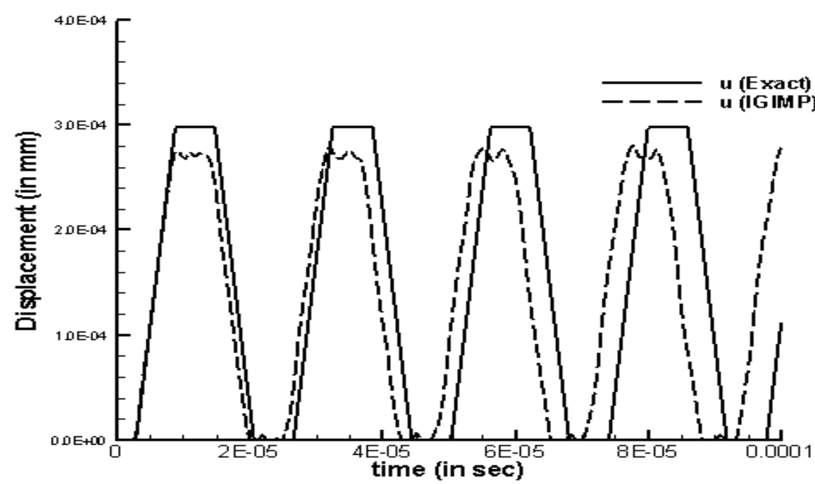
Force BC

Results: Implicit ($\Delta t=10^{-8}$, 10^4 timesteps, 30x1 grid, 25MPs per cell)





