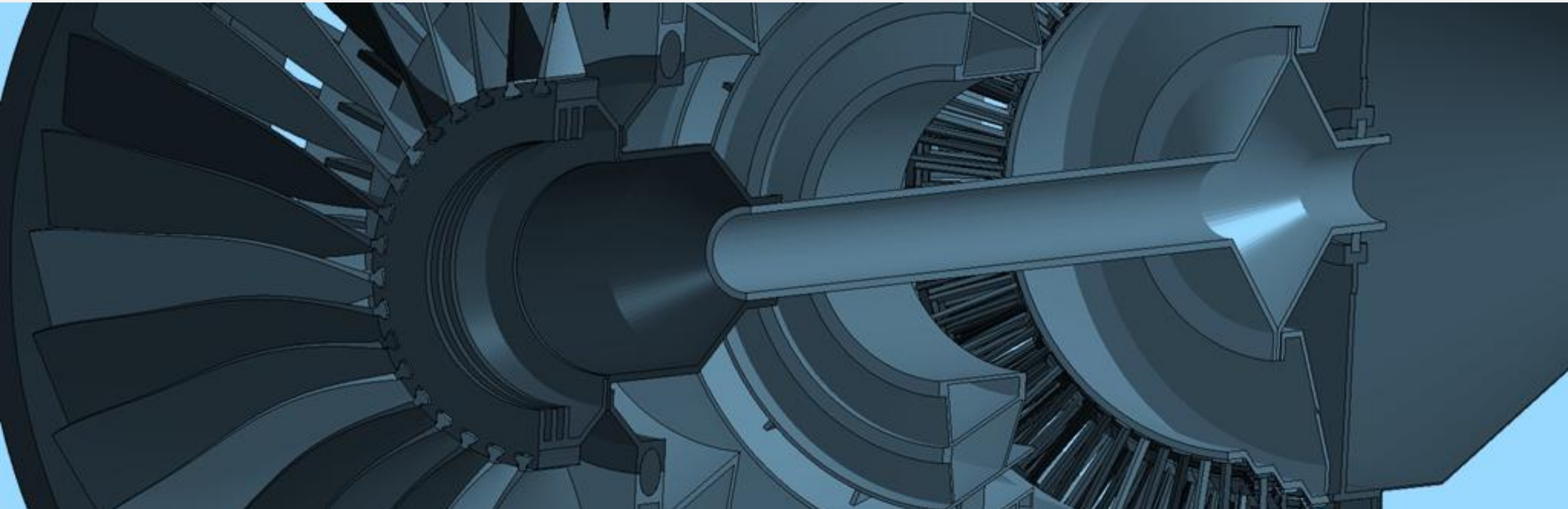
University Technology Centre (UTC)
Multidisciplinary Process Integration

Rolls-Royce

EUROPEAN UNION
European Regional Development Fund

Brandenburgische Technische Universität Cottbus - Senftenberg



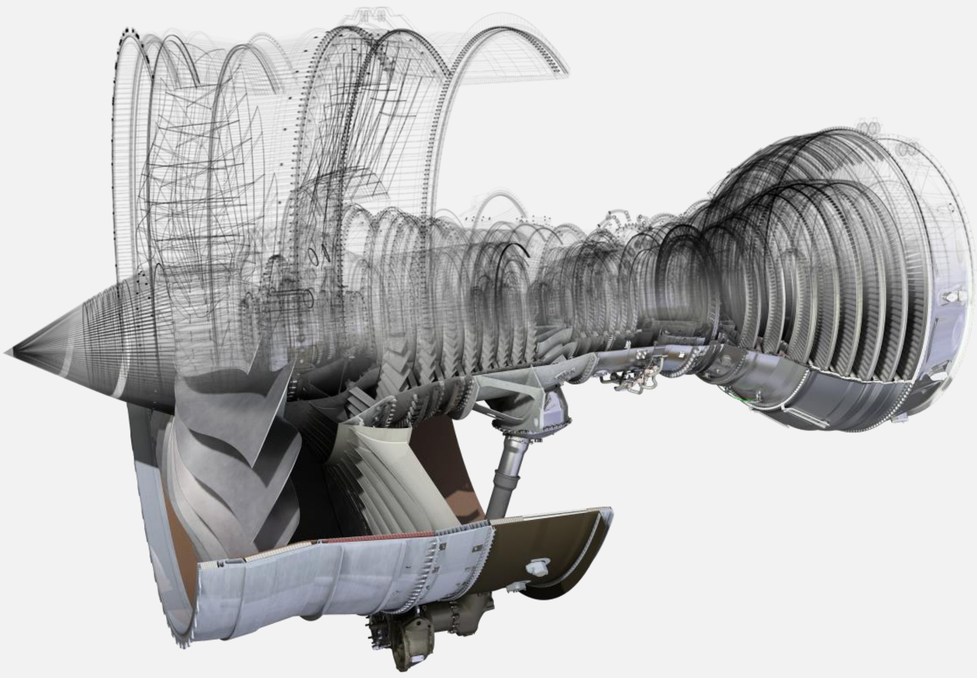
# Instability Problems of Implicit FEM Solution Procedures for Fast Rotating Structures – Instability Sources and Solutions

## WCCM 2018, New York


Chair of Structural Mechanics and Vehicle Vibrational Technology
Prof. Dr.-Ing. Arnold Kühhorn | Dr.-Ing. Markus Kober

Rolls-Royce plc.
Dr.-Ing. Akin Keskin

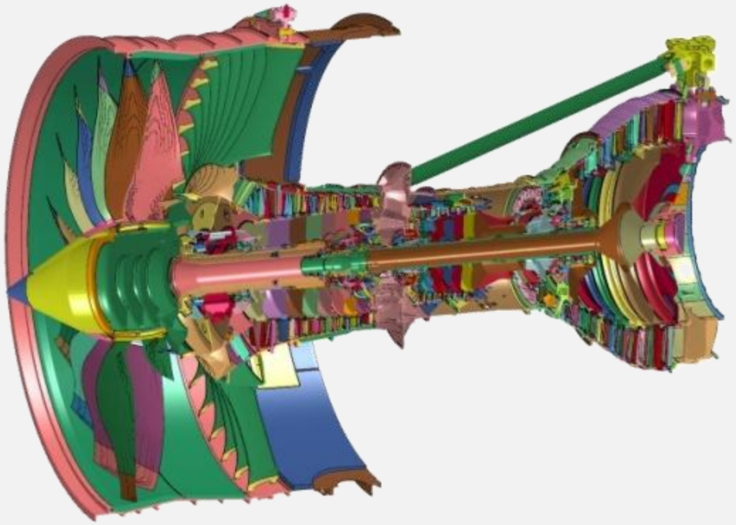
# Outline



- Motivation
- Description of problem
  - Spinning plate
- Stability considerations of time-integration algorithms
- Simplified model and reasons for instability
- Possible Solutions
- Complex FEM example
- Summary

## Objective

- In aerospace industry more and more detailed FEM models are used for prediction of engine behavior
- Goal is the thermo-mechanical simulation of a running engine with almost no idealizations or simplifications over a time-span of a few seconds



