Automated Solution of Equations with Uncertain Parameters

Andrzej Pownuk

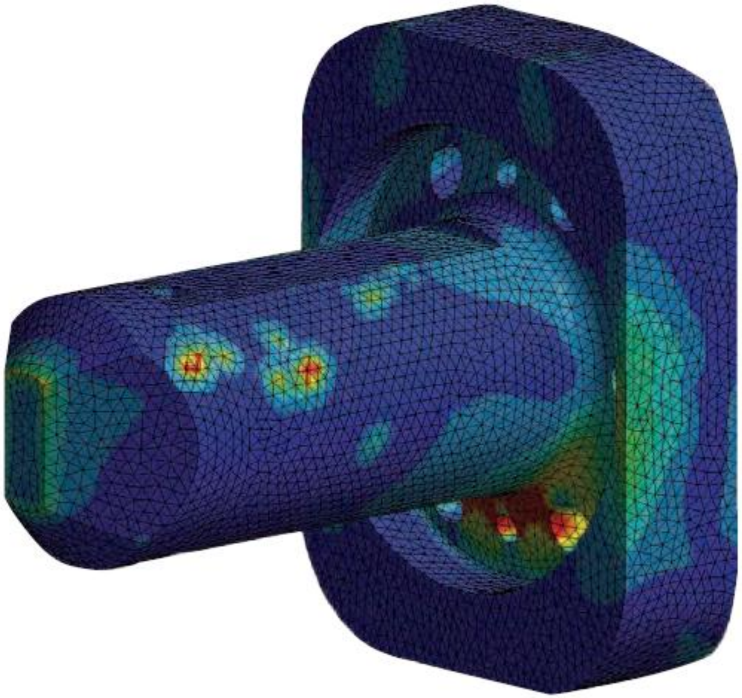
The University of Texas at El Paso

http://andrzej.pownuk.com

Outline of the presentation

- Equations with the uncertain parameters and their applications
- New approach for the solution of the equations with the interval parameters
- Generalizations and conclusions

Mathematical model of a machine



$$\begin{cases} \sum_{j=1}^{3} \frac{\partial \sigma_{ij}}{\partial x_j} + \rho f_i = \rho \frac{\partial^2 u_i}{\partial t^2} \\ \sigma_{ij} = \sum_{k=1}^{3} \sum_{j=1}^{3} C^{ijkl} \varepsilon_{kl} \\ \varepsilon_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \\ u_i = u_i^*, x \in \partial V_u \\ \sum_{j=1}^{3} \sigma^{ij} n_j = t_i^*, x \in \partial V_\sigma \\ u_i = u_i(x,0), x \in V \end{cases}$$

Such simulations are possible since early 1970s

O.C. Zienkiewicz, Ivo M. Babuška, P.G. Ciarlet, P. Solin,…

Mathematical models

physical problem

$$
\begin{aligned}
\sigma_{ij,j} + F_i &= 0 \text{ in } \Omega, \\
\epsilon_{ij} &= \frac{1}{2}(u_{j,i} + u_{i,j}), \\
\sigma_{ij} &= C_{ijkl}\,\epsilon_{kl}.
\end{aligned}
$$

mathematical models

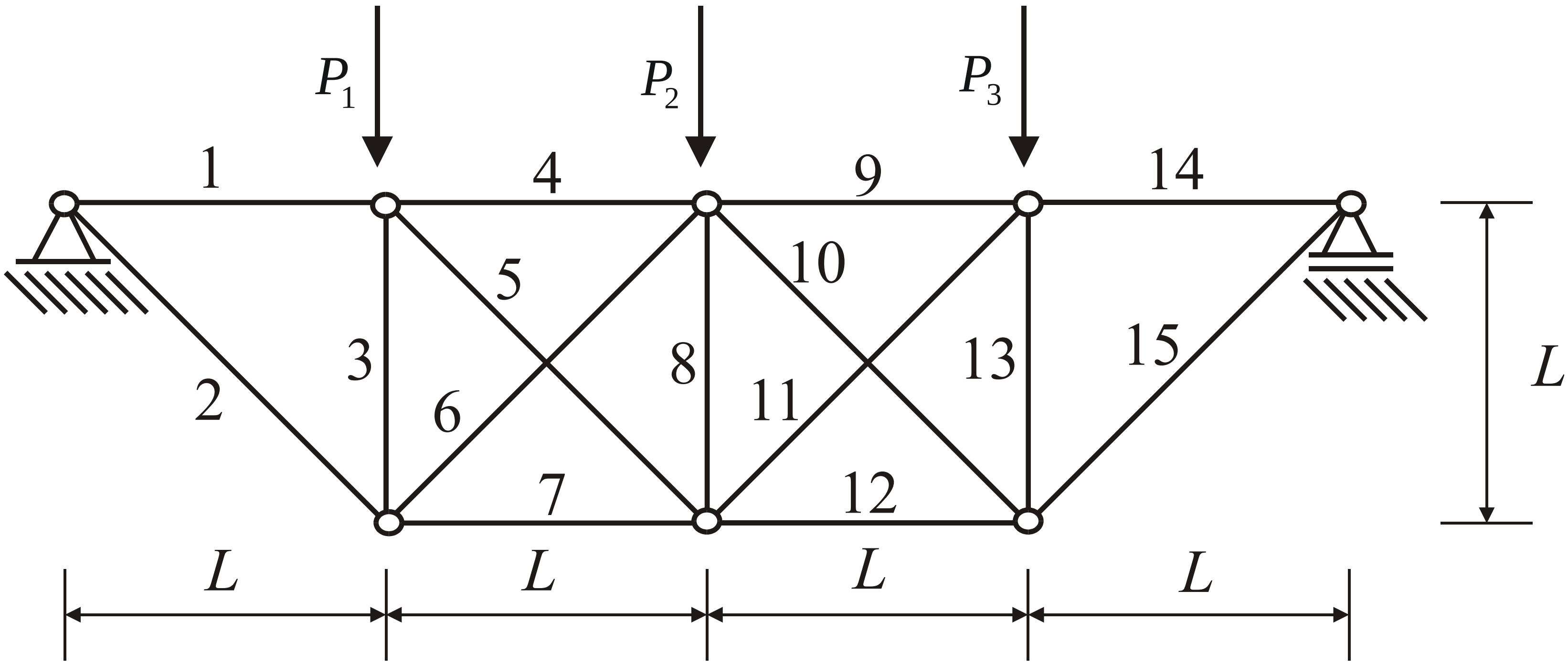
cheap

expensive

experiments

experimental results ≈ predictions

Truss structure with uncertain forces

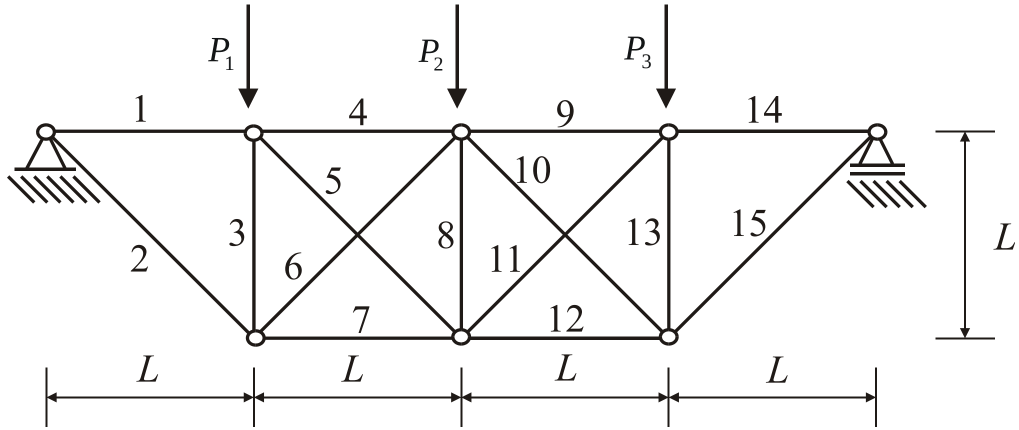


Perturbated forces

$$P = P_0 \pm \Delta P$$

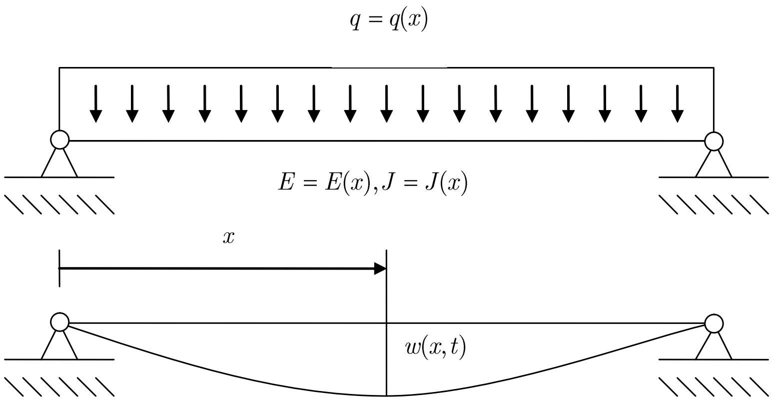
5% uncertainty

No	1	2	3	4	5	6	7	8
ERROR %	10	9,998586	10,00184	10,00126	60,18381	11,67825	9,998955	31,8762
No	9	10	11	12	13	14	15	
ERROR %	10,00126	11,67825	60,18381	9,998955	10,00184	10	9,998586	



Vibrations of beams

$$
\begin{cases}
EJ\dfrac{\partial^4 w}{\partial x^4} = q - \rho A \dfrac{\partial^2 w}{\partial t^2} \\
w(0,t) = 0 \\
w(L,t) = 0 \\
\dfrac{\partial^2 w(0,t)}{\partial x^2} = 0 \\
\dfrac{\partial^2 w(L,t)}{\partial x^2} = 0 \\
w(x,0) = w_0(x) \\
v(x,0) = \dfrac{\partial w(x,0)}{\partial t} = v_0(x) \\
E \in \mathbf{E}, q \in \mathbf{q}, A \in \mathbf{A}
\end{cases}
$$



