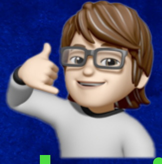
# Comparison of Explicit and Implicit Finite Element Methods

Dr. Lonny L. Thompson

March 14, 2025

- Let's examine the differences, advantages, and disadvantages, factors for deciding which method to use for finite element analysis (FEA), typical applications with examples, and how to use explicit methods effectively.
- Hybrid Implicit-Explicit methods and mass scaling approximations are also discussed with examples.
- Deciding between implicit and explicit methods can be particularly challenging and nuanced in certain cases.



# System of Differential Equations to Solve in Time

For structural dynamics, after finite element discretization in space and assembly of global matrices, the system of 2nd-order differential equations in time describing the dynamic equations of motion are:

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) = F(t)$$

$M$ : Mass matrix

$C$ : Damping matrix

$K$ : Stiffness matrix

$u(t)$ : Displacement vector as function of time

$\dot{u}(t)$: Velocity vector as time derivative of displacement.

$\ddot{u}(t)$: Acceleration vector as time derivative of velocity.

$F(t)$ : External time-dependent load vector

# Implicit and Explicit Time Stepping Methods

1. **Implicit Method:** Implicit methods solve equilibrium equations implicitly at the current time step by solving a system of equations:

$$M\ddot{u}_{n+1} + C\dot{u}_{n+1} + Ku_{n+1} = F_{n+1}$$

- Common examples include
  - the Newmark-$\beta$ method and
  - the Backward Euler method.
- In implicit methods, displacements at the next time step $u_{n+1}$ are solved from equations at the unknown time step, $t_{n+1}$.

$$K_{\text{eff}}\, u_{n+1} = F_{n+1} + M\tilde{a}_n + C\tilde{v}_n$$

where

$$K_{\text{eff}} = a_0 M + a_1 C + K$$

is an ‘effective’ stiffness matrix, and $a_0, a_1$ are parameters that depend on sub-step evaluation points and the scheme used.

- The right-hand side terms are known from the previous step. For Newmark-$\beta$, with parameters $\beta, \gamma$, the update equations are:

$$\left(\frac{M}{\beta\Delta t^2} + \frac{\gamma C}{\beta\Delta t} + K\right) u_{n+1} = F_{n+1} + M\left(\frac{u_n}{\beta\Delta t^2} + \frac{\dot{u}_n}{\beta\Delta t} + \frac{\ddot{u}_n(1-2\beta)}{2\beta}\right) + C\left(\frac{\gamma u_n}{\beta\Delta t} + \frac{\dot{u}_n(\gamma-\beta)}{\beta} + \frac{\ddot{u}_n\Delta t(\gamma-2\beta)}{2\beta}\right)$$

- Typically, the implicit solver factorizes the effective stiffness matrix at each time step with iterations required for nonlinear problems to update the effective 'tangent stiffness matrix,' resulting in relatively intensive computation per step but stable for larger steps.
- Since the solution from the previous step $t_n$ is usually a good approximation to the next step, the number of iterations required is generally lower than a statics iterative solution.
- Typically, to dampen unwanted high-frequency numerical artifacts from insufficient spatial finite element mesh resolution for structural dynamics, implicit methods such as the HHT-$\alpha$ method are used.

2. **Explicit Method:** Explicit methods directly compute the next step using previously known values without solving simultaneous equations:

$$M\ddot{u}_n + C\dot{u}_n + Ku_n = F_n$$

- Typical examples include the Central Difference method and the Forward Euler method.
- For explicit methods, the equations are solved directly without iteration.
- Often, the damping is neglected (or assumed lumped), and the central difference method explicitly calculates the next-step displacement directly from the current $t_n$ and previous $t_{n-1}$ steps.

$$u_{n+1} = \Delta t^2 M^{-1} (F_n - Ku_n) + 2u_n - u_{n-1}$$

No inversion is required for stiffness or damping matrices at each step.