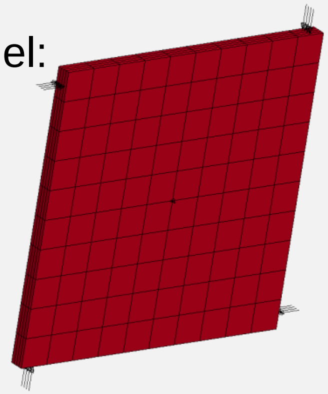
Dynamic models with millions of DOF
(including contact, non-linear material behavior
and huge displacements)

- But since the explicit analysis of even **40ms** of engine model takes **a few weeks on thousands of cores**, the **explicit analysis is not an option** for the simulation of a running engine over a few seconds

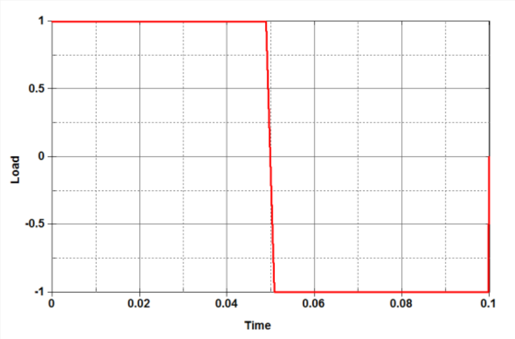
⇨ **Implicit time-integration could be a solution**

## Implicit Simulation of Spinning Plate with Newmark algorithm

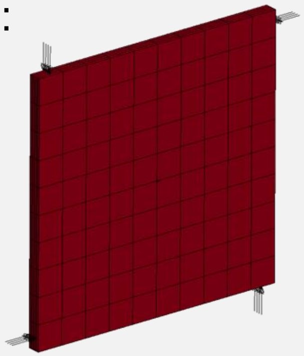
Model:

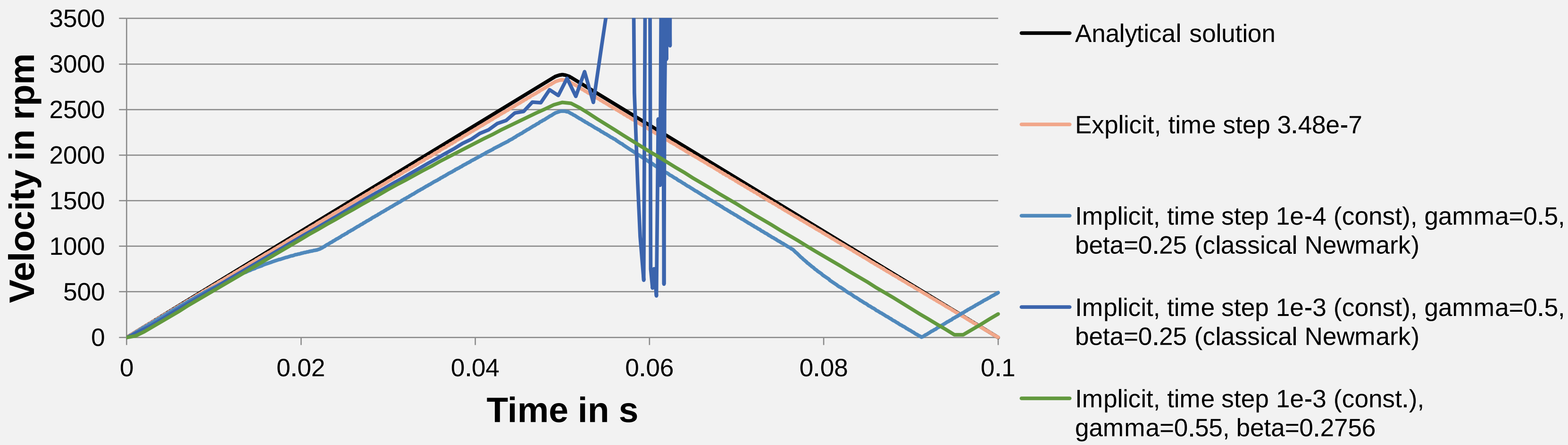


Load curve:




Simulation:



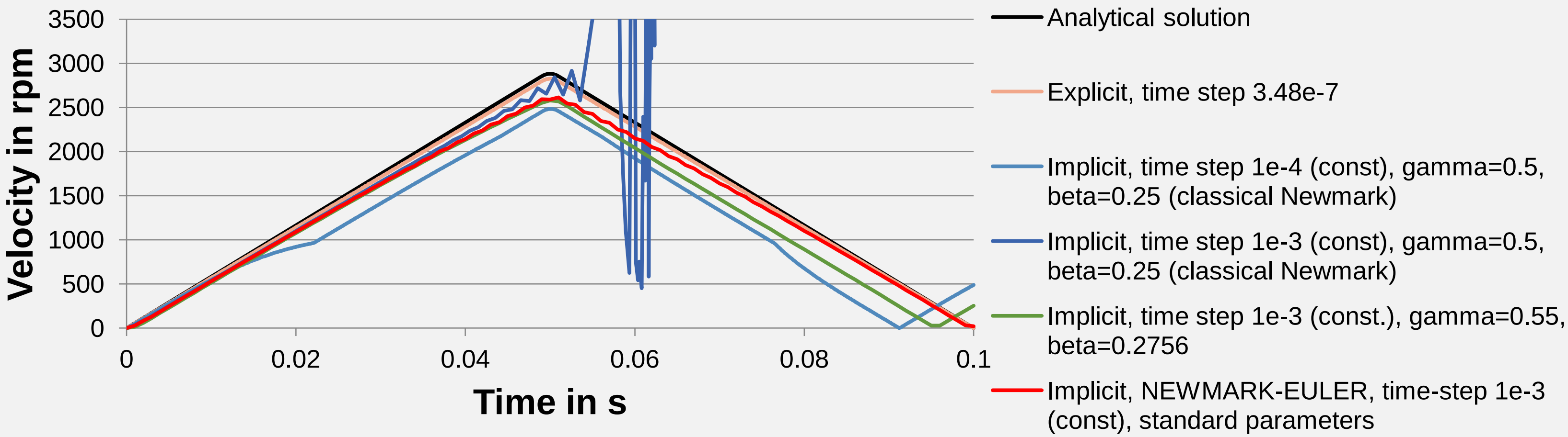



- Newmark algorithm becomes instable with standard parameters
- Change of parameters improves situation but new parameters are only valid for particular problem

# Description of Problem

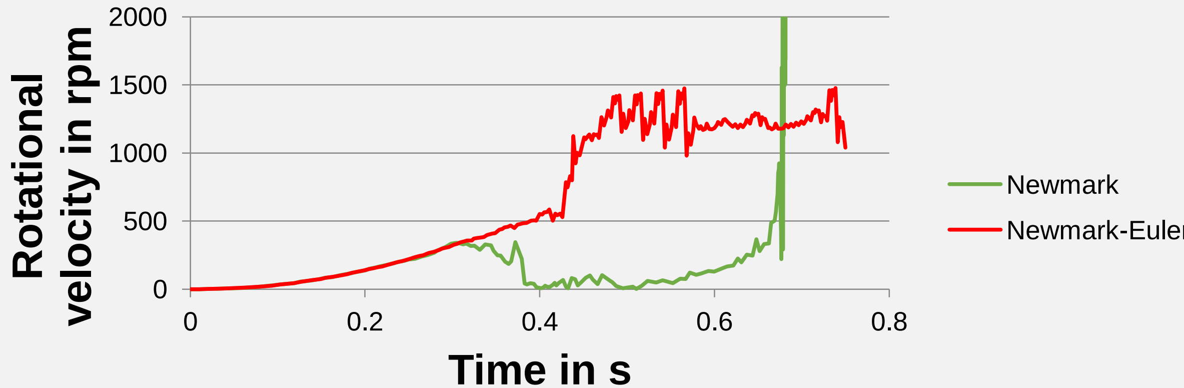
Brandenburgische Technische Universität Cottbus - Senftenberg