Vibrations of beams

vibration.mpeg

Vibrations with uncertain parameters

vibrations-uncertainty.mpeg

Random vibrations

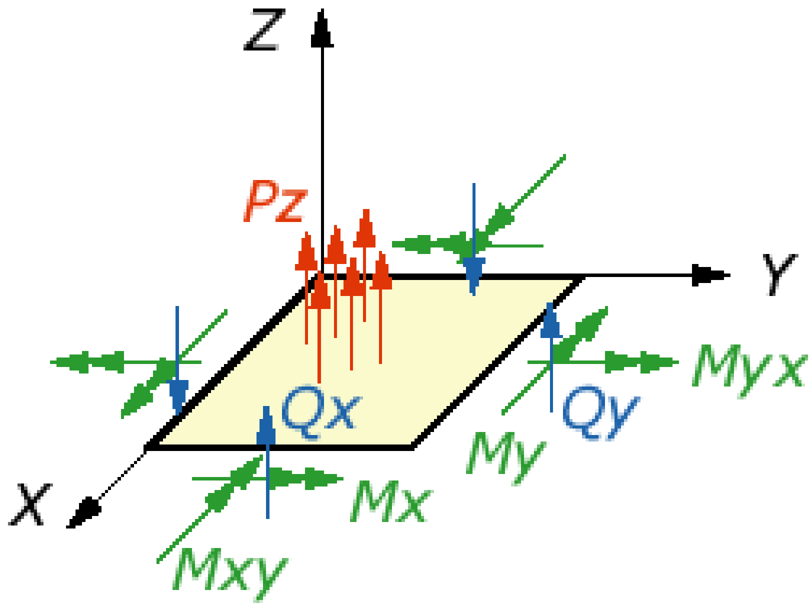
vibrations-random.mpeg

Interval displacements

vibrations-interval.mpeg

Dynamics of plates

$$
\begin{cases}
D\left(\dfrac{\partial^4 u}{\partial x^4} + 2\dfrac{\partial^4 u}{\partial x^2 \partial y^2} + \dfrac{\partial^4 u}{\partial y^4}\right) = q - \rho h \dfrac{\partial^2 u}{\partial t^2} \\
u(0,y,t) = 0 \\
u(L,y,t) = 0 \\
u(x,0,t) = 0 \\
u(x,L,t) = 0 \\
\dfrac{\partial^2 u}{\partial x^2}(0,y,t) = 0 \\
\dfrac{\partial^2 u}{\partial x^2}(L,y,t) = 0 \\
\dfrac{\partial^2 u}{\partial y^2}(x,0,t) = 0 \\
\dfrac{\partial^2 u}{\partial y^2}(x,L,t) = 0 \\
u(x,y,0) = u^*(x,y) \\
\dfrac{\partial u(x,y,0)}{\partial t} = v^*(x,y) \\
E \in \mathbf{E}, q \in \mathbf{q}, h \in \mathbf{h}
\end{cases}
$$



Dynamics of plates

plate-vibrations.mpeg

Uncertain solution

- Set-valued parameters

$$u(x,t,\mathbf{p}) = \{u(x,t,p) : p \in \mathbf{p}\}$$

$$u(x,t,\mathbf{p}) = [\underline{u}(x,t), \overline{u}(x,t)]$$

Uncertain solution

- Set-valued parameters and the optimization methods

$$\underline{u}(x,t) = \min\{u(x,t,p) : p \in \mathbf{p}\}$$

$$\overline{u}(x,t) = \max\{u(x,t,p) : p \in \mathbf{p}\}$$

Uncertain solution

- Set-valued parameters

$$\underline{u}(x,t) = u(x,t,p^{min})$$

$$\overline{u}(x,t) = u(x,t,p^{\max})$$

$$p^{\min} = \begin{cases} \arg\min\limits_{p} u(x,t,p) \\ s.t.\, p \in \mathbf{p} \end{cases}, \quad p^{max} = \begin{cases} \arg\max\limits_{p} u(x,t,p) \\ s.t.\, p \in \mathbf{p} \end{cases}$$

Example

$$u = p_1^2 + p_2^2$$

$$p_1 \in [1,2], p_2 \in [2,3], \mathbf{p} = [1,2] \times [2,3]$$

$$\underline{u} = u(\underline{p}_1, \underline{p}_2) = 1^2 + 2^2 = 5$$

$$\overline{u} = u(\overline{p}_1, \overline{p}_2) = 2^2 + 3^2 = 13$$

$$u(\mathbf{p}) = \left[\underline{u}, \overline{u}\right] = \left[5, 13\right]$$

Example

$$u = p_1^2 + p_2^2$$

$$p_1 \in [1,2], p_2 \in [2,3], \mathbf{p} = [1,2] \times [2,3]$$

$$p^{min} = (\underline{p}_1, \underline{p}_2) = (1,2), p^{\max} = (\overline{p}_1, \overline{p}_2) = (2,3)$$

$$\underline{u} = u(p^{min}) = 1^2 + 2^2 = 5$$

$$\overline{u} = u(p^{\max}) = 2^2 + 3^2 = 13$$

Probabilistic solution

- Random parameters

$$p:\Omega \ni \omega \rightarrow p(\omega) \in R^m$$

$$u(x,t):\Omega \ni \omega \rightarrow u(x,t,p(\omega)) \in R^n$$

Interval and probabilistic solution

- Probabilistic interpretation of the interval solution.

$$[\underline{y}, \overline{y}] = \{f(x) : x \in [\underline{x}, \overline{x}]\}$$

$$P\{\omega : Y(\omega) \in [\underline{y}, \overline{y}]\} = 1$$

Set-valued and probabilistic solution

$$P_{\Omega}\left\{\omega : u(x, y, p(\omega)) \in [\underline{u}(x,t), \overline{u}(x,t)], \omega \in \Omega\right\} = 1$$

Interval and probabilistic solution

- Width of the solution
  - Interval solution (worst case analysis)

$$width(nX) = n \cdot width(X)$$

  - Probabilistic solution

$$width(nX) = \sqrt{n} \cdot width(X)$$

Taylor method

$$w_{i,j}\left(p\right) \approx w_{i,j}\left(p_0\right) + \sum_k \frac{\partial w_{i,j}}{\partial p_k}(p_0)(p_k - p_0)$$

$$\underline{w}_{i,j} \approx w_{i,j}\left(p_0\right) - \sum_k \left|\frac{\partial w_{i,j}}{\partial p_k}(p_0)\right| \Delta p_k$$

$$\overline{w}_{i,j} \approx w_{i,j}\left(p_0\right) + \sum_k \left|\frac{\partial w_{i,j}}{\partial p_k}(p_0)\right| \Delta p_k$$

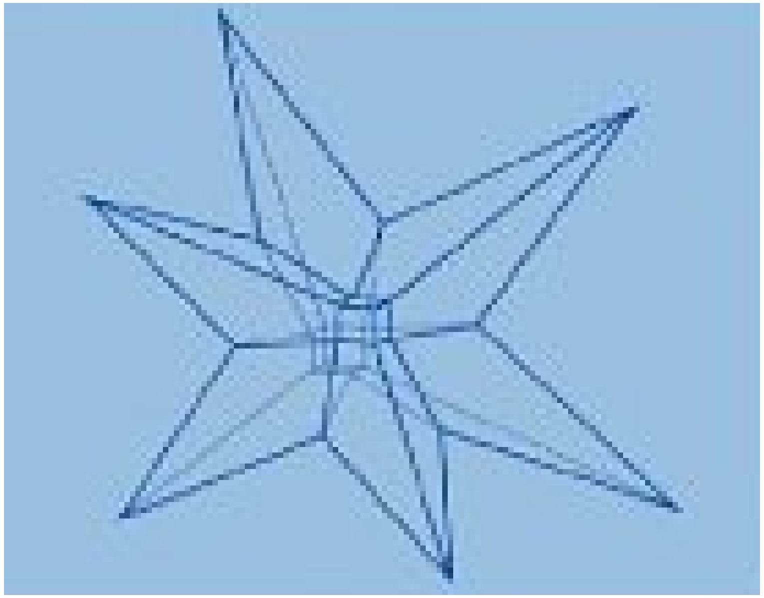
Graphical representation of the solution

$$GraphDiscrete\left(w,\left[0,L\right],\left[0,T\right],n_x,n_t,\boldsymbol{p}\right)=$$

$$=\left\{\begin{bmatrix}\left(w_{0,0}(p),x_0,t_0\right) & \cdots & \left(w_{0,n_t}(p),x_0,t_{n_t}\right)\\ \vdots & \vdots & \vdots\\ \left(w_{n_x,0}(p),x_{n_x},t_0\right) & \cdots & \left(w_{n_x,n_t}(p),x_{n_x},t_{n_t}\right)\end{bmatrix}:p\in\boldsymbol{p}\right\}.$$

System of linear interval equations

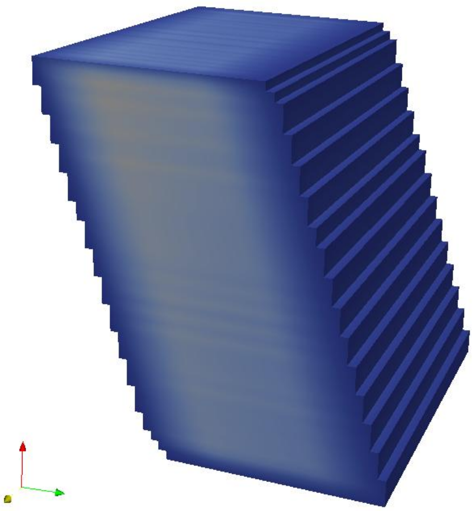
$$\mathbf{A}x = \mathbf{b}$$

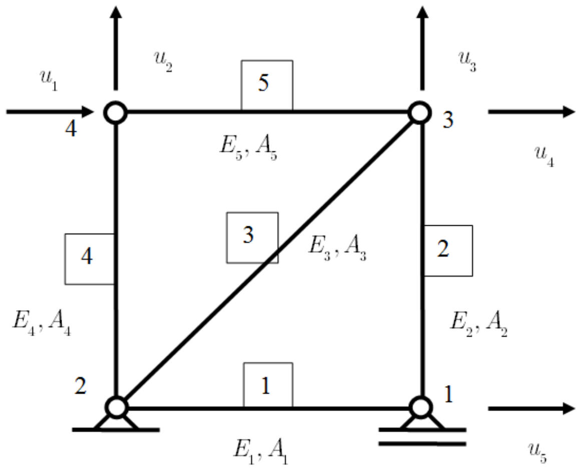


$$x(\mathbf{A},\mathbf{b}) = \{x : Ax = b, A \in \mathbf{A}, b \in \mathbf{b}\}$$

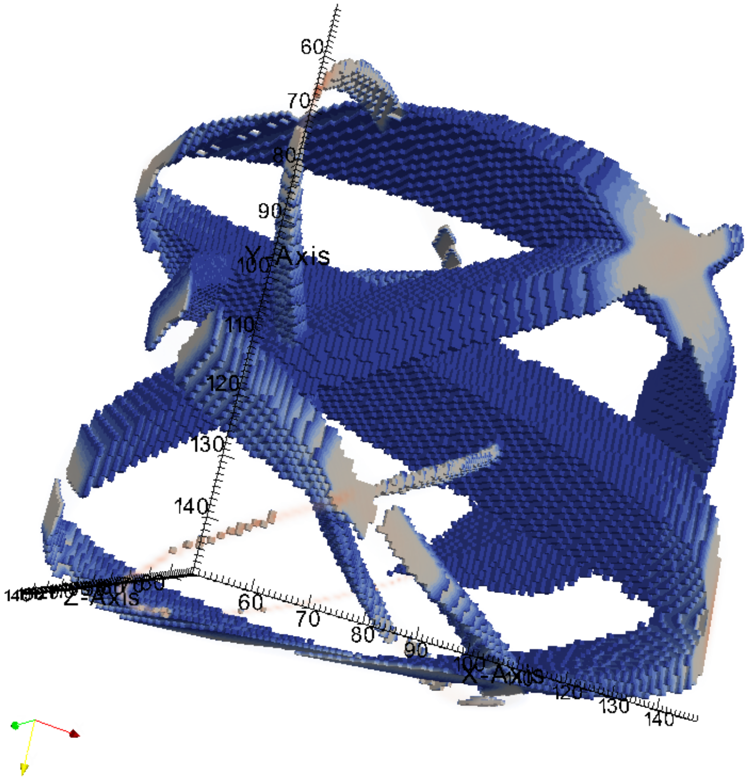
Solution set in 3D

$$u(\mathbf{p}) = \left\{ \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} \dfrac{2\sqrt{2}P_1}{A_3E_3} + \dfrac{P_1}{A_5E_5} + \dfrac{P_1 + P_3}{A_2E_2} \\ -\dfrac{P_2}{A_4E_4} \\ -\dfrac{P_1 + P_3}{A_2E_2} \end{bmatrix} : E_i \in \left[\underline{E}_i, \overline{E}_i\right], P_i \in \left[\underline{P}_i, \overline{P}_i\right] \right\}$$