- The mass matrix $M$ is typically diagonalized with, for example, lumped mass so that the inversion is a trivial scaling.

| Aspect | Implicit Method | Explicit Method |
|---|---|---|
| Matrix Equation Form | $(M+C+K)u_{n+1} = F_{n+1}$ (implicit solution) | $u_{n+1} = M^{-1}[\ldots]$ (explicit direct) |
| Matrix Inversion | Required (iterative/direct solver) | Not required (diagonal mass matrix) |
| Stability | Unconditionally stable | Conditionally stable (CFL condition) |
| Time Step Size | Larger (stable) | Very small (CFL condition) |
| Computational Effort per step | Higher (requires matrix inversion/iterations) | Low (direct) |
| Computational cost (overall) | Moderate to High (fewer large steps) | High for long-duration analyses (small steps) |
| Solver Type | Iterative (nonlinear) | Direct calculation |
| Suitable problems | Stiff or nonlinear, steady-state, quasistatic | Short-duration dynamic, impact, transient |
| Accuracy | Nonlinear, quasistatic, damped problems | Transient, wave propagation |

## Advantages and Disadvantages

### Explicit Method:

- **Advantages:**
  - Simple and direct computation
  - Easily parallelizable
  - Optimal for dynamic and impact problems
  - Fast per step
- **Disadvantages:**
  - Stability conditions limit time-step size
  - Less suitable for long-duration problems

### Implicit Method:

- **Advantages:**
  - Stable for larger time steps
  - Accurate for long-term, nonlinear analyses
  - Handles stiff problems effectively
- **Disadvantages:**
  - Computationally intensive per step
  - Requires robust iterative solvers
  - More challenging parallelization

## Typical Applications

### Implicit Method:

- Nonlinear structural analysis (large deformation, plasticity)
- Quasi-static structural analysis
- Large damping
- Thermal and coupled field problems

### Explicit Method:

- Crash tests, automotive impact analysis
- Wave propagation, explosions, shock loading
- High-velocity impact and ballistic problems

## Decision Criteria

| Criteria | Explicit Preferred | Implicit Preferred |
|---|---|---|
| Simulation duration | Short-duration, highly transient | Long-duration, quasi-static |
| Nonlinear complexity | Moderate (impact events) | High, static/quasi-static problems |
| Stability concerns | Small steps manageable | Stability is priority |
| Computational resources | Parallel resources available | Limited parallelization |

- Explicit methods require significantly less memory than implicit methods.
- The requirement for a small time step size is due to the stability limit of the numerical integration, not the physical behavior.
- Implicit FEA time integration requires more memory, while explicit FEA needs more processor performance and speed to update with many small time steps.
- Since explicit methods do not require matrix solutions, they are very easily solved with parallel processing with little data exchange, allowing these algorithms to scale speed across many processors much more effectively than implicit methods.

## Summary

- Due to their stability and permissible larger time steps, implicit methods are ideal for large-scale, long-duration, or stiff problems. Explicit methods are suitable for short-duration, highly dynamic events.
- Implicit methods are preferable for complex, long-term simulations due to stability and large allowable time steps. Explicit methods are advantageous for short, dynamic simulations requiring rapid computations.

# Explicit Method Example: Automobile Crash Simulation

- In part, the goal of a frontal crash test involving a car impacting a rigid barrier at high speed is to predict the deformation and energy dissipation of the vehicle structure, accelerations experienced by occupants, and structural failure locations.
- This type of simulation is well-suited for explicit methods since it is a short-duration event ( $\sim 100$ milliseconds) with severe deformation and highly nonlinear behavior (material plasticity, large deflections, contact dynamics).
- Explicit methods effectively handle rapidly changing contact conditions and crushing due to sudden impact.
- The dynamic explicit time integration must use very small time steps ( microseconds) to satisfy stability criteria (Courant-Friedrichs-Lewy condition), which is physically needed anyway to resolve the extremely short impact event accurately.

## Implicit Method Example:
## Seismic Response of High-rise Building

- Evaluating the dynamic response of a multi-story steel-frame building subjected to earthquake ground motions lasting several seconds or minutes, the primary goals include determining structural stability, evaluating drift, and assessing stresses throughout the earthquake.

- Implicit methods are best suited for this extended-duration event, lasting several seconds to minutes, involving low-frequency vibrations driven by seismic acceleration applied at the base as boundary conditions with Rayleigh or modal damping in the dynamic analysis.

- The structural response is dominated by elastic deformation, possibly moderate plastic deformation in localized regions.