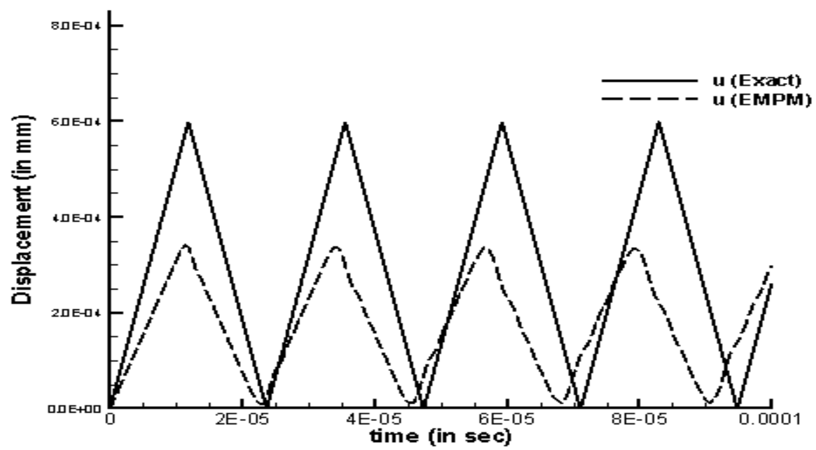


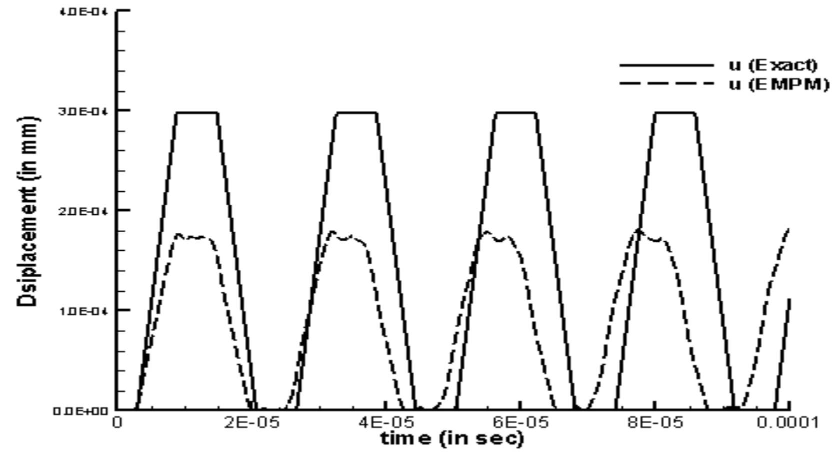


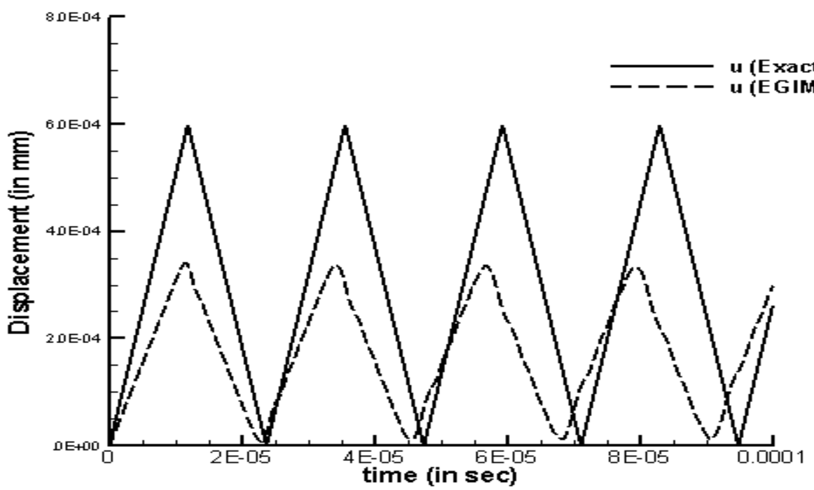
at x=L

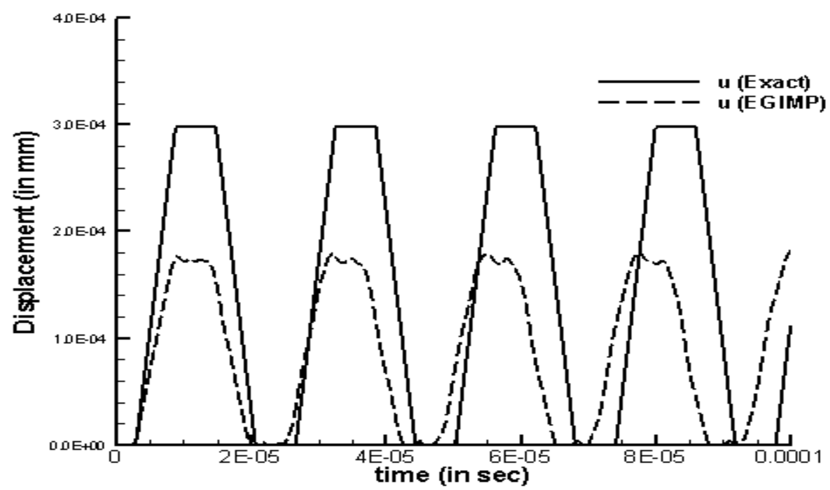
at x=L/2

Results: Explicit ($\Delta t=10^{-8}$, 10^4 timesteps, 30x1 grid, 25MPs per cell)