Solution set in 3D



$$\begin{cases} \dfrac{dx}{dt} = pr\cos\left(pt\right) \\ \dfrac{dy}{dt} = -pr\sin\left(pt\right) \\ \dfrac{dx}{dt} = p \\ x(0) = 0 \\ y(0) = 1 \\ z(0) = 0 \end{cases} \quad , p \in \left[\underline{p}, \overline{p}\right], r \in \left[\underline{r}, \overline{r}\right]$$

Uncertainty

Problem with real parameters

$$2x = 4$$

$2 \in [1,3]$

$$x = \frac{4}{2} = 2$$

$4 \in [3,5]$

Problem with interval parameters

$$[1,3]x = [3,5]$$

$$x = ?$$

Algebraic Solution

$$[1,2]x = [1,4]$$

$$x = [1,2]$$

because

$$[1,2] \cdot [1,2] = [1,4]$$

United Solution Set

$$[1,2]x=[1,4]$$

$$\mathbf{x}=\frac{[1,4]}{[1,2]}=\left[\frac{1}{2},4\right]$$

because

$$\mathbf{x}=\{x:ax=b,a\in[1,2],b\in[1,4]\}$$

Comparison of the solution sets

$$[1,2]x=[1,4]$$

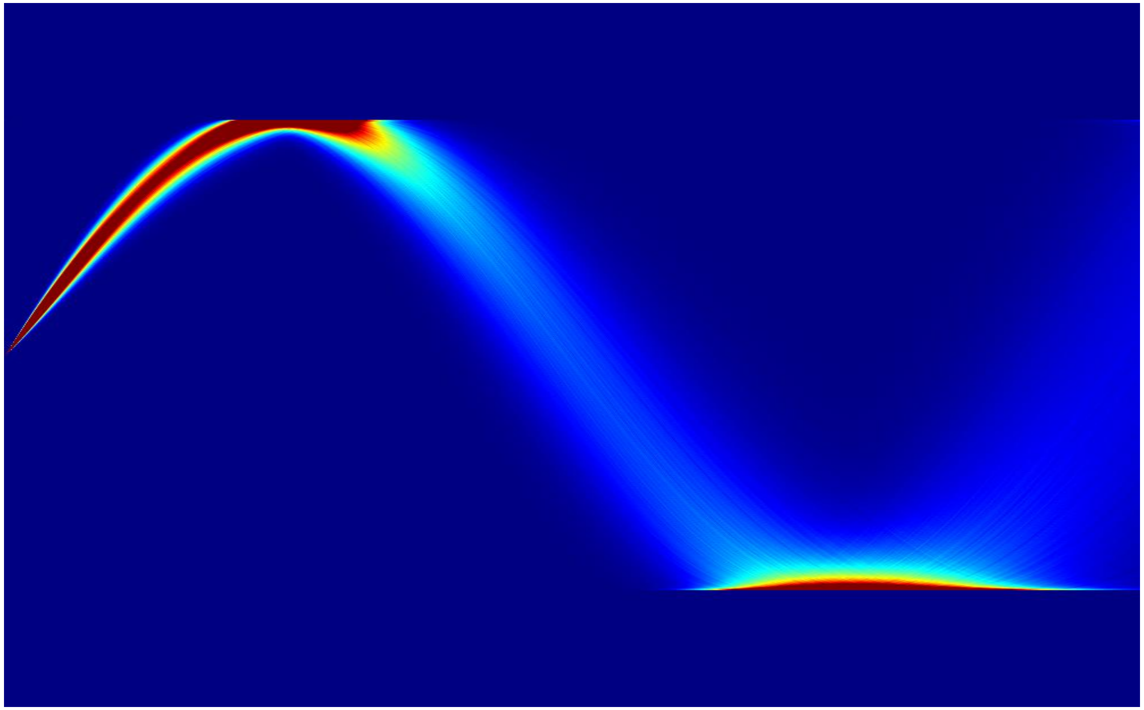
$$\mathbf{x}=[1,2] \quad \neq \quad \mathbf{x}=\frac{[1,4]}{[1,2]}=\left[\frac{1}{2},4\right]$$

Algebraic Solution **United Solution Set**

There are many ways how it is possible
to extend equations with the real parameters
into equations with the interval parameters.

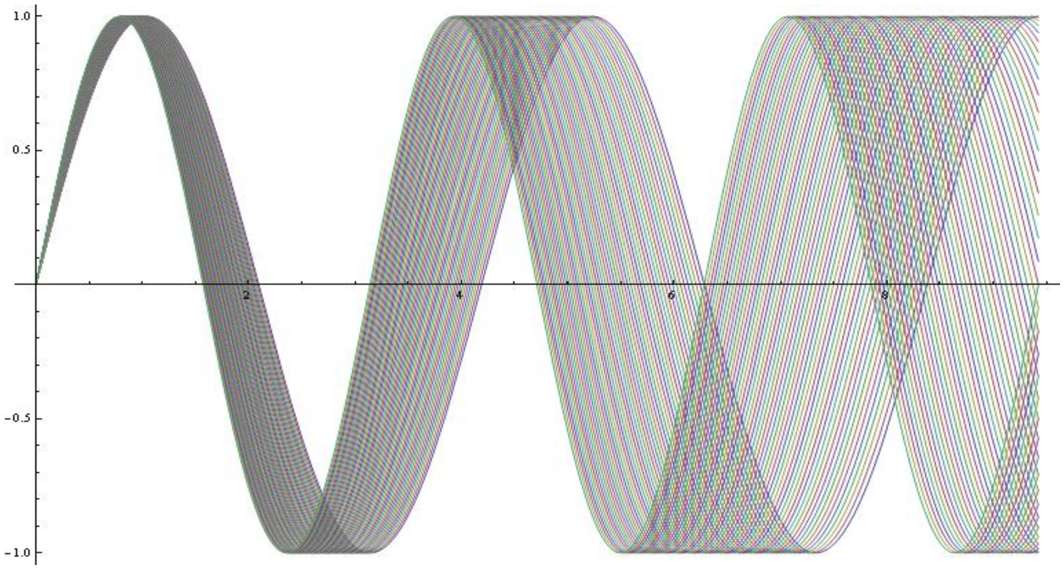
Stochastic differential equations

$$\begin{cases} y' = p\cos(pt) \\ y(0) = 0 \\ p \sim N(0,1) \end{cases}$$



Interval equation

$$\begin{cases} y' = p\cos(pt) \\ y(0) = 0 \\ p \in \left[\underline{p}, \overline{p}\right] \end{cases}$$



Practical applications

- Presented approach can be applied for the solution of practical engineering problems.
- It is possible to solve nonlinear and large scale problems.

Post-processing

- PDE

$$L(p)u = f(p)$$

- Set-valued solution

$$K(p)u = Q(p), p \in \mathbf{p} \Rightarrow u(\mathbf{p})$$

- Post-processing

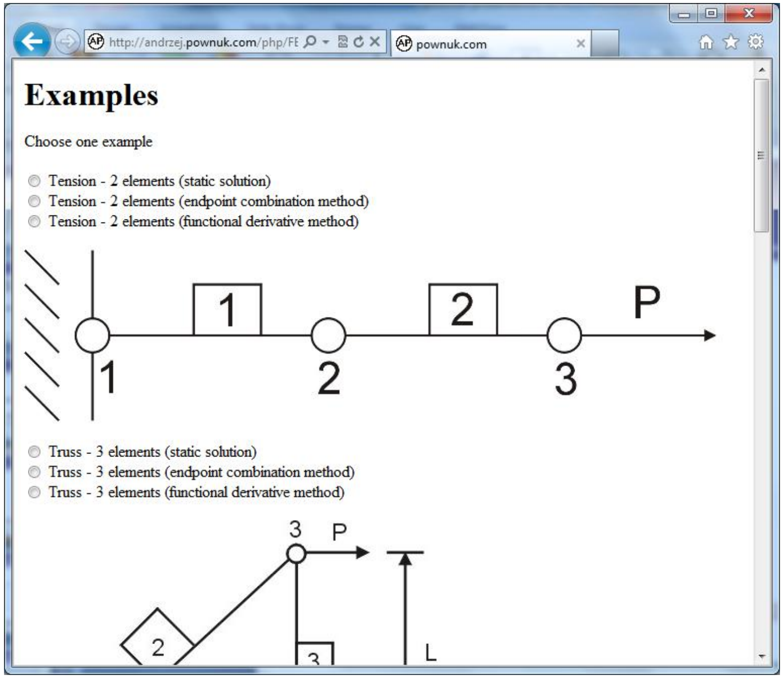
$$\begin{cases} \varepsilon = Du \\ \sigma = C\varepsilon \end{cases} \Rightarrow \begin{cases} \varepsilon(\mathbf{p}) \neq Du(\mathbf{p}) \\ \sigma(\mathbf{p}) \neq C\varepsilon(\mathbf{p}) \end{cases}$$

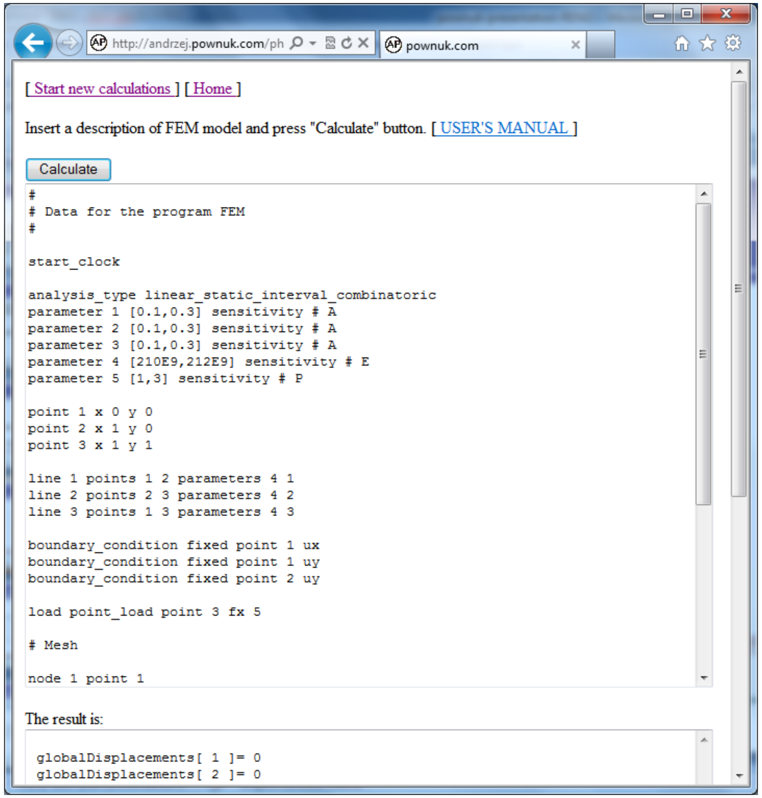
Post-processing

$$K(p)u = Q(p) \Rightarrow K(p)\frac{\partial u}{\partial p} = \frac{\partial Q(p)}{\partial p} - \frac{\partial K(p)}{\partial p}u \Rightarrow \frac{\partial u}{\partial p}$$

$$\begin{cases} \varepsilon = Du \\ \sigma = C\varepsilon \end{cases} \Rightarrow \begin{cases} \dfrac{\partial \varepsilon}{\partial p} = \dfrac{\partial D}{\partial p}u + D\dfrac{\partial u}{\partial p} \\ \dfrac{\partial \sigma}{\partial p} = \dfrac{\partial C}{\partial p}\varepsilon + C\dfrac{\partial \varepsilon}{\partial p} \end{cases}$$