- Literature suggests more advanced time-integration methods like Newmark-Euler:




- Newmark-Euler works better than classical Newmark algorithm but also becomes instable in certain situations (automatic time-step control is used):

Driven by pressure load on blades



# Stability Considerations of Time-integration Algorithms

- Stability analysis is useful to discover possible weak points of an integration algorithm and to show the borders of its applicability in terms of stability
- Many time integration methods (e.g. Newmark, Newmark-Euler) can be written in the form

$$\begin{bmatrix} \ddot{x}_{n+1} \\ \dot{x}_{n+1} \\ x_{n+1} \end{bmatrix} = \mathbf{A} \cdot \begin{bmatrix} \ddot{x}_n \\ \dot{x}_n \\ x_n \end{bmatrix} + \mathbf{L}_{n+1} r$$

$\mathbf{A}$ = Matrix of integration approximation
$\mathbf{L}$ = Load operator
$r$ = External loads

- Matrix A depends on the time integration method (and the mechanical problem), that is used
- An integration method is stable if the **spectral radius** of the matrix $\mathbf{A}$ (depending on $\Delta$t/T) is always smaller or equal than 1

Spectral radius of $\mathbf{A}$: $\rho(\mathbf{A}) = \max|\lambda_i|, \quad i = 1, 2, 3$

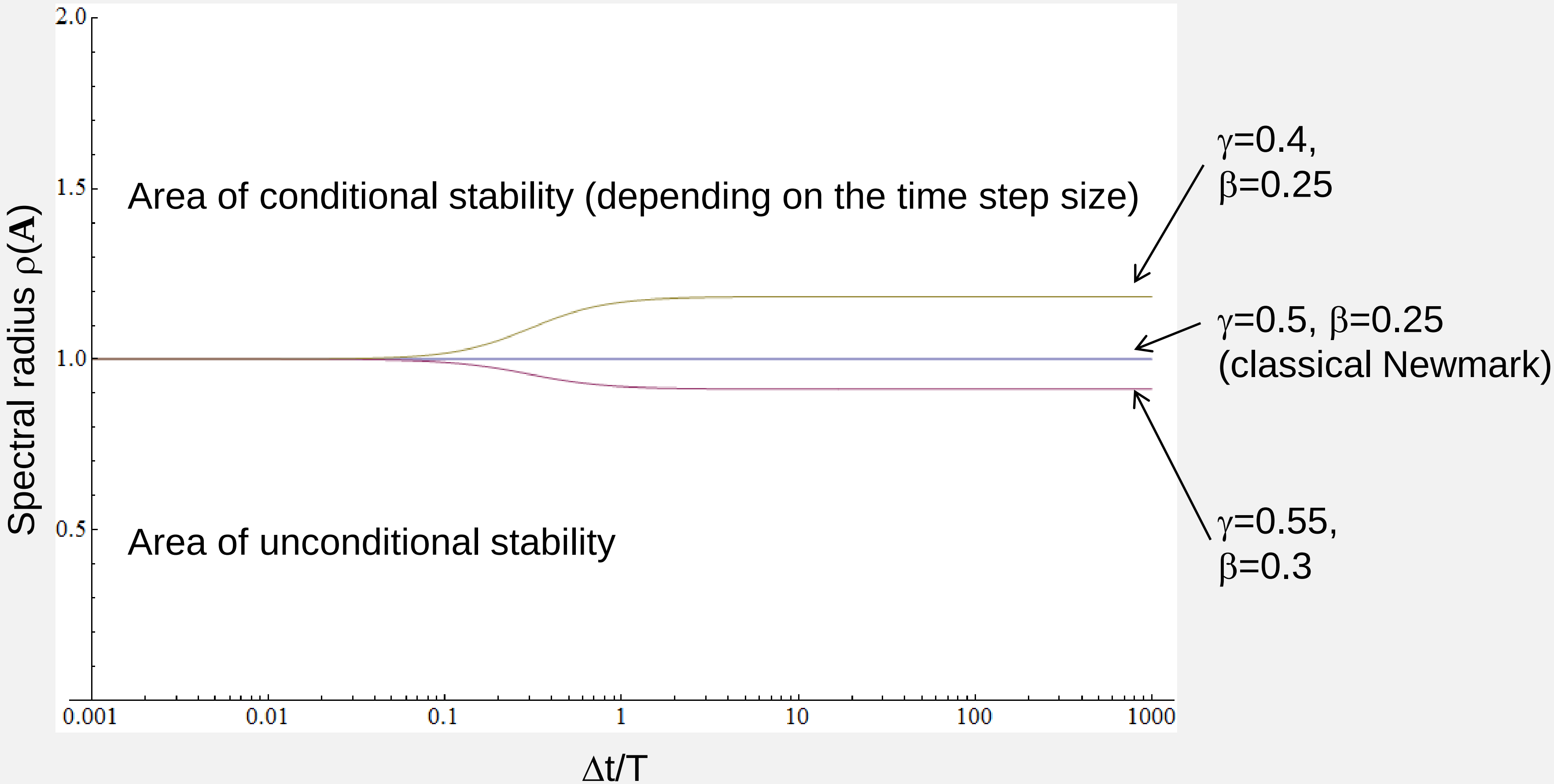
Stability criterion: $\rho(\mathbf{A}) \leq 1$

- For stability analysis it is sufficient to consider the simple undamped free vibration problem:

$\ddot{x} + \omega^2 x = 0$ => $r$ = 0

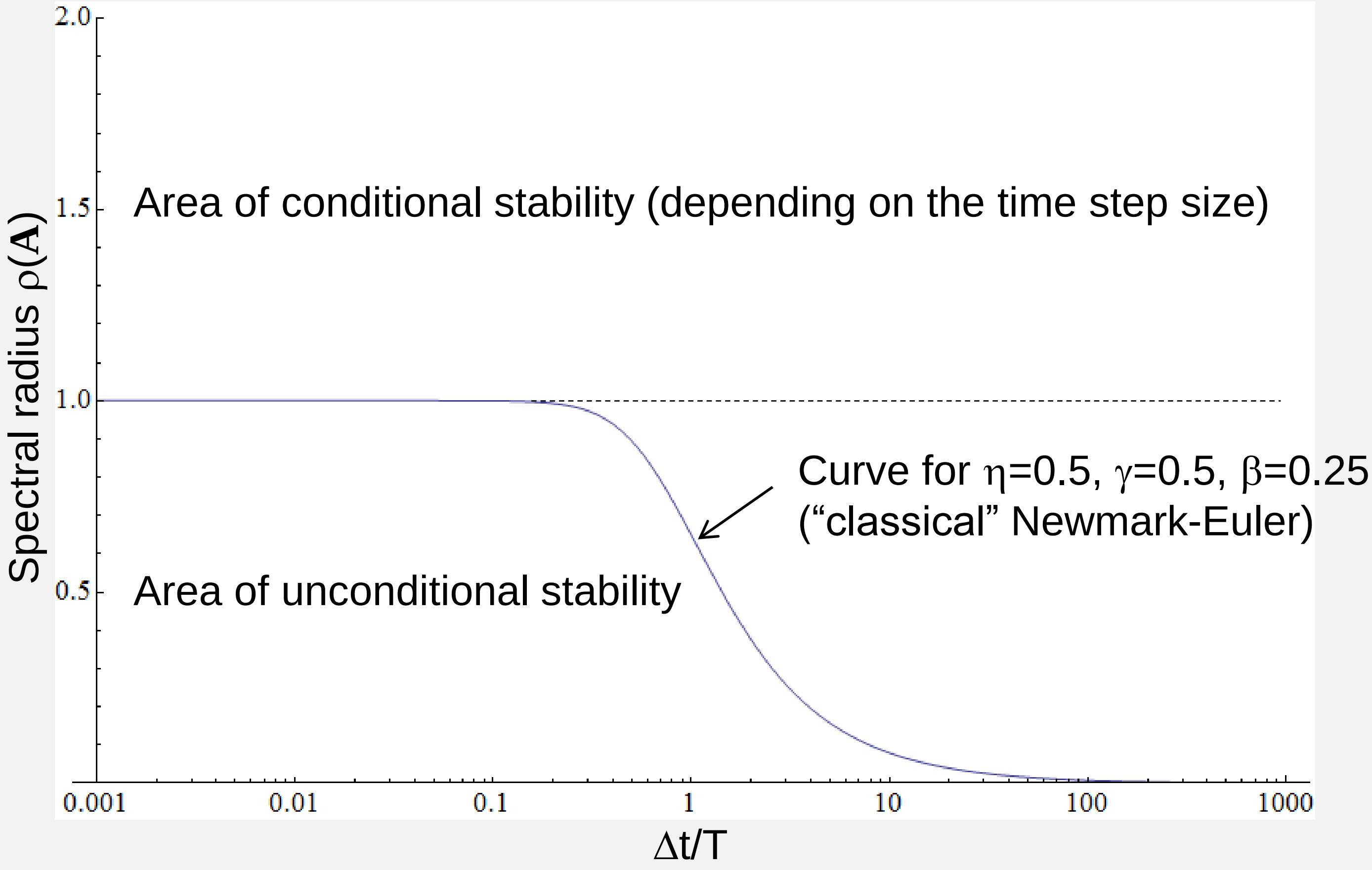
## Stability Map of Newmark algorithm

- With T = 2π/ω = 1 it is possible to compute a stability map (plot of spectral radius over Δt/T):



## Stability Map of Newmark-Euler algorithm

- With $T = 2\pi/\omega = 1$ it is possible to compute a stability map (plot of spectral radius over $\Delta t/T$):



# Stability Considerations of Time-integration Algorithms

- Stability analysis shows that neither the Newmark algorithm nor the Newmark-Euler algorithm become instable with their standard parameters for arbitrary big time-steps
- But used method for stability analysis is actually only valid for linear problems

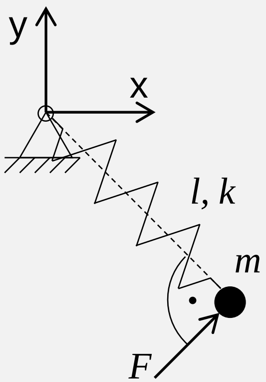
  => Time-step size has to be small enough (computation of every time-step can be considered to be linear)
- Here, time-steps are small enough
- For special cases (e.g. Newmark algorithm with standard parameters) methods from control theory can be used to show stability of algorithm also for nonlinear problems (large time steps)

**Time-integration algorithm cannot be the source of instability**

# Simplified Model and Reasons for Instability

## Rotating pendulum with spring (gravity neglected)

### Model and equation of motion in Cartesian coordinates




Equations of motion:

$$m\ddot{x} + F\cdot\frac{y}{\sqrt{x^2+y^2}} + kx\frac{\sqrt{x^2+y^2}-l_0}{\sqrt{x^2+y^2}} = 0$$

$$m\ddot{y} - F\cdot\frac{x}{\sqrt{x^2+y^2}} + ky\frac{\sqrt{x^2+y^2}-l_0}{\sqrt{x^2+y^2}} = 0$$

with $l_0$= non-elongated length of spring and k=stiffness of spring

Equations of motion with NEWMARK time-integration (for m=1kg and F=0N):

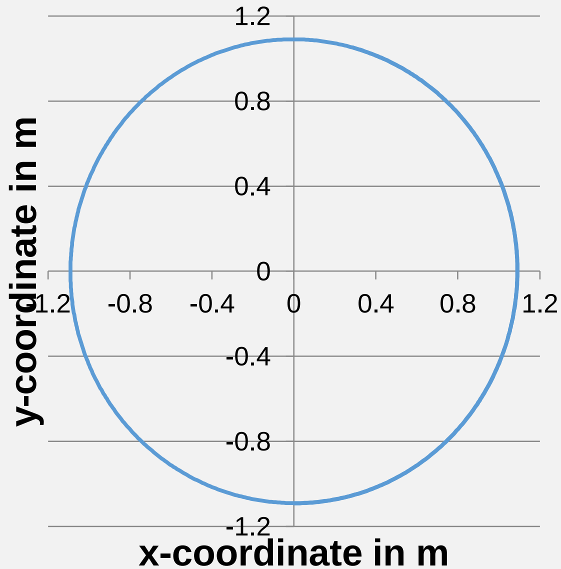
$$f_x = \frac{-2x_n + 2x_{n+1} + \Delta t\left(-2\dot{x}_n + (2\beta-1)\Delta t\ddot{x}_n\right)}{2\beta\Delta t^2} + kx_{n+1}\frac{\sqrt{x_{n+1}^2+y_{n+1}^2}-l_0}{\sqrt{x_{n+1}^2+y_{n+1}^2}} = 0 \qquad f_y = \frac{-2y_n + 2y_{n+1} + \Delta t\left(-2\dot{y}_n + (2\beta-1)\Delta t\ddot{y}_n\right)}{2\beta\Delta t^2} + ky_{n+1}\frac{\sqrt{x_{n+1}^2+y_{n+1}^2}-l_0}{\sqrt{x_{n+1}^2+y_{n+1}^2}} = 0$$