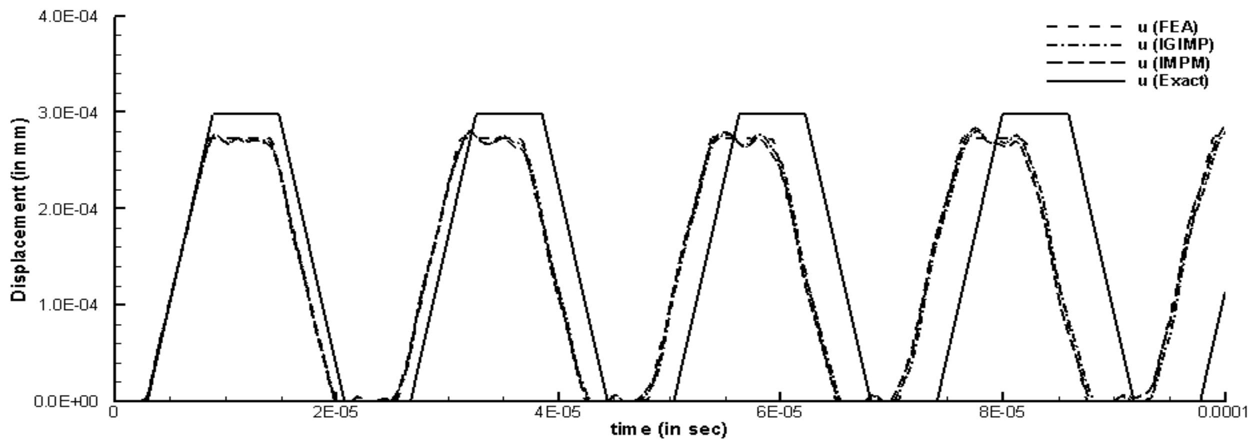




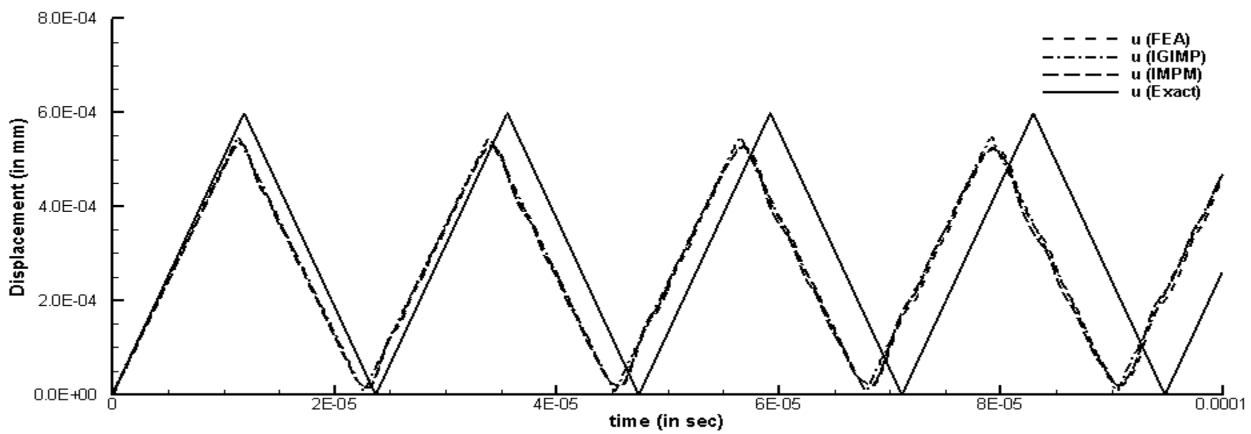
at x=L

at x=L/2

Displacement Results: Implicit MPM and GIMP ($\Delta t=10^{-8}$, 10^4 time steps, 30x1 grid, 25MPs per cell) with FEA ($\Delta t=10^{-8}$, 10^4 time steps, 30x2 grid, 8 node plane strain elements)