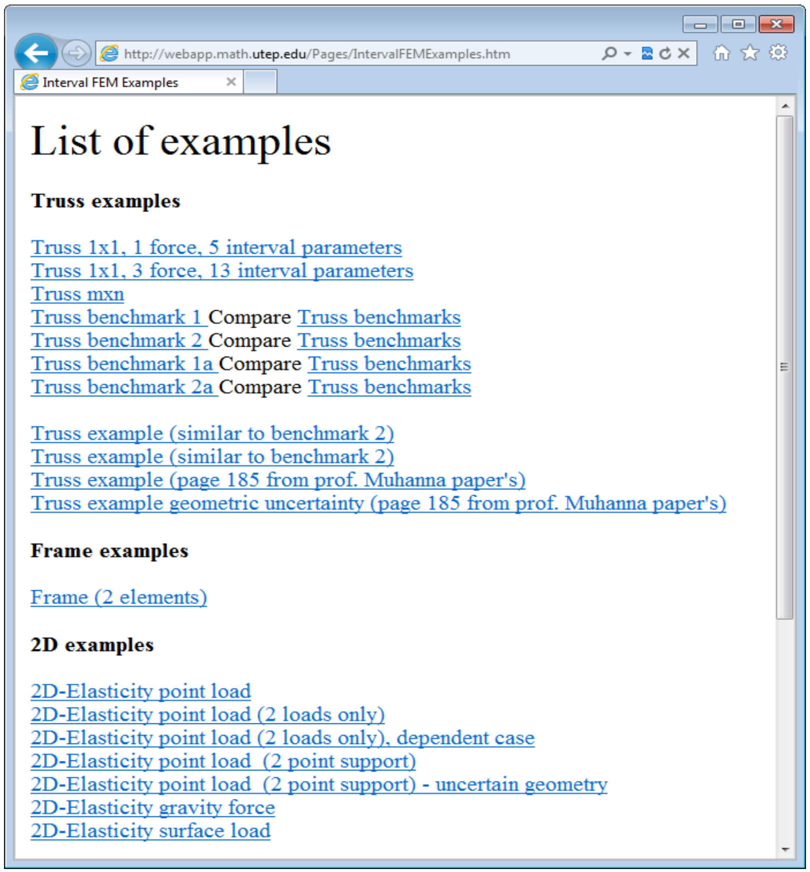
General interval FEM





http://andrzej.pownuk.com/php/FEM2/

Automatically generated test problems


http://webapp.math.utep.edu/Pages/IntervalFEMExamples.htm

Interval FEM Examples

List of examples

Truss examples

Truss 1x1, 1 force, 5 interval parameters
Truss 1x1, 3 force, 13 interval parameters
Truss mxn
Truss benchmark 1 Compare Truss benchmarks
Truss benchmark 2 Compare Truss benchmarks
Truss benchmark 1a Compare Truss benchmarks
Truss benchmark 2a Compare Truss benchmarks

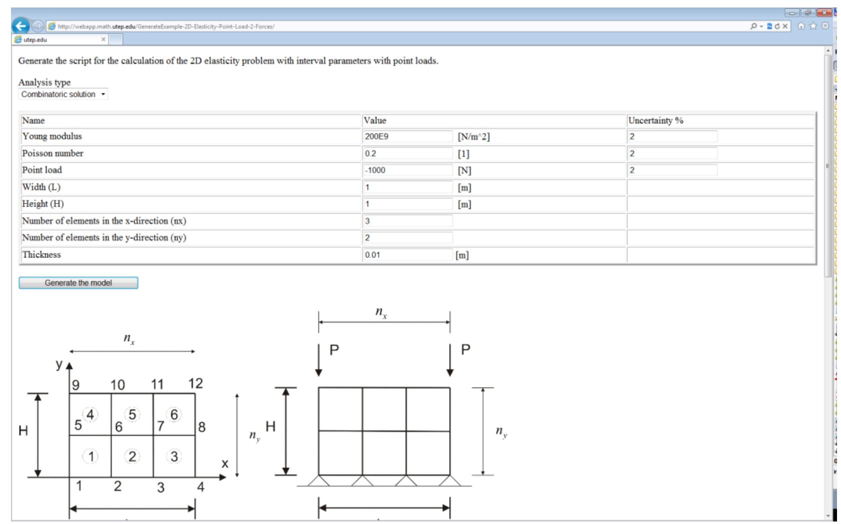
Truss example (similar to benchmark 2)
Truss example (similar to benchmark 2)
Truss example (page 185 from prof. Muhanna paper's)
Truss example geometric uncertainty (page 185 from prof. Muhanna paper's)

Frame examples

Frame (2 elements)

2D examples

2D-Elasticity point load
2D-Elasticity point load (2 loads only)
2D-Elasticity point load (2 loads only), dependent case
2D-Elasticity point load (2 point support)
2D-Elasticity point load (2 point support) - uncertain geometry
2D-Elasticity gravity force
2D-Elasticity surface load


http://webapp.math.utep.edu/GenerateExample-2D-Elasticity-Point-Load-2-Forces/

utep.edu

Generate the script for the calculation of the 2D elasticity problem with interval parameters with point loads.

Analysis type
Combinatoric solution

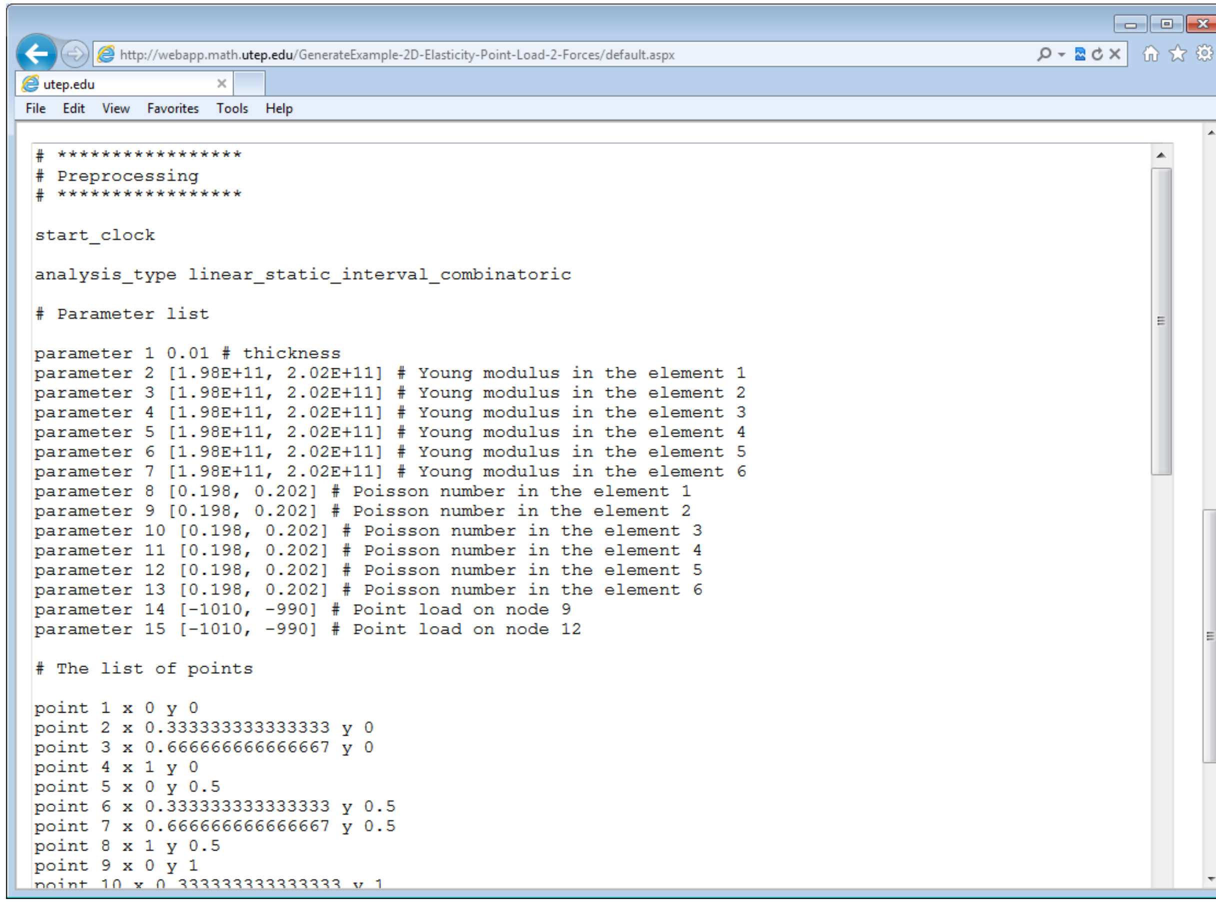
Name	Value		Uncertainty %
Young modulus	200E9	[N/m^2]	2
Poisson number	0.2	[1]	2
Point load	-1000	[N]	2
Width (L)	1	[m]	
Height (H)	1	[m]	
Number of elements in the x-direction (nx)	3		
Number of elements in the y-direction (ny)	2		
Thickness	0.01	[m]	

Generate the model

http://webapp.math.utep.edu/Pages/IntervalFEMExamples.htm

Automatically generated test problems

DSL
(Domain Specific Languages)