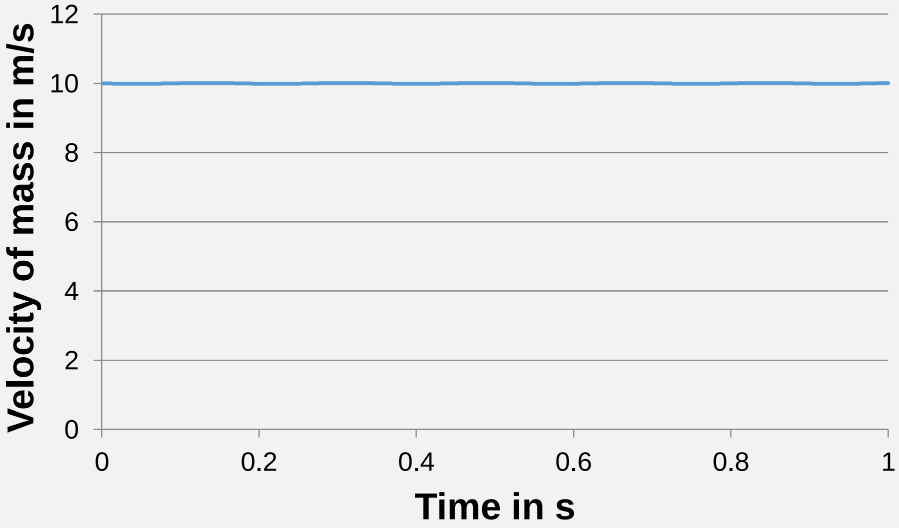
Solution by Newton iteration:

$$\begin{bmatrix} x_{n+1}^{k+1} \\ y_{n+1}^{k+1} \end{bmatrix} = \begin{bmatrix} x_{n+1}^{k} \\ y_{n+1}^{k} \end{bmatrix} - J^{-1}\left(x_{n+1}^k, y_{n+1}^k\right)\cdot\begin{bmatrix} f_x\left(x_{n+1}^k, y_{n+1}^k\right) \\ f_y\left(x_{n+1}^k, y_{n+1}^k\right) \end{bmatrix} \quad with \quad J = \begin{bmatrix} \dfrac{\partial f_x}{\partial x_{n+1}} & \dfrac{\partial f_x}{\partial y_{n+1}} \\ \dfrac{\partial f_y}{\partial x_{n+1}} & \dfrac{\partial f_y}{\partial y_{n+1}} \end{bmatrix}$$

# Simplified Model and Reasons for Instability

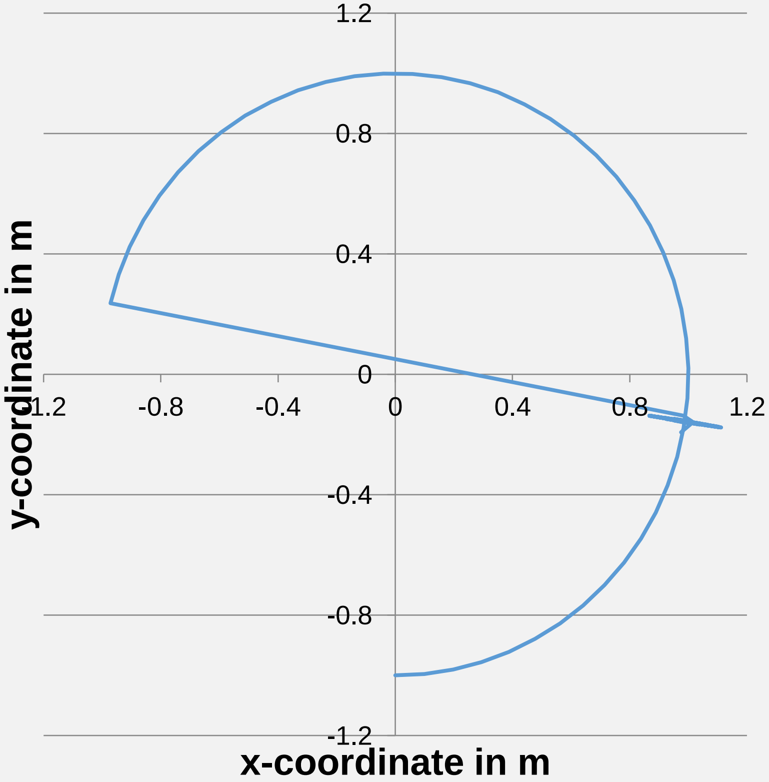
Results for
**Δt=0.001s** and $t_{max}$=1s ($\gamma$=0.5, $\beta$=0.25) with $l(0)$=1.09161m**, *k* =1000N/m**, $l_0$=1m, $v_o$=10m/s:




**What happens if stiffness k and time-step size Δt are increased?**

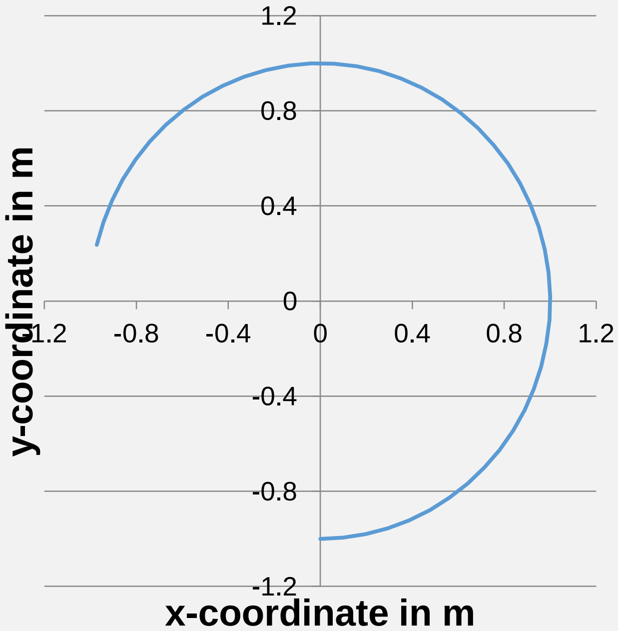
Increase of stiffness from **k=1000N/m** to **k=$10^7$N/m**
and
increase of time-step size from **Δt=0.001s** to **Δt=0.01s**



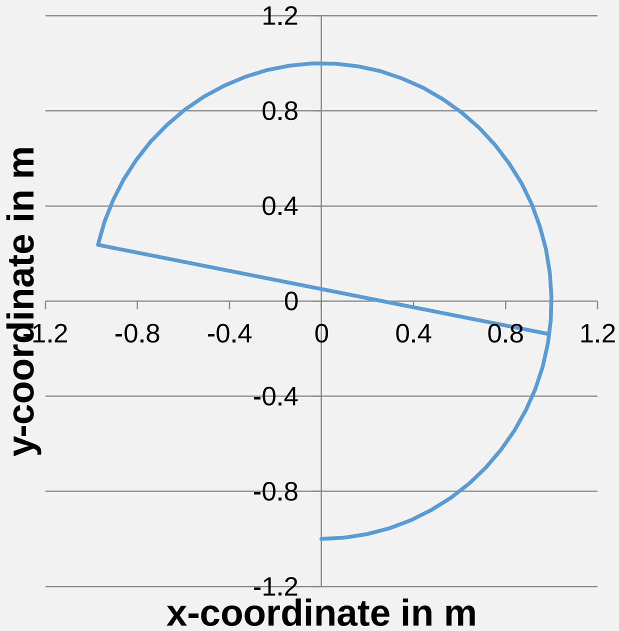

**For Δt=0.01s and k=$10^7$N/m instability appears!**

Brandenburgische Technische Universität Cottbus - Senftenberg

- Consideration of situation of rotating pendulum with spring at the time step before instability ($\Delta t$=0.01s and k=$10^7$N/m):




Before instability (t=0.45s)




After instability (t=0.46s)