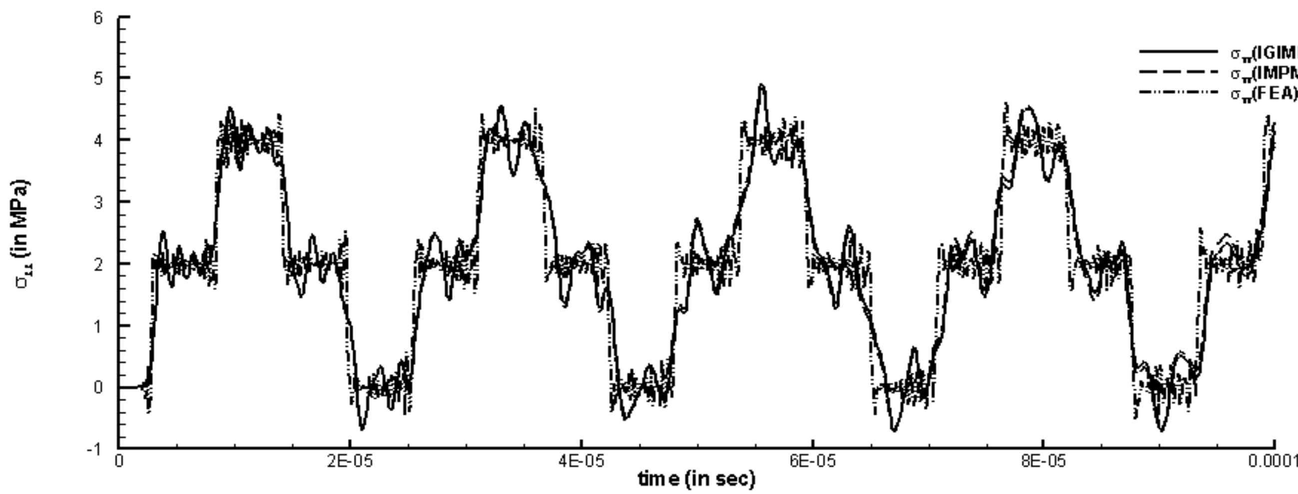


at x=L/2

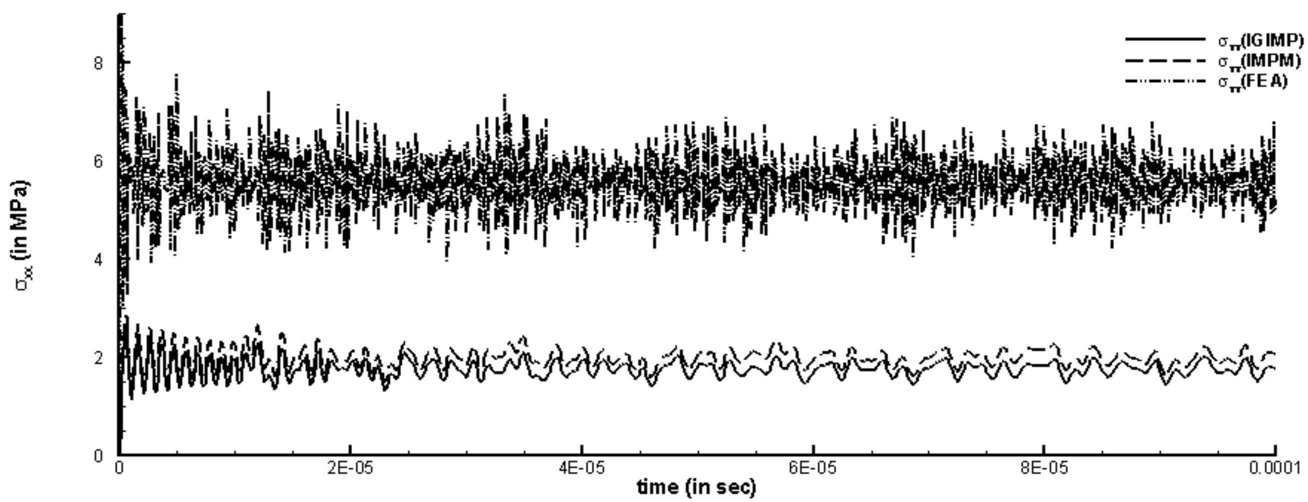


at x=L

Stress Results: Implicit MPM and GIMP ($\Delta t=10^{-8}$, 10^4 time steps, 30x1 grid, 25MPs per cell) with FEA ($\Delta t=10^{-8}$, 10^4 time steps, 30x2 grid, 8 node plane strain elements)



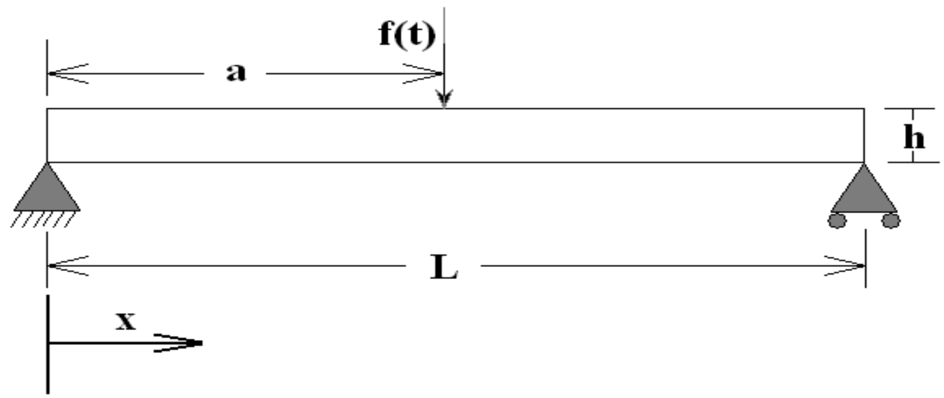
at x=L/2



at x=L

Benchmark Problem #2: Forced vibration of beam*

$F(t) = F_0 \sin(\omega\, t)$ applied at distance “a” from the edge
L = 10 m
b (thickness) = 1 m
h = 0.5 m
E = 200GPa, v=0.3, ρ=7.8 g/cc
F_0 =-4N, ω=150 Hz