- Using implicit time integration, nonlinear equations are solved at each step iteratively, allowing large, stable steps ( $\sim$ milliseconds to seconds). The larger time steps allowed by implicit methods significantly reduce computation time for this analysis compared to explicit methods.

## Comparative Summary

| Aspect | Explicit Example | Implicit Example |
|---|---|---|
| Problem Type | Impact, crash dynamics | Seismic, earthquake response |
| Duration | Very short (milliseconds) | Long (seconds to minutes) |
| Dominant Effects | High-frequency wave propagation | Low-frequency vibrations |
| Time Step | Microseconds (tiny steps) | Milliseconds to seconds (larger steps) |
| Computational Effort | Many fast, small steps | Fewer larger, computationally heavy steps |

# Examples where the choice between Implicit and Explicit is not as clear

## Nonlinear Problems with both Dynamic and Quasi-Static Characteristics

- Problems like forming processes, incremental loading of structures, or mechanical interactions with nonlinearities have tradeoffs in the severity of nonlinearity and an acceptable run-time.

- Implicit methods handle nonlinearity efficiently, with equilibrium checks within iterations and large time steps. However, the iterative solver might have complexities that require many iterations to converge.

- Explicit methods handle severe nonlinearities effectively without iterations, but small time steps can become computationally expensive over prolonged load durations.

## Manufacturing Forming Processes

- An example of the tradeoffs when considering between explicit and implicit methods is sheet metal stamping of an automotive panel from a flat sheet; the forming process involves moderate dynamics (the punch moves at an intermediate velocity of about 1 m/s), significant nonlinear deformation frictional contact, and moderate-duration loading ( $\sim 0.5$ seconds).

- The steel is modeled with plastic hardening (nonlinear) behavior; explicit time-stepping requires an estimated CFL time step:

$$c_{\max} = \sqrt{\frac{E}{\rho}} = \sqrt{\frac{2.1 \times 10^{11}}{7850}} \approx 5189 \text{ m/s}$$

With a mesh size of around 2 mm:

$$\Delta t_{\text{explicit}} = \frac{0.002\,\text{m}}{5189\,\text{m/s}} \approx 3.85 \times 10^{-7}\,\text{s} \approx 0.385\mu\text{s}$$

The number of required explicit steps for 0.5-second duration:

$$N = \frac{0.5s}{3.85 \times 10^{-7}s} \approx 1.3 \times 10^{6} \text{ steps}$$

which is time-consuming.

- Using an implicit method allows significantly larger time steps and efficiently handles moderate dynamics without excessive computational burden. Assuming a conservative time step:

  $$\Delta t_{\text{implicit}} \approx 1 \times 10^{-3}\,\text{s} = 1\,\text{ms}$$

  $$N = \frac{0.5\,\text{s}}{1 \times 10^{-3}\,\text{s}} = 500 \text{ steps}$$

  Within each step, iterative nonlinear solutions are solved using a good initial guess from the previous time step.

- Convergence difficulties may occur during severe and rapid contact and nonlinear plasticity changes, potentially significantly increasing the computational cost per step.

- Choosing the best method may involve - preliminary simulations comparing run times, evaluation of solver robustness in implicit solutions, and checking explicit mass scaling accuracy loss effects to speed up the analysis.

## Complex contact analysis

- For problems involving intermittent or frictional contacts (e.g., bolted joint slippage, multi-body assembly), implicit methods provide robust solutions by solving equilibrium-based contact conditions but can have difficulties converging.
- Explicit methods handle contact easily due to direct time integration and no iterative equilibrium solving, yet the stability conditions limit the allowable time step.
- The choice depends on contact complexity and acceptable simulation run-time versus solver robustness.

## Multi-frequency response analysis

- Problems with mixed-frequency content can be especially changing for explicit methods, which easily capture high-frequency behavior but may require prohibitively small steps to maintain stability through low-frequency vibrations over longer durations.

- Structures experiencing a mixture of high-frequency and low-frequency dynamic behaviors, such as structures with flexible appendages or vibrating equipment, may be better solved using implicit or hybrid methods (combining implicit and explicit features) to resolve low-frequency responses efficiently but are also challenged by the solver frequently having to resolve rapid high-frequency transient effects.