## What is the numerical situation right before instability?

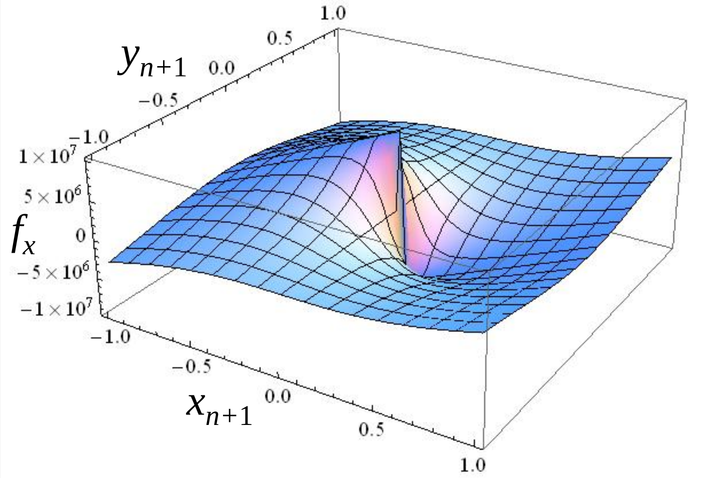
- NEWMARK equations that have to be solved for $x_{n+1}$ and $y_{n+1}$ by Newton's algorithm:

$$f_x = \frac{-2x_n + 2x_{n+1} + \Delta t\left(-2\dot{x}_n + (2\beta - 1)\Delta t \ddot{x}_n\right)}{2\beta\Delta t^2} + kx_{n+1}\frac{\sqrt{x_{n+1}^2 + y_{n+1}^2} - l_0}{\sqrt{x_{n+1}^2 + y_{n+1}^2}} = 0$$

$$f_y = \frac{-2y_n + 2y_{n+1} + \Delta t\left(-2\dot{y}_n + (2\beta - 1)\Delta t \ddot{y}_n\right)}{2\beta\Delta t^2} + ky_{n+1}\frac{\sqrt{x_{n+1}^2 + y_{n+1}^2} - l_0}{\sqrt{x_{n+1}^2 + y_{n+1}^2}} = 0$$

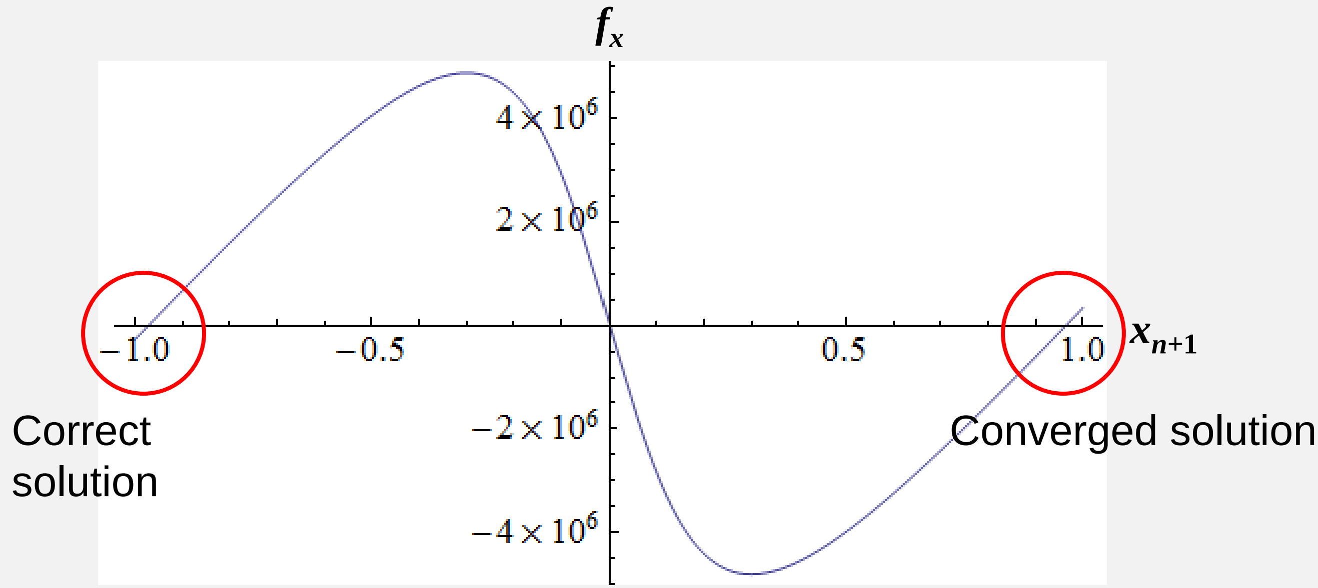
- At t=0.45s:

$$\beta = 0.25, \Delta t = 0.01\, s, k = 10^7\ N/m,$$
$$x_n = -0.9717077973201222\ m,$$
$$y_n = 0.23614647740490619\ m,$$
$$\dot{x}_n = -2.3517560831270767\ m/s,$$
$$\ddot{x}_n = -91.33130927254165\ m/s^2,$$



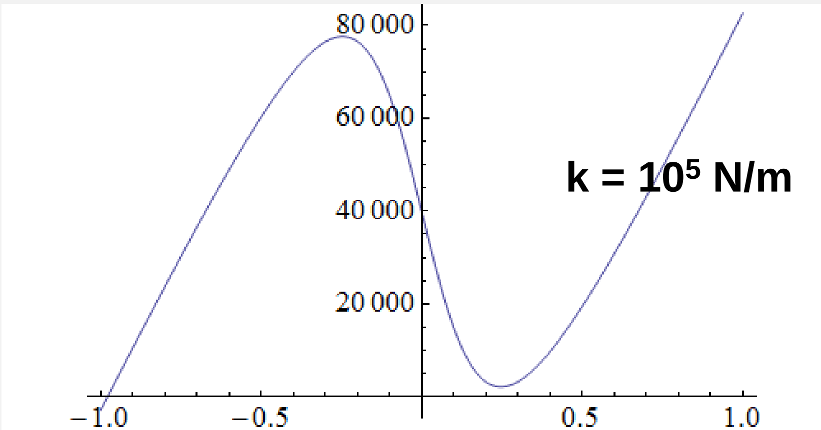
# Simplified Model and Reasons for Instability

Cut through 3D diagram at $y_{n+1} = y_n = 0.23614647740490619\ m$:



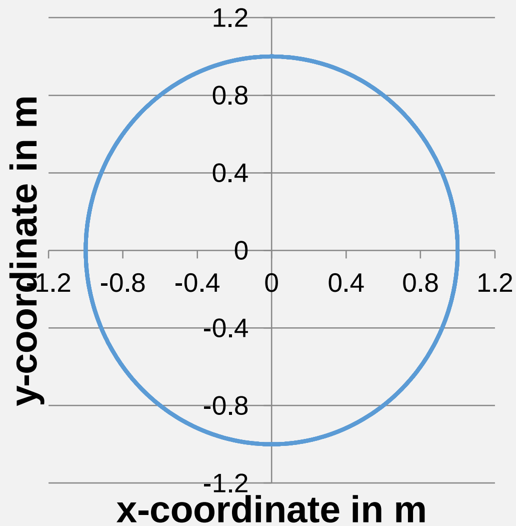

- Obviously problem has 3 zeros
- Newton iteration converges into wrong equilibrium as solution for Newmark equation
- “Instability” is just a problem of the Newton iteration
- Between $-1 < x_{n+1} < 1$ the slope of the function $f_x$ is very big, which probably causes numerical problems when finding the zero at $x_{n+1} = -1$