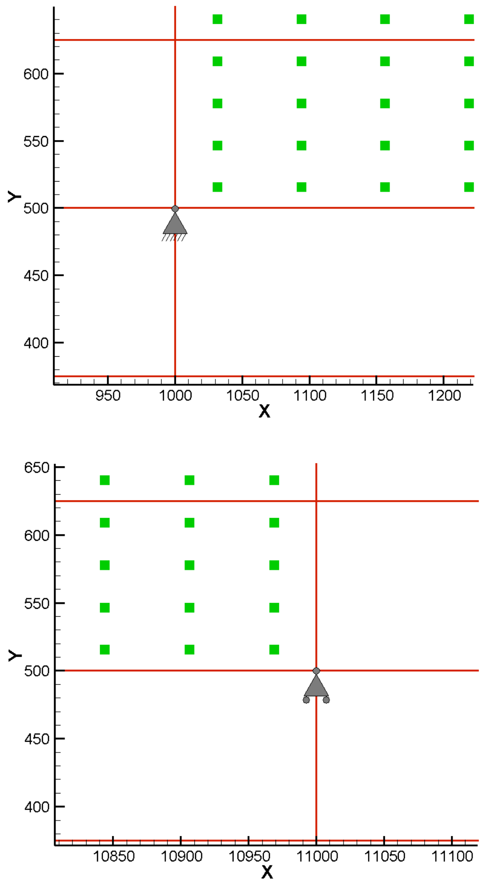


$$w(x,t) = \frac{2F_0 L^3}{\pi^4 EI} \sum_{n=0}^{\infty} \frac{\sin\left(\frac{n\pi a}{L}\right)\sin\left(\frac{n\pi x}{L}\right)}{n^4 - \left(\frac{\omega}{\omega_1}\right)^2} \left[\sin(\omega t) - \frac{\omega}{n^2\omega_1}\sin(n^2\omega t)\right]$$

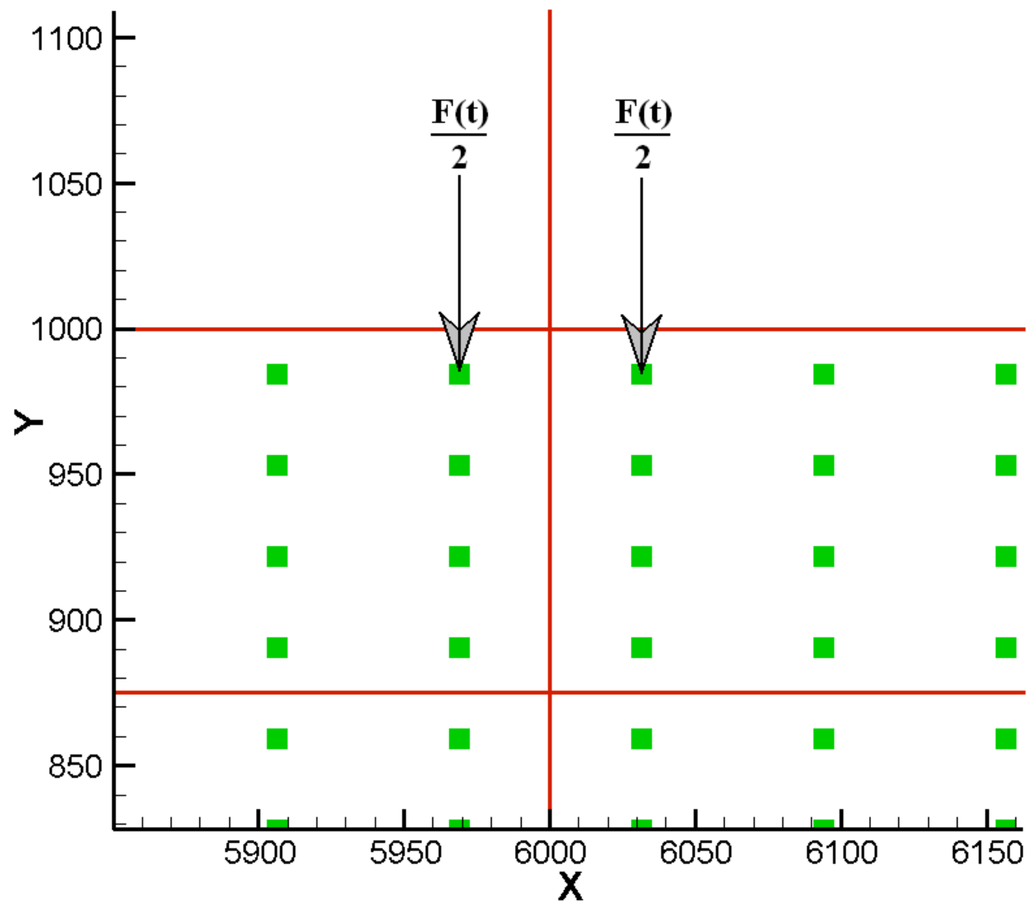
$$\omega_n = \frac{n^2\pi^2}{L^2}\sqrt{\frac{EI}{\rho A}}$$