- Balancing the accurate resolution of both frequency extremes makes this choice complicated.

## Multi-physics and coupled problems

- Multi-physics or coupled-field problems involving thermal-mechanical, fluid-structure, or electromagnetic-mechanical interactions can be challenging.
- Implicit and hybrid methods provide stability and flexibility for solving coupled equations but can become computationally costly for iteratively solving coupled nonlinear equations.
- Explicit methods alone are easy to implement coupling but may limited by severe stability restrictions in one or both the physical fields interacting.

## Hybrid Implicit-Explicit Methods

- Hybrid implicit-explicit methods, commonly known as IMEX methods in the CFD community, combine the strengths of both implicit and explicit integration approaches within a single simulation.

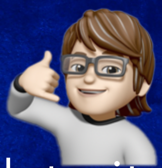
- These hybrid methods use explicit integration in regions or during phases involving severe nonlinearity, rapid dynamics, or wave propagation and implicit integration in regions or during phases with slow dynamics, stability challenges, or stiff equations.

- IMEX schemes generally partition equations from the structural dynamic matrix equations

$$M\ddot{u}(t) + C\dot{u}(t) + Ku(t) = F(t)$$

  into two subsets:

  - $F^{\text{explicit}}$: the explicit part is easily evaluated, less stiff, nonlinear terms, or contact dynamics,
  - $F^{\text{implicit}}$: the implicit part handles stiff parts, linear structural response, damping, and slowly varying nonlinearities.



- The general IMEX scheme is to solve and update iteratively.

$$M\ddot{u}_{n+1} + C\dot{u}_{n+1} + Ku_{n+1}^{\text{implicit}} = F_{n+1}^{\text{implicit}}, \quad (\text{ implicit update })$$
$$M\ddot{u}_{n} + C\dot{u}_{n} + Ku_{n}^{\text{explicit}} = F_{n}^{\text{explicit}}, \quad (\text{ explicit update })$$

- An example is fluid-structure interaction (FSI), where an explicit fluid dynamics solver handles fast transient flows, and an implicit solver is used for the structural deformation for slower structural response.
- An example of a sequential explicit-implicit time integration is where FEA software can run explicit analysis first for, say, initial impacts, then transition to implicit in simulating slower structural relaxation, for example, metal forming or spring-back simulations.

# Courant Stability Condition on Time-Steps for Explicit Methods

- The Courant stability condition, also known as the Courant-Friedrichs-Lewy (CFL) condition, determines the maximum allowable time step size $(\Delta t)$ in explicit finite element simulations to maintain numerical stability.

- For explicit integration schemes (such as the central difference method), the CFL condition for structural dynamics can be expressed as:

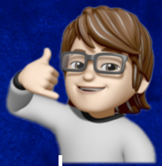
$$\Delta t \leq \frac{L_{\min}}{c_{\max}}$$

$\Delta t$ : Time step size.

$L_{\min}$ : Smallest characteristic length (element size) in the mesh.

$c_{\max}$ : Maximum wave propagation speed within the material.

- The physical meaning of this condition is that the numerical wavefront must not pass through more than one element in a single time step. Thus, smaller element sizes require proportionally smaller time steps.



- The CFL condition ensures that stress waves do not propagate further than the smallest dimension of an element within each time step. If waves travel too far in a single step, numerical errors grow and accumulate, causing instability and erroneous oscillations.

- If the CFL condition is violated, numerical instability occurs with non-physical and divergent solutions produced in the simulation.

- In structural dynamics, the maximum wave propagation speed $(c_{\max})$ for solids is usually the longitudinal wave speed.

- For simplified conditions (such as one-dimensional bars), this reduces to:

$$c_{\max} = \sqrt{\frac{E}{\rho}}$$

$E$ : Young's modulus

$\rho$ : Material density

## Practical Example of Courant Stability Condition for Explicit Analysis

- For a steel component analyzed explicitly: $E = 210$ GPa and $\rho = 7850$ kg/m$^3$, the approximate longitudinal wave speed is

$$c_{\max} = \sqrt{\frac{210 \times 10^9 \text{ Pa}}{7850 \text{ kg/m}^3}} \approx 5189 \text{m/s}$$

- If the smallest mesh element dimension is $L_{\min} = 1$ cm, the CFL condition gives:

$$\Delta t \leq \frac{0.01 \text{ m}}{5189 \text{ m/s}} \approx 1.93 \times 10^{-6} \text{ s} \quad (\approx 1.93 \mu\text{s})$$

- This indicates an extremely small allowable time step is needed, requiring many steps for longer-duration simulations.
- Reducing mesh size by half reduces the allowable time step by half, increasing computational cost further.