## Influence of parameters k and Δt to number of zeros of function $f_x$




- No parameters changed except k
- **k decreased** from $10^7$N/m to $10^5$N/m

**Only one zero**

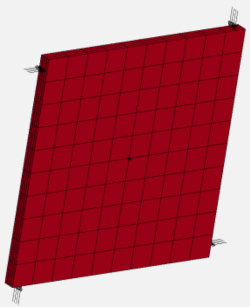


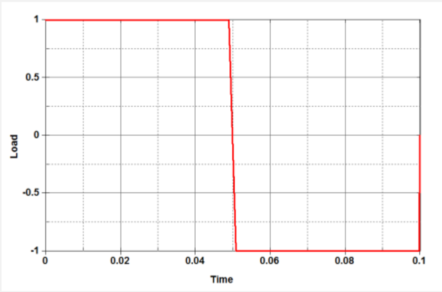

**No “instability” even after 50s of simulation time!**

Brandenburgische Technische Universität Cottbus - Senftenberg

## FEM Example

- By **changing the stiffness** of the model or the time-step size, **stability can be reached** in combination with the Newmark algorithm
- This can also be shown at the spinning plate example:

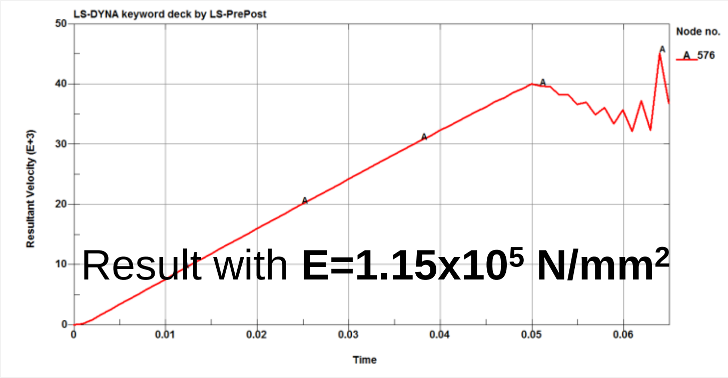




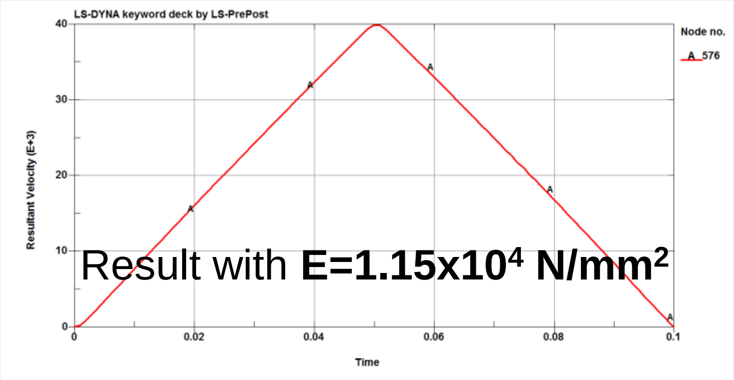

Load curve

In analogy to explicit dynamics a „**stiffness scaling**“ is possible!

Velocity vs. time plots:



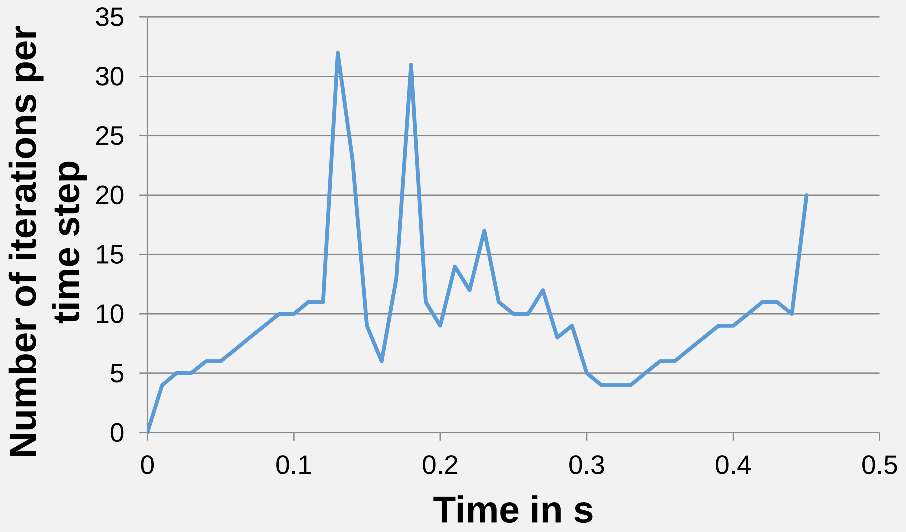

Result with **E=1.15x10$^5$ N/mm$^2$**




Result with **E=1.15x10$^4$ N/mm$^2$**

# Possible Solution

**Number of iterations per time step (for k=$10^7$ N/m and $\Delta t$=0.01s)**




- Number of time steps per iteration is very high
- Actually the Newton algorithm converges fast or something is wrong

**Solution: Improved time-step control**

- If the Newton algorithm needs more than xx iterations per time-step, the time-step size should be reduced
- If the number of iterations per time-step is lower than a certain value, the time-step size should be increased

# Possible Solution

Brandenburgische Technische Universität Cottbus - Senftenberg

## Results with simple algorithm for time-step control

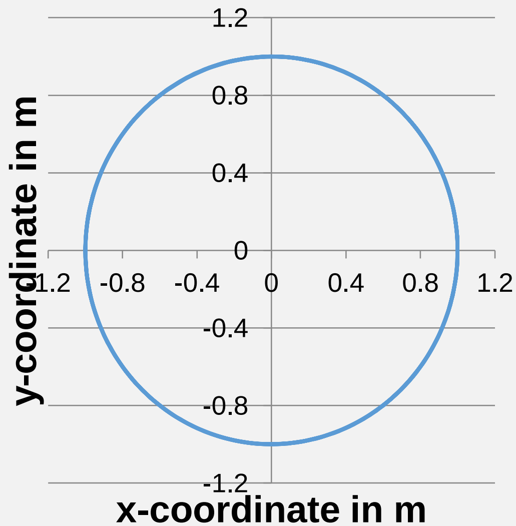
- Implementation: If number of iterations is bigger than 9, time-step size is reduced by

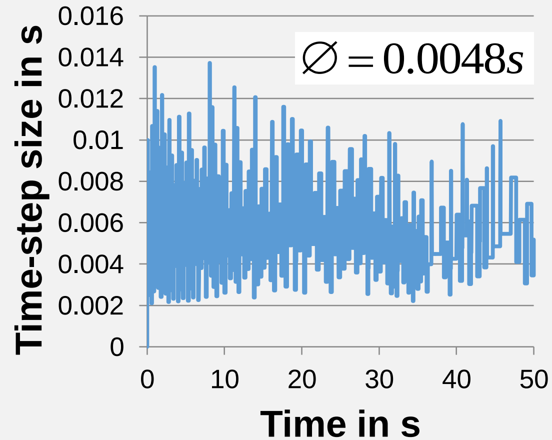
$$\Delta t = \Delta t - 0.5 \cdot \Delta t$$