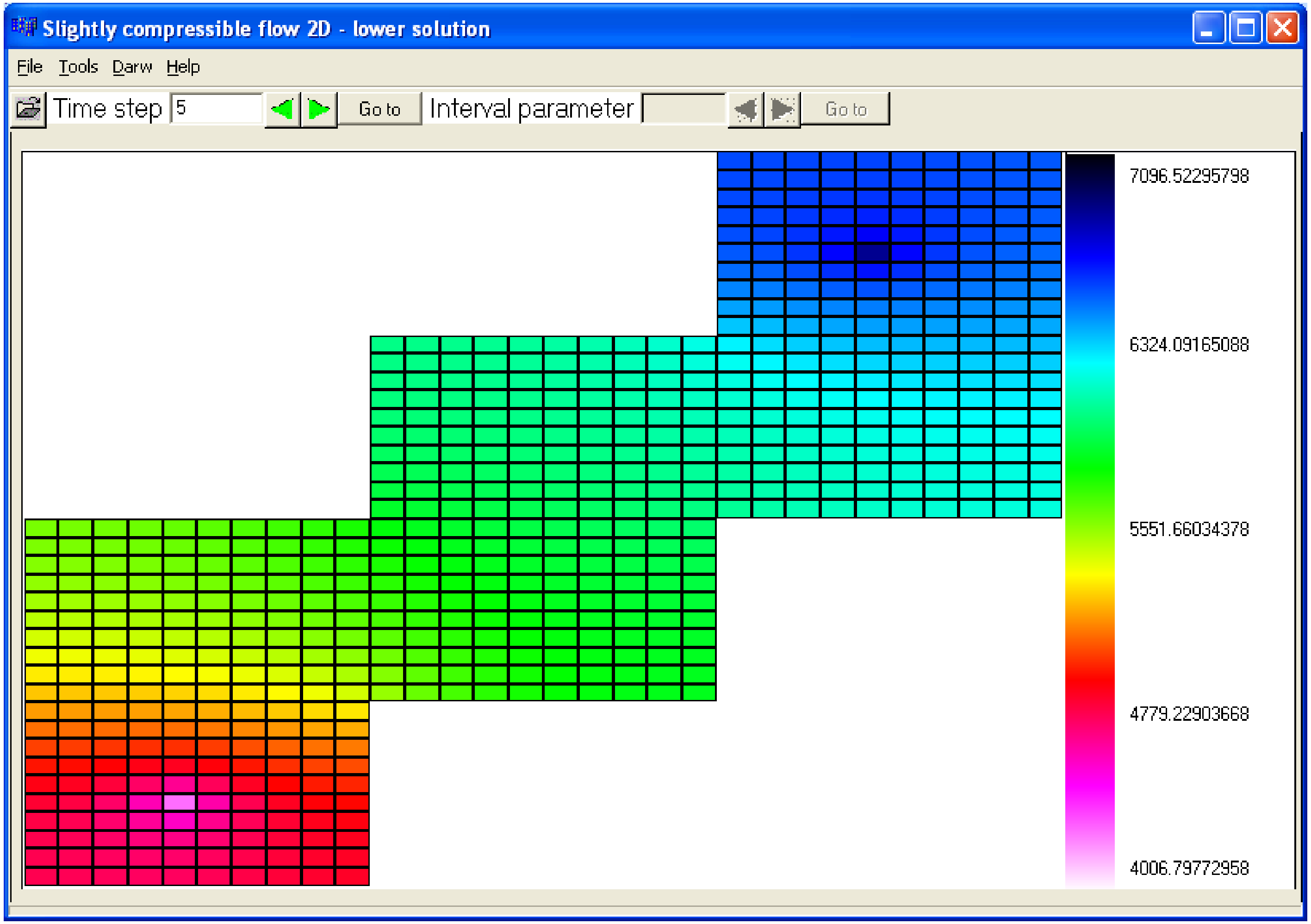



```
# *****************
# Preprocessing
# *****************

start_clock

analysis_type linear_static_interval_combinatoric

# Parameter list

parameter 1 0.01 # thickness
parameter 2 [1.98E+11, 2.02E+11] # Young modulus in the element 1
parameter 3 [1.98E+11, 2.02E+11] # Young modulus in the element 2
parameter 4 [1.98E+11, 2.02E+11] # Young modulus in the element 3
parameter 5 [1.98E+11, 2.02E+11] # Young modulus in the element 4
parameter 6 [1.98E+11, 2.02E+11] # Young modulus in the element 5
parameter 7 [1.98E+11, 2.02E+11] # Young modulus in the element 6
parameter 8 [0.198, 0.202] # Poisson number in the element 1
parameter 9 [0.198, 0.202] # Poisson number in the element 2
parameter 10 [0.198, 0.202] # Poisson number in the element 3
parameter 11 [0.198, 0.202] # Poisson number in the element 4
parameter 12 [0.198, 0.202] # Poisson number in the element 5
parameter 13 [0.198, 0.202] # Poisson number in the element 6
parameter 14 [-1010, -990] # Point load on node 9
parameter 15 [-1010, -990] # Point load on node 12

# The list of points

point 1 x 0 y 0
point 2 x 0.333333333333333 y 0
point 3 x 0.666666666666667 y 0
point 4 x 1 y 0
point 5 x 0 y 0.5
point 6 x 0.333333333333333 y 0.5
point 7 x 0.666666666666667 y 0.5
point 8 x 1 y 0.5
point 9 x 0 y 1
point 10 x 0.333333333333333 y 1
```

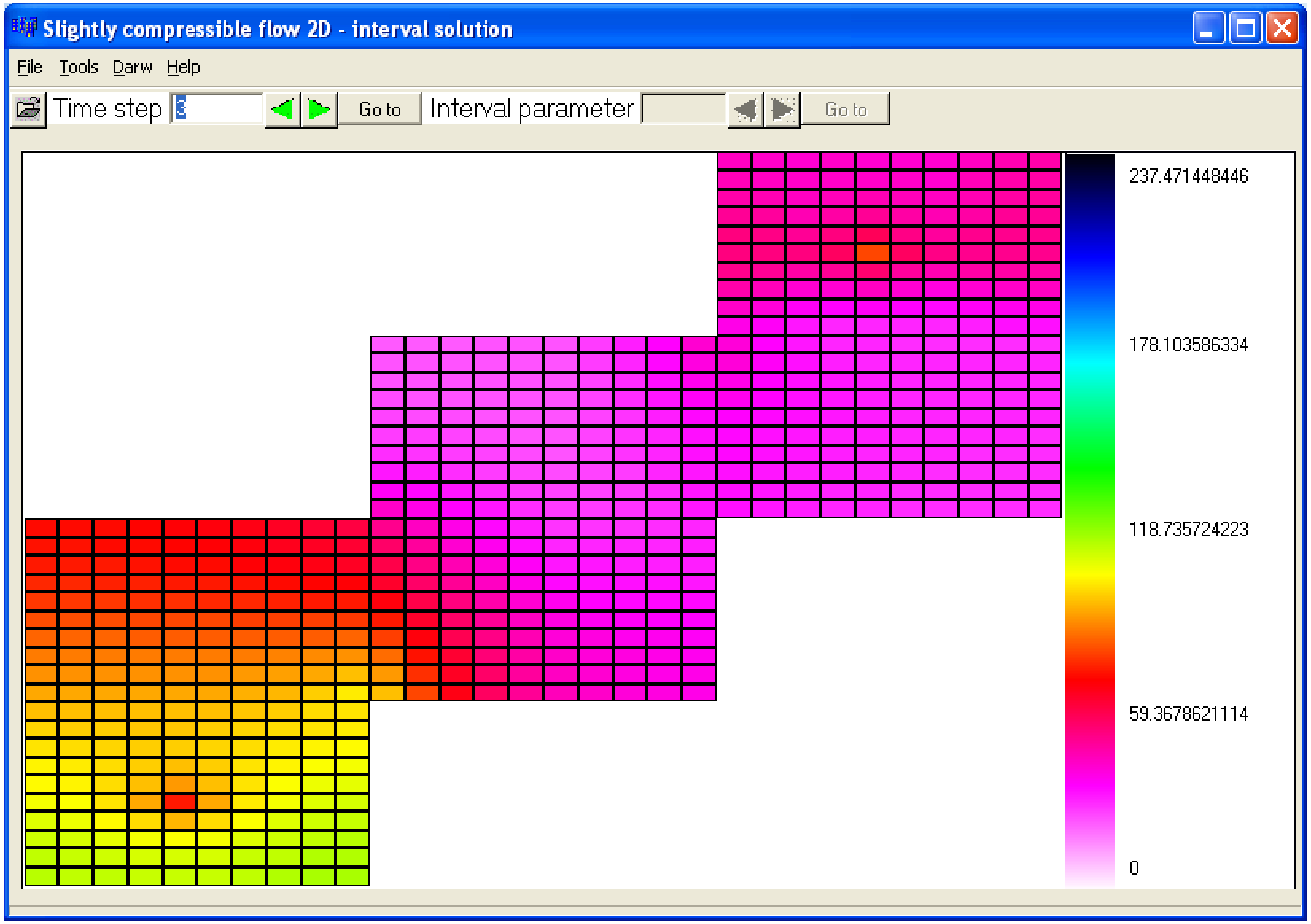
Slightly compressible flow equations

$$\frac{\partial}{\partial x}\left(\frac{\beta_c A_x k_x}{\mu B}\frac{\partial p}{\partial x}\right)\Delta x+\frac{\partial}{\partial y}\left(\frac{\beta_c A_y k_y}{\mu B}\frac{\partial p}{\partial y}\right)\Delta y+q_{sc}=\frac{V_b \phi c}{\alpha_c B^o}\frac{\partial p}{\partial t}$$