***E.Volterra and E.C.Zachmanoglou.** *Dynamics of Vibrations*

Boundary Conditions in MPM for Problem #2

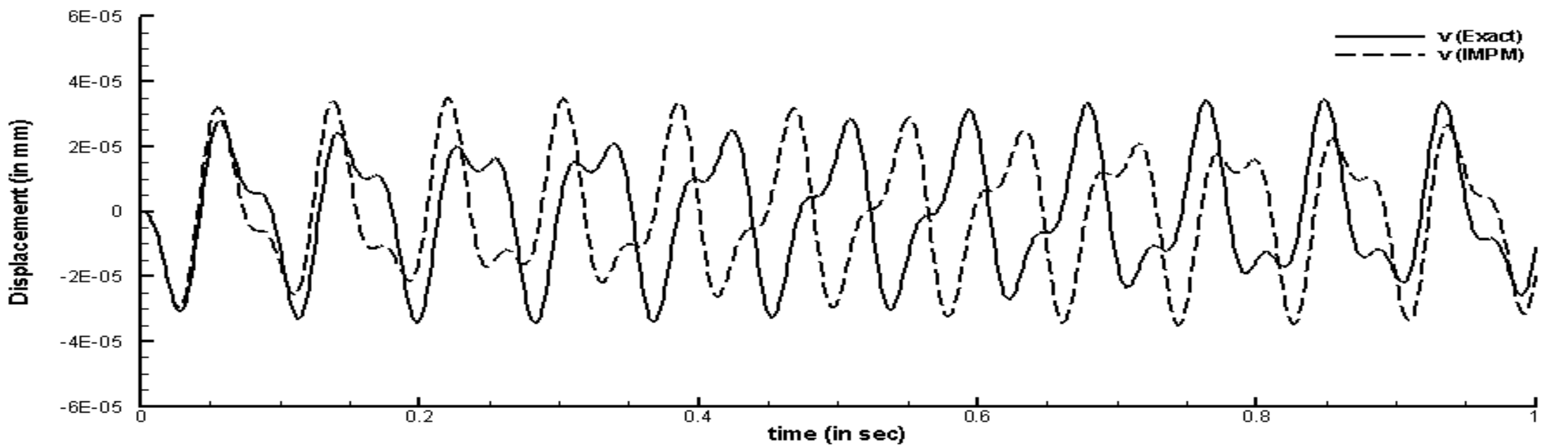


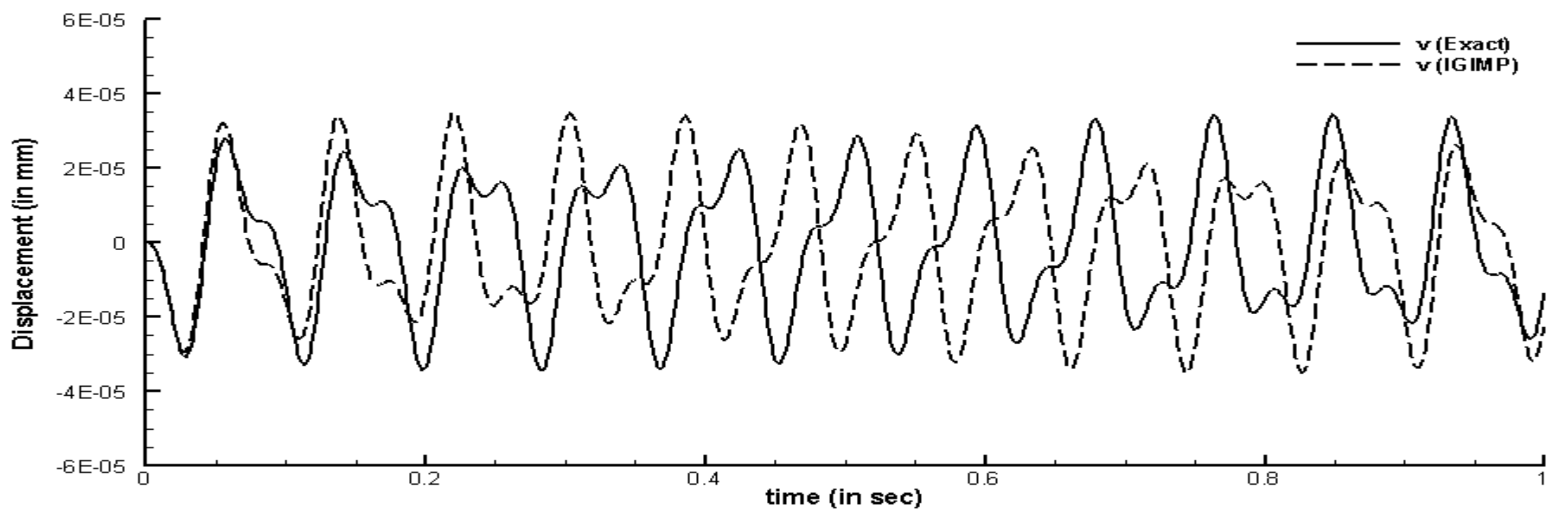