# Mass Scaling Approximation for Explicit Methods

- Strategies for practical implementation of explicit methods include mass scaling, which artificially increases mass (carefully controlled) to lower wave speed and allow larger time steps.

- While this improves efficiency for longer-time integration, the trade-off is loss of accuracy. When using explicit methods, the user must carefully avoid excessively small elements in critical regions unless essential. Otherwise, the simulation time can be excessive.

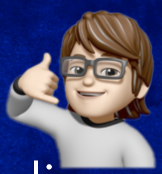
- Mass scaling is an explicit analysis technique in finite element simulations where the mass of certain elements is artificially increased.

- The main purpose is to lower the wave propagation speed ( $c_{\max}$ ), thereby allowing larger time steps according to the Courant stability (CFL) condition.

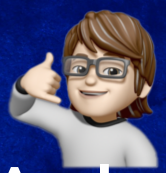
- Recall the CFL condition:

$$\Delta t \leq \frac{L_{\text{min}}}{c_{\text{max}}}, \quad c_{\text{max}} = \sqrt{\frac{E}{\rho}}$$

- By artificially increasing density $(\rho)$, wave speed decreases, increasing the allowable time step.
- However, mass scaling affects the accuracy of the dynamic response of the structure by reducing natural frequencies $(\omega = \sqrt{k/m})$, altering the frequency response of the structure.
- Mass inertial properties are distorted, potentially damping rapid dynamic responses or inaccurately modeling structural behavior under dynamic loads.
- Large mass scaling can significantly distort stress wave propagation, impact results, and energy dissipation, resulting in unrealistic or overly conservative results.



- To minimize the negative impacts of mass scaling on explicit solution accuracy:
  - Only apply mass scaling in local regions that are least critical to dynamic response accuracy (e.g., distant from critical impact zones).
  - Use the smallest mass increase necessary to achieve a practical increase in time step size, typically limiting the mass scaling factor to less than 10-20%.
  - Continuously monitor and check kinetic, internal, and total energy balances. Excessive mass scaling will manifest as unrealistic energy distributions or significant artificial kinetic energy.
  - Perform preliminary analyses to quantify the sensitivity of results to mass scaling. Adjust scaling to ensure acceptably small deviations from reference solutions.



## Practical Example of Mass Scaling for Explicit Analysis

- An example is a crash simulation where a portion of the vehicle structure away from the impact zone is overly refined with the initial time step without scaling: $0.1\mu s$, which is considered an impractical computational cost.
- By introducing a mass scaling factor of 1.5 ( $50\%$ mass increase) in these zones and increasing the allowable time step to $0.15\mu s$, the results can be simulated with reasonable time and cost.
- The effects of this mass scaling should be checked by comparing deformation patterns, peak accelerations, and energy distributions with and without mass scaling.
- Ensuring variations in key response parameters remain below acceptable engineering uncertainty tolerance (say <5-10%).
- Best practices are to limit mass scaling to non-critical areas rather than global scaling and to apply just enough scaling to achieve computational feasibility without sacrificing critical solution accuracy.
- Energy metrics and response parameters should be monitored and tracked throughout the simulation to ensure solution accuracy.

## Conclusions

- It is not always clear whether implicit or explicit time-stepping is preferred for structural dynamics applications.
- Care must be used to correctly apply these methods, especially explicit methods, to ensure the time-step is kept below the stability limit.
- If mass scaling is used to increase this step size, this must be examined carefully so that inaccuracies are within acceptable limits.

# References:


1. Thomas J.R. Hughes, (1987,2000). The Finite Element Method: Linear Static and Dynamic Finite Element Analysis. Dover Publications.

2. Hairer, E., & Wanner, G. (1996). Solving Ordinary Differential Equations II: Stiff and Differential-Algebraic Problems. Springer.