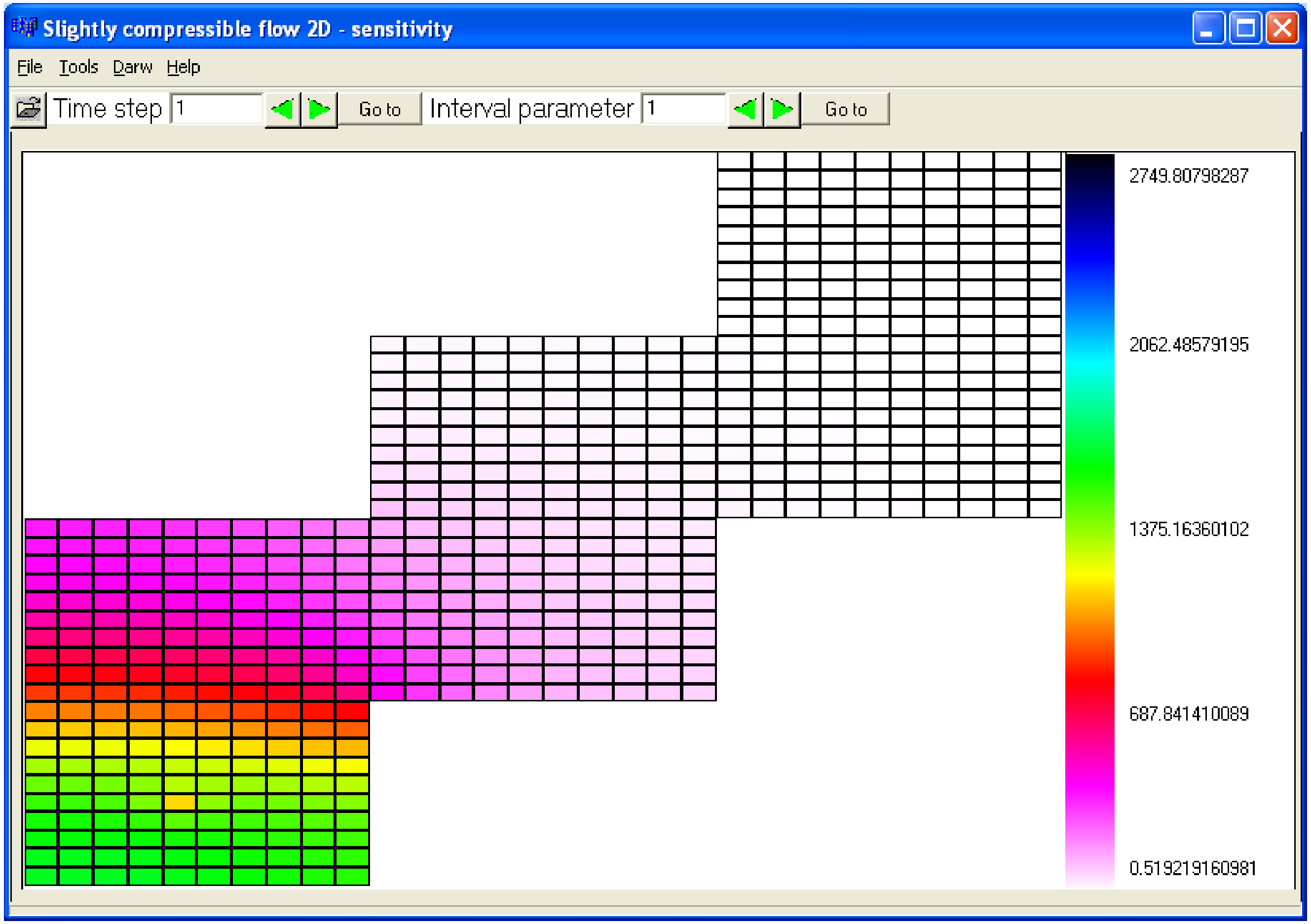
Slightly compressible flow



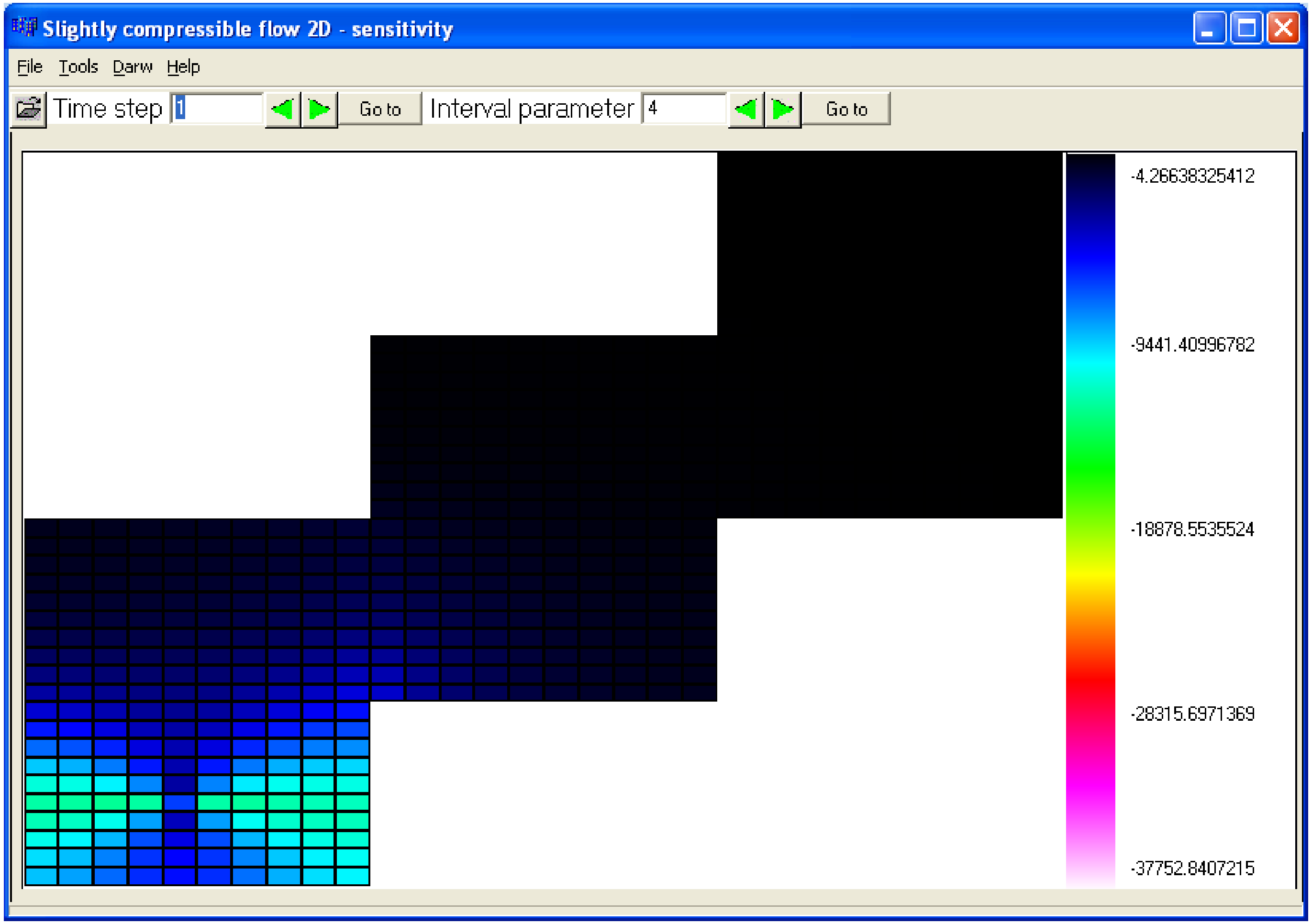
Slightly compressible flow



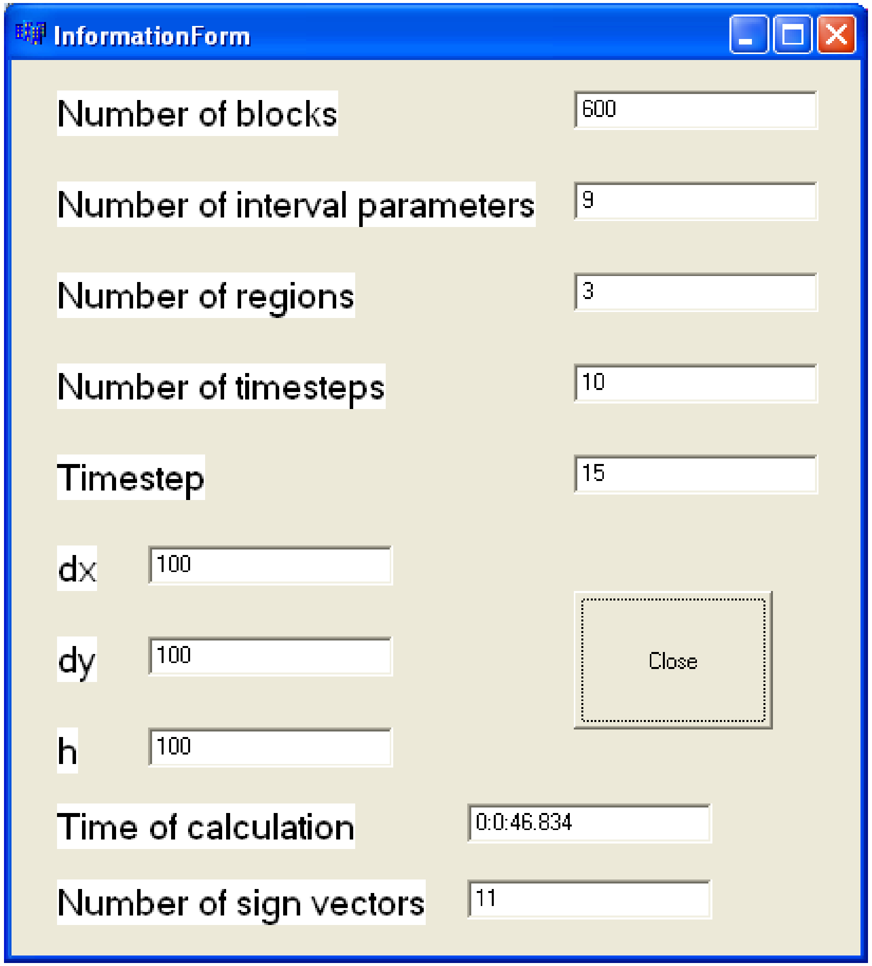
Slightly compressible flow



Slightly compressible flow

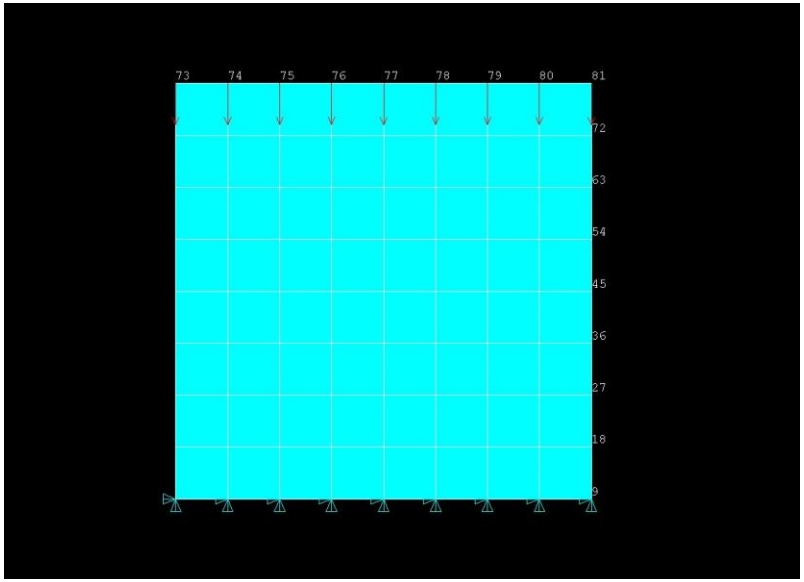


Input Parameters

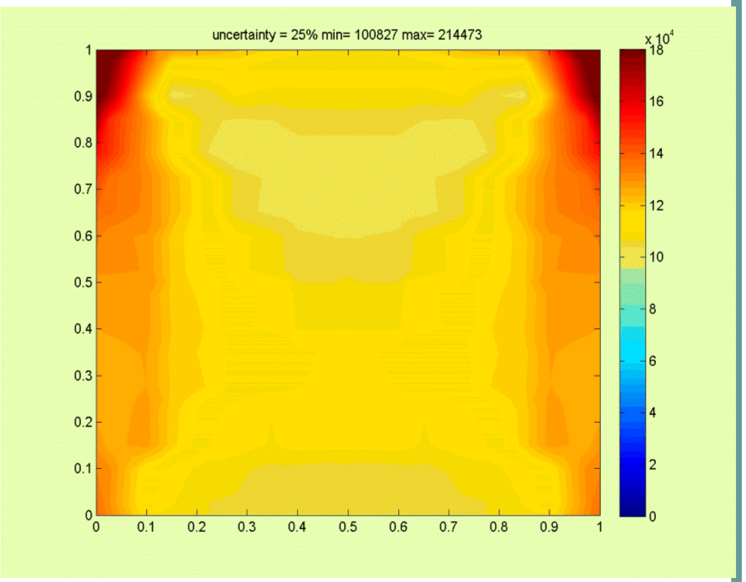


2D elasticity problem with the interval parameters

Model



Solution

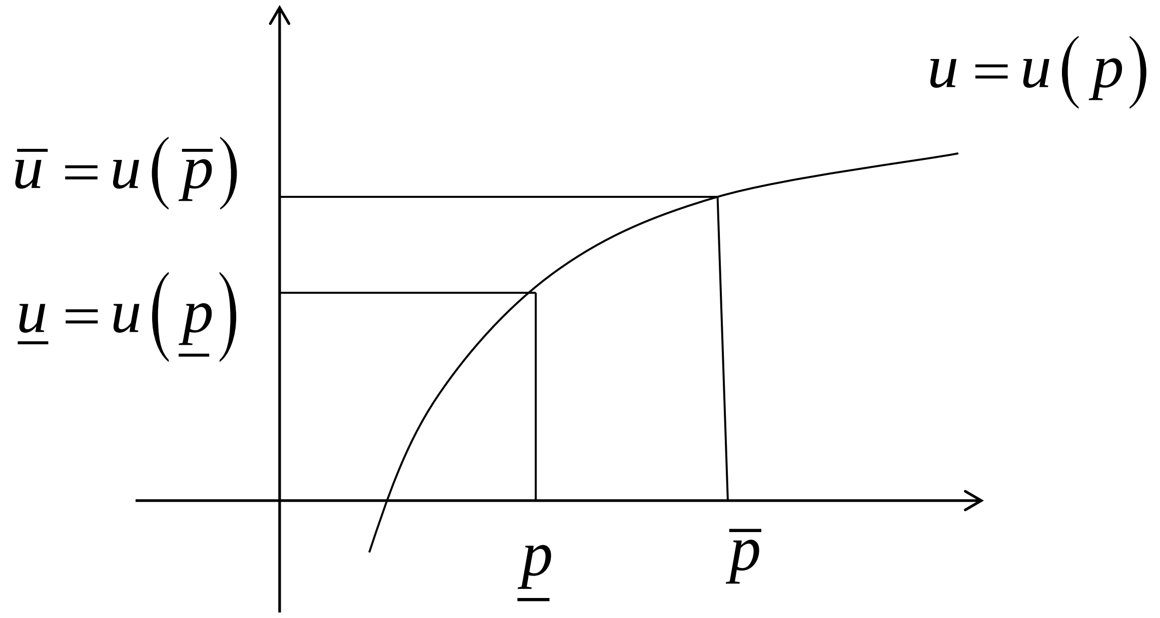


Mathematical model

$$\mu u_{i,jj} + (\mu+\lambda) u_{j,ij} + F_i = \rho \partial_{tt} u_i \quad \text{or} \quad \mu \nabla^2 \mathbf{u} + (\mu+\lambda)\nabla(\nabla\cdot\mathbf{u}) + \mathbf{F} = \rho \frac{\partial^2 \mathbf{u}}{\partial t^2}.$$