Displacement BC



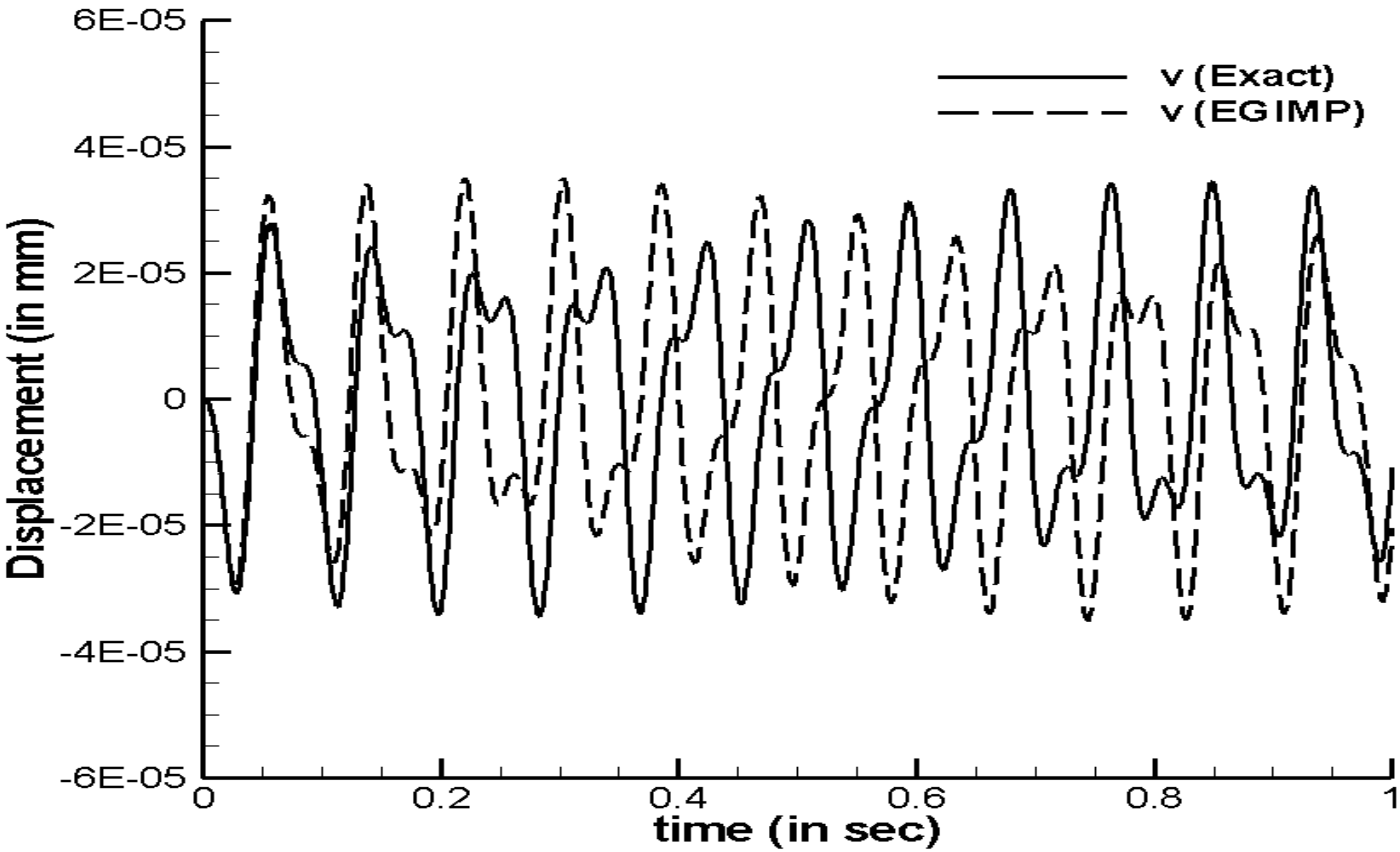
Force BC

Results: Implicit ($\Delta t=10^{-3}$, 10^3 time-steps, 40x4 grid, 16MPs per cell)