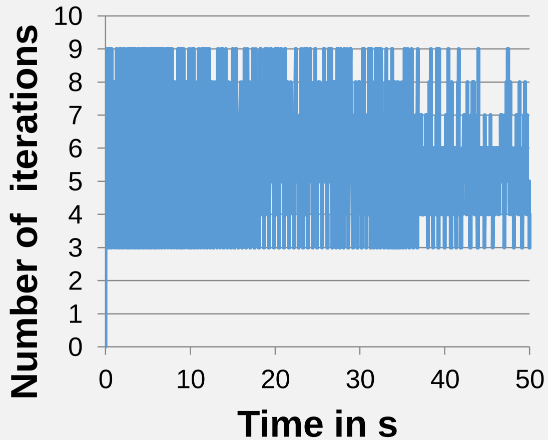
and **(very important!)** result of last time increment (which needed more than 9 iterations) is deleted and Newton algorithm is repeated with smaller time-step. If number of iterations per time step is smaller than 4, time-step size is increased by

$$\Delta t = \Delta t + 0.5 \cdot \Delta t$$

Results for k=$10^7$ N/m, simulation time = 50s, time-step control as above:



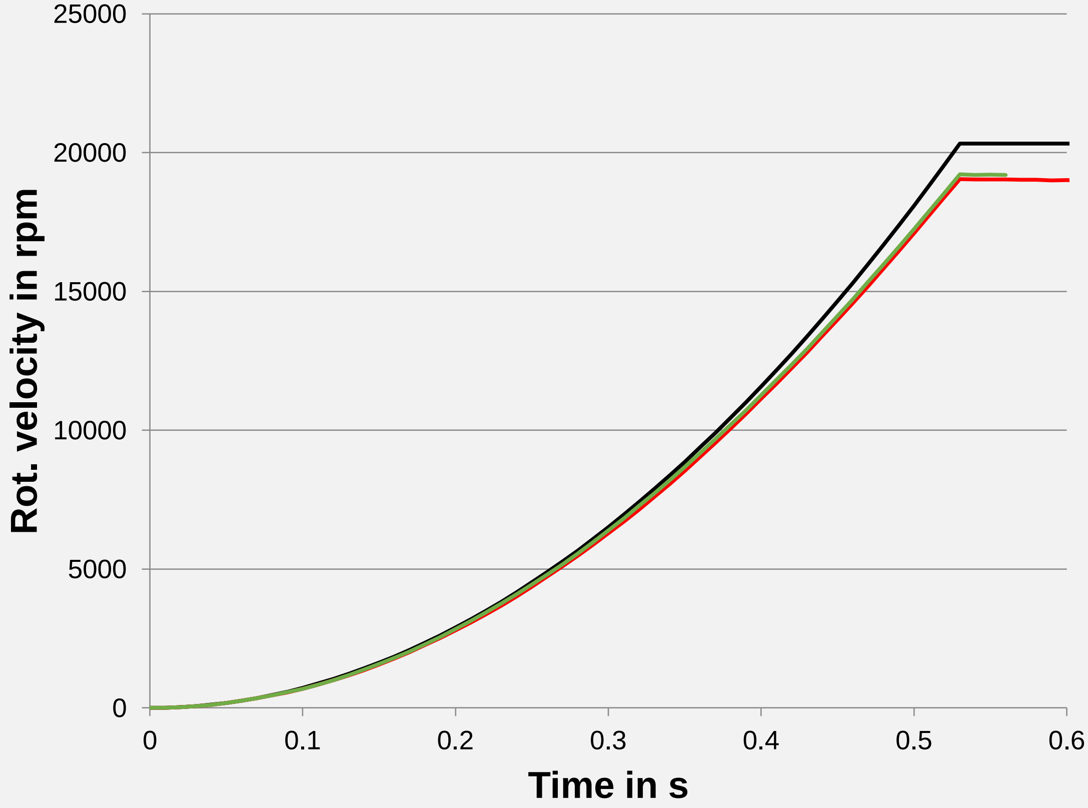
# Possible Solution

- Also other tests for convergence for Newton algorithm are possible:

  Check for monotonicity of iterated results:
  - Error may not increase during an equilibrium iteration
  - If error of previous Newton iteration is smaller than the current one, last result is deleted and time-step is decreased by
    $\Delta t = \Delta t - 0.5 \cdot \Delta t$
  - If number of iterations per time-step is smaller than 5 the time-step is increased by
    $\Delta t = \Delta t + 0.5 \cdot \Delta t$

- This strategy has also been implemented and tested successfully but seems to be slower (10046 function evaluations for 1s of simulation time compared to 1332 function evaluations with the previous strategy, 750 function evaluations for constant time-step size of 0.01s with instability)

## Solution with standard time-step control but changed parameters



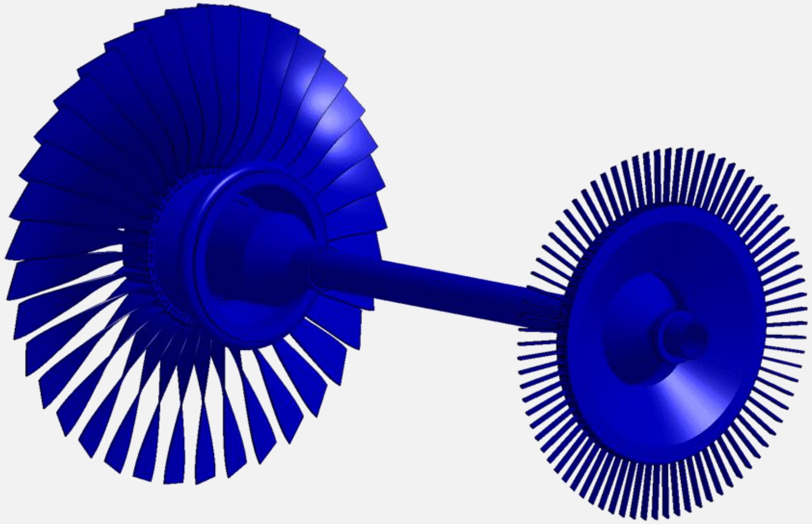

- Differences between “analytical” solution and FEM simulation are caused by deformations due to centrifugal loads which are not considered in the “analytical “solution

# Summary

- Instability issues of implicit simulation of flexible rotating structures can now be explained
- Issues are not caused by time-integration scheme (e.g. Newmark time-integration) but by numerical problems of the Newton iteration
- Also improved time-integration schemes like Newmark-Euler and HHT suffer from the same instability problems of the Newton iteration
- Correctness of findings has also been demonstrated for simple and more complex FEM simulations
- Solution for instability problems is an improved time-step control algorithm
  - ✓ Successfully tested in Python script
  - ✓ as well as in commercial FE codes

# Thank you for your attention!

Brandenburgische Technische Universität Cottbus - Senftenberg



EUROPEAN UNION

European Regional Development Fund

**Dr. Markus Kober**
Chair of Structural Mechanics and Vehicle Vibrational Technology
Prof. Dr.-Ing. Arnold Kühhorn
Brandenburg University of Technology Cottbus-Senftenberg

b-tu.de/fg-strukturmechanik

**Dr. Akin Keskin**
Rolls-Royce plc.

Acknowledgment

This research was funded within the support program ProFIT by the “Investitionsbank des Landes Brandenburg” with financial resources of the European Regional Development Fund and supported by Rolls-Royce plc.