Montonicity of the solution

$$\underline{u} = u\left(\underline{p}_1, \overline{p}_2, \ldots, \underline{p}_m\right), \overline{u} = u\left(\overline{p}_1, \underline{p}_2, \ldots, \overline{p}_m\right)$$



Monotonicity of the solution

- Monotone solution

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} = \begin{bmatrix} p_1 + p_2 \\ p_2 \end{bmatrix} \qquad u_1 = \frac{p_1}{2} + p_2, \;\; u_2 = \frac{p_1}{2}$$

- Non-monotone solution

$$u^2 - p^4 = 0 \qquad u_1 = p^2, u_2 = -p^2$$

Interval solution for monotone functions (gradient descent method)

If $\frac{\partial u}{\partial p} \geq 0$ then $p^{min} = \underline{p}, p^{max} = \overline{p}$

If $\frac{\partial u}{\partial p} < 0$ then $p^{min} = \overline{p}, p^{max} = \underline{p}$

$$\underline{u} = u(p^{min}), \overline{u} = u(p^{\max})$$

Interval solution
Verification of the results by using search method with 3 intermediate points

Table VIII. Displacements in the case of 4 (2x2) finite elements and 5% uncertainty

	Combinatoric	Combinatoric	Gradient free	Gradient free		
DOF	$\underline{u}$	$\overline{u}$	$\underline{u}$	$\overline{u}$	Error $\underline{u}$ %	Error $\overline{u}$ %
5	-8.770130E-23	1.113670E-22	-8.770130E-23	1.113670E-22	0.000000E+00	0.000000E+00
6	-4.404760E-07	-3.605710E-07	-4.404760E-07	-3.605710E-07	0.000000E+00	0.000000E+00
7	4.822140E-08	5.890760E-08	4.822140E-08	5.890760E-08	0.000000E+00	0.000000E+00
8	-6.087950E-07	-4.983560E-07	-6.087950E-07	-4.983560E-07	0.000000E+00	0.000000E+00
11	-5.890760E-08	-4.822140E-08	-5.890760E-08	-4.822140E-08	0.000000E+00	0.000000E+00
12	-6.087950E-07	-4.983560E-07	-6.087950E-07	-4.983560E-07	0.000000E+00	0.000000E+00
13	-1.851990E-23	2.469330E-22	-1.851990E-23	2.469330E-22	0.000000E+00	0.000000E+00
14	-3.540790E-07	-2.898470E-07	-3.540790E-07	-2.898470E-07	0.000000E+00	0.000000E+00
15	-4.815970E-07	-3.942320E-07	-4.815970E-07	-3.942320E-07	0.000000E+00	0.000000E+00
16	-1.840780E-06	-1.506850E-06	-1.840780E-06	-1.506850E-06	0.000000E+00	0.000000E+00
17	3.942320E-07	4.815970E-07	3.942320E-07	4.815970E-07	0.000000E+00	0.000000E+00
18	-1.840780E-06	-1.506850E-06	-1.840780E-06	-1.506850E-06	0.000000E+00	0.000000E+00

Interval solution

Table X. Displacements in the case of 9 (3x3) finite elements and 5% uncertainty

	Combinatoric		Gradient free			
DOF	$\underline{u}$	$\overline{u}$	$\underline{u}$	$\overline{u}$	Error $\underline{u}$ %	Error $\overline{u}$ %
5	2.991120E-09	4.758950E-08	2.991120E-09	4.758950E-08	0.000000E+00	0.000000E+00
6	-3.401560E-07	-2.737550E-07	-3.401560E-07	-2.738240E-07	0.000000E+00	2.520502E-02
7	-2.700300E-08	3.178720E-08	-2.681040E-08	3.178720E-08	7.132541E-01	0.000000E+00
8	-4.259370E-07	-2.781400E-07	-4.259370E-07	-2.781400E-07	0.000000E+00	0.000000E+00
11	-4.758950E-08	-2.991120E-09	-4.758950E-08	-2.991120E-09	0.000000E+00	0.000000E+00
12	-3.401560E-07	-2.737550E-07	-3.401560E-07	-2.738240E-07	0.000000E+00	2.520502E-02
15	-3.178720E-08	2.700300E-08	-3.178720E-08	2.681040E-08	0.000000E+00	7.132541E-01
16	-4.259370E-07	-2.781400E-07	-4.259370E-07	-2.781400E-07	0.000000E+00	0.000000E+00
17	-8.584880E-09	1.665680E-07	-8.584880E-09	1.665680E-07	0.000000E+00	0.000000E+00
18	-5.498820E-07	-4.324890E-07	-5.498820E-07	-4.324890E-07	0.000000E+00	0.000000E+00

Interval solution

Table XI. Combinations of parameters which correspond to the interval displacements 9 elements (3x3) and 5% uncertainty

	Combinatoric	Combinatoric	Gradient free	Gradient free		
DOF	$\underline{u}$	$\overline{u}$	$\underline{u}$	$\overline{u}$	$\underline{u}$	$\overline{u}$
5	0,1,1,0,1,0,1,1,1,0,1	1,0,0,1,0,1,0,0,0,1,0	0,1,1,0,1,0,1,1,1,0,1	1,0,0,1,0,1,0,0,0,1,0	0	0
6	0,0,0,0,1,0,1,1,1,0,1	1,1,1,$\boxed{0}$,0,1,0,0,0,1,0	0,0,0,0,1,0,1,1,1,0,1	1,1,1,$\boxed{1}$,0,1,0,0,0,1,0	0	1
7	0,0,1,0,1,0,1,1,1,0,1	1,$\boxed{0}$,0,1,0,1,0,0,0,1,0	0,$\boxed{1}$,1,0,1,0,1,1,1,0,1	1,0,0,1,0,1,0,0,0,1,0	1	0
8	0,0,1,1,0,1,0,0,0,0,1	1,1,0,0,1,0,1,1,1,1,0	0,0,1,1,0,1,0,0,0,0,1	1,1,0,0,1,0,1,1,1,1,0	0	0
11	0,0,1,1,0,1,0,0,0,0,1	1,1,0,0,1,0,1,1,1,1,0	0,0,1,1,0,1,0,0,0,0,1	1,1,0,0,1,0,1,1,1,1,0	0	0
12	0,0,0,0,1,0,1,1,1,1,0	1,1,1,1,0,$\boxed{0}$,0,0,0,0,1	0,0,0,0,1,0,1,1,1,1,0	1,1,1,1,0,$\boxed{1}$,0,0,0,0,1	0	1
15	0,0,1,1,0,1,0,0,0,0,1	1,$\boxed{0}$,0,0,1,0,1,1,1,1,0	0,0,1,1,0,1,0,0,0,0,1	1,$\boxed{1}$,0,0,1,0,1,1,1,1,0	0	1
16	1,0,0,1,0,1,0,0,0,1,0	0,1,1,0,1,0,1,1,1,0,1	1,0,0,1,0,1,0,0,0,1,0	0,1,1,0,1,0,1,1,1,0,1	0	0
17	0,1,1,1,1,1,0,1,0,0,1	1,0,0,0,0,0,1,0,1,1,0	0,1,1,1,1,1,0,1,0,0,1	1,0,0,0,0,0,1,0,1,1,0	0	0
18	0,0,0,0,0,0,1,1,1,0,1	1,1,1,1,1,1,0,0,0,1,0	0,0,0,0,0,0,1,1,1,0,1	1,1,1,1,1,1,0,0,0,1,0	0	0

Interval solution

Table XII. Interval displacements for 10% uncertainty and 9 finite elements

DOF	$\underline{u}$	$\overline{u}$	$\underline{u}$	$\overline{u}$	Error $\underline{u}$ %	Error $\overline{u}$ %
5	-1.934900E-08	7.022680E-08	-1.934900E-08	7.022470E-08	0.000000E+00	2.990311E-03
6	-3.796920E-07	-2.451030E-07	-3.796920E-07	-2.458330E-07	0.000000E+00	2.978340E-01
7	-5.781790E-08	6.089000E-08	-5.677190E-08	6.089000E-08	1.809128E+00	0.000000E+00
8	-5.089660E-07	-2.112750E-07	-5.089660E-07	-2.112750E-07	0.000000E+00	0.000000E+00
11	-7.022680E-08	1.934900E-08	-7.022470E-08	1.934900E-08	2.990311E-03	0.000000E+00
12	-3.796920E-07	-2.451030E-07	-3.796920E-07	-2.458330E-07	0.000000E+00	2.978340E-01
15	-6.089000E-08	5.781790E-08	-6.089000E-08	5.677190E-08	0.000000E+00	1.809128E+00
16	-5.089660E-07	-2.112750E-07	-5.089660E-07	-2.112750E-07	0.000000E+00	0.000000E+00
17	-9.469690E-08	2.590120E-07	-9.315200E-08	2.590120E-07	1.631416E+00	0.000000E+00
18	-6.187590E-07	-3.820950E-07	-6.187590E-07	-3.820950E-07	0.000000E+00	0.000000E+00