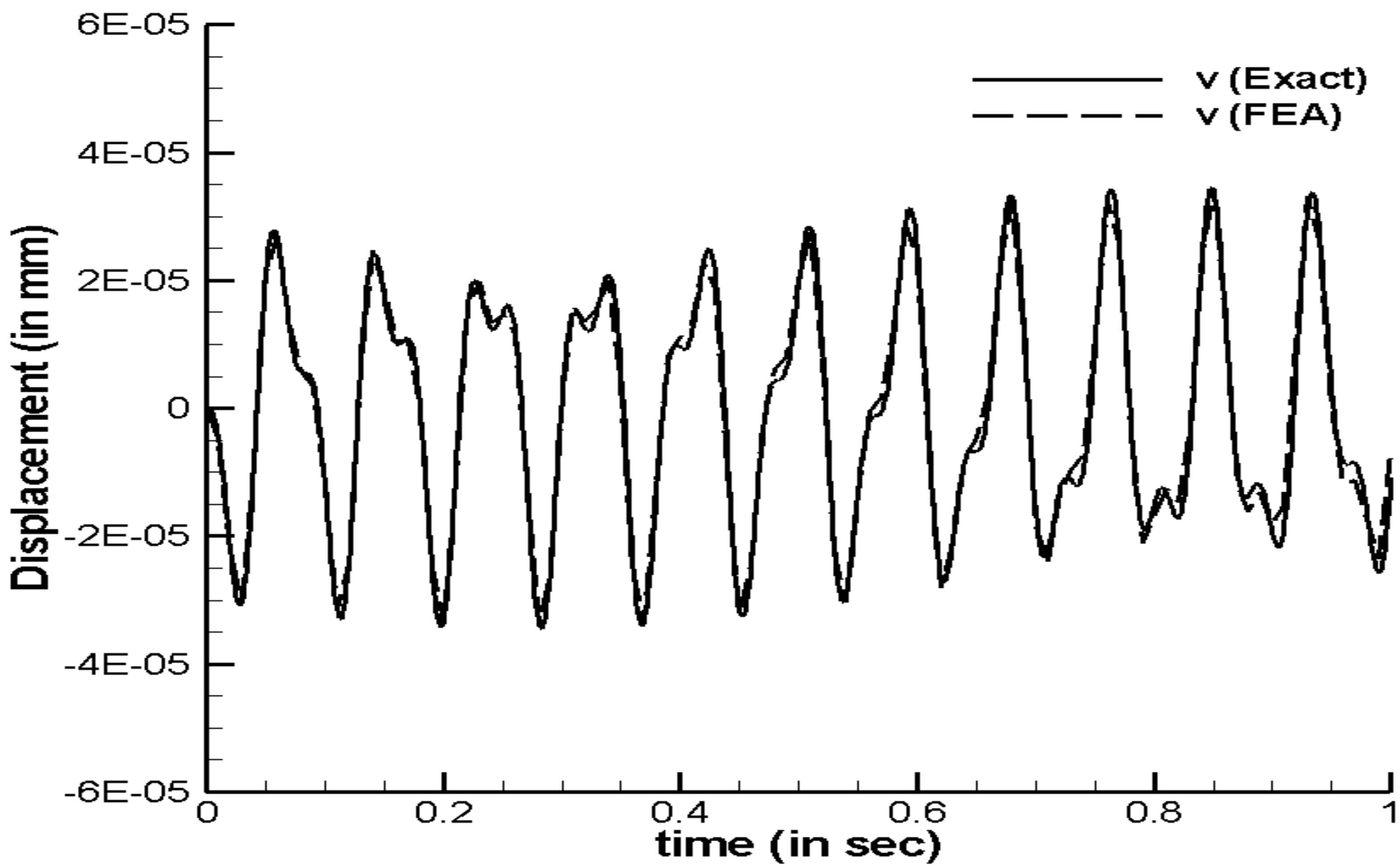




Results: Explicit ($\Delta t=10^{-7}$, 10^7 time-steps, 40x4 grid, 16MPs per cell)



Results: Implicit FEA ($\Delta t=10^{-3}$, 10^3 time-steps, 40x4 Plane Strain Elements, 4 node quad.)



Conclusions and Future Work

- Implicit algorithm for GIMP seems to agree well with Implicit MPM as well as Explicit MPM

- Discrepancies were observed between exact solution and MPM solutions for dynamic benchmark problems

- Extend the IGIMP algorithm for large deformation problems

Animation (Traveling wave solution. IGIPM)