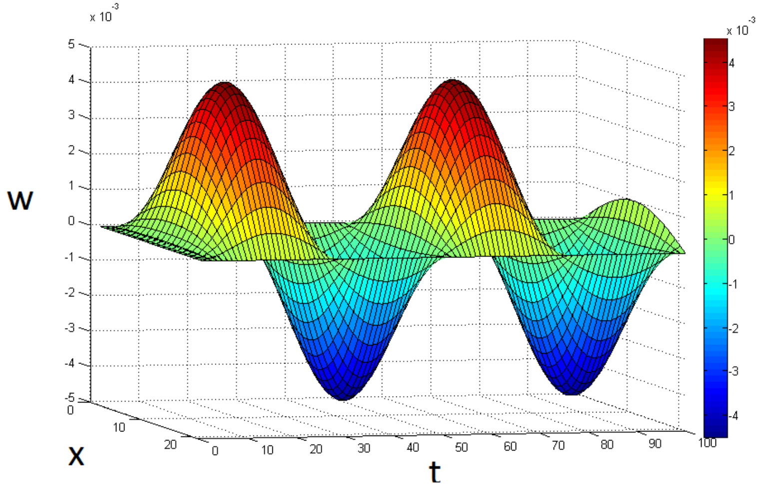
Interval solution

Table XV. Interval displacements 10% uncertainty

$\underline{u}$	$\overline{u}$	$\underline{u}$	$\overline{u}$		
0,1,1,0,1,0,1,1,1,0,1	1,0,0,1,0,1,$\boxed{1}$,0,0,1,0	0,1,1,0,1,0,1,1,1,0,1	1,0,0,1,0,1,$\boxed{0}$,0,0,1,0	0	1
0,0,0,0,1,0,1,1,1,0,1	1,1,1,$\boxed{0}$,0,1,0,0,0,1,0	0,0,0,0,1,0,1,1,1,0,1	1,1,1,$\boxed{1}$,0,1,0,0,0,1,0	0	1
0,0,1,0,1,0,1,1,1,0,1	1,$\boxed{0}$,0,1,0,1,0,0,0,1,0	0,$\boxed{1}$,1,0,1,0,1,1,1,0,1	1,0,0,1,0,1,0,0,0,1,0	1	0
0,0,1,1,0,1,0,0,0,0,1	1,1,0,1,1,0,1,1,1,1,0	0,0,1,1,0,1,0,0,0,0,1	1,1,0,1,1,0,1,1,1,1,0	0	0
0,0,1,1,0,1,0,0,1,0,1	1,1,0,0,1,0,1,1,$\boxed{1}$,1,0	0,0,1,1,0,1,0,0,$\boxed{0}$,0,1	1,1,0,0,1,0,1,1,1,1,0	1	0
0,0,0,0,1,0,1,1,1,1,0	1,1,1,1,0,$\boxed{0}$,0,0,0,0,1	0,0,0,0,1,0,1,1,1,1,0	1,1,1,1,0,$\boxed{1}$,0,0,0,0,1	0	1
0,0,1,1,0,1,0,0,0,0,1	1,$\boxed{0}$,0,0,1,0,1,1,1,1,0	0,0,1,1,0,1,0,0,0,0,1	1,$\boxed{1}$,0,0,1,0,1,1,1,1,0	0	1
1,0,0,1,0,1,0,0,0,1,0	0,1,1,0,1,1,1,1,1,0,1	1,0,0,1,0,1,0,0,0,1,0	0,1,1,0,1,1,1,1,1,0,1	0	0
0,0,1,0,1,1,0,1,0,0,1	1,$\boxed{0}$,0,$\boxed{0}$,0,0,1,0,1,1,0	0,$\boxed{1}$,1,$\boxed{1}$,1,1,0,1,0,0,1	1,0,0,0,0,0,1,0,1,1,0	2	0
0,0,0,0,0,0,1,1,1,0,1	1,1,1,1,1,1,0,0,0,1,0	0,0,0,0,0,0,1,1,1,0,1	1,1,1,1,1,1,0,0,0,1,0	0	0
0,1,1,1,1,1,0,0,0,0,1	1,0,0,0,0,0,1,1,1,1,0	0,1,1,1,1,1,0,0,0,0,1	1,0,0,0,0,0,1,1,1,1,0	0	0
0,0,1,0,0,1,0,0,0,0,1	1,1,0,1,1,0,1,1,1,1,0	0,0,1,0,0,1,0,0,0,0,1	1,1,0,1,1,0,1,1,1,1,0	0	0
0,0,1,0,0,0,1,0,1,0,1	1,$\boxed{0}$,0,1,1,$\boxed{0}$,0,1,0,1,0	0,0,1,0,0,0,1,0,1,0,1	1,$\boxed{1}$,0,1,1,$\boxed{1}$,0,1,0,1,0	0	2

Interval vibrations and monotonicity

$$EJ\frac{\partial^4 w}{\partial x^4} = q - \rho A\frac{\partial^2 w}{\partial t^2}$$



Uncertain solution

- Set-valued parameters

$$\underline{u}(x,t) = u(x,t,p^{min})$$

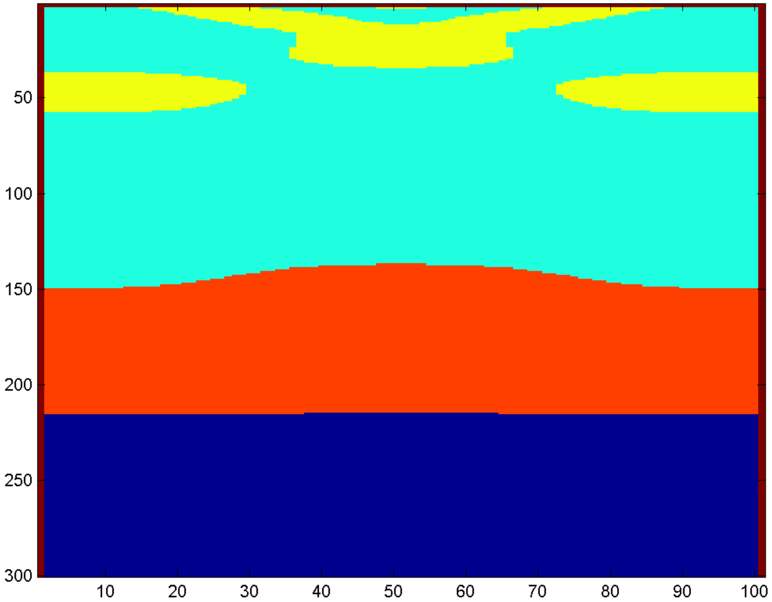
$$\overline{u}(x,t) = u(x,t,p^{\max})$$

$$p^{\min} = \begin{cases} \arg\min\limits_{p} u(x,t,p) \\ s.t. p \in \mathbf{p} \end{cases}, p^{max} = \begin{cases} \arg\max\limits_{p} u(x,t,p) \\ s.t. p \in \mathbf{p} \end{cases}$$

Vibrations of beam

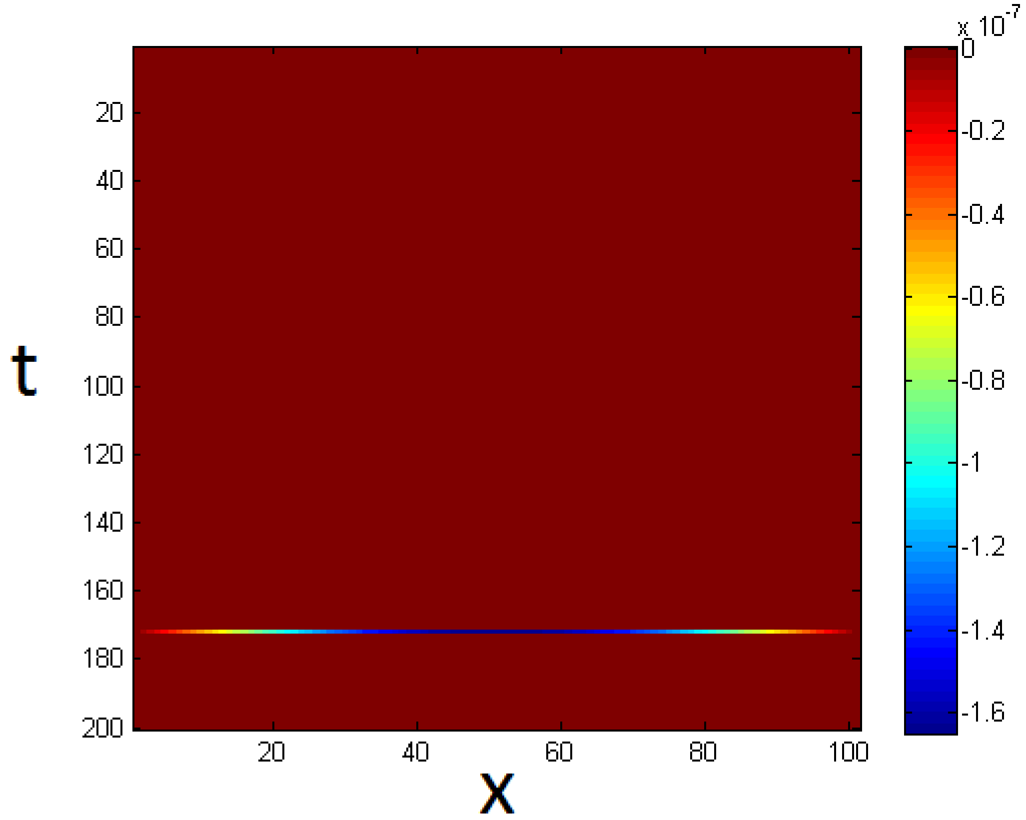
$$\underline{u} = u\left(p^{min}\right)$$

p^{min}

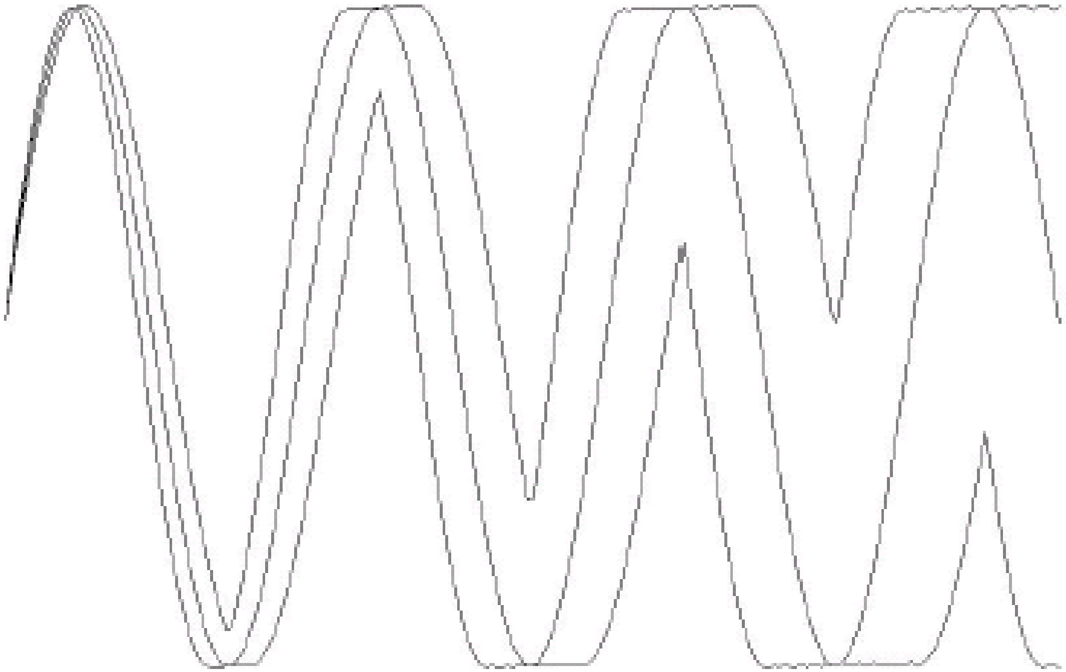


Vibrations of beam

$$\underline{u}^{(2)} - \underline{u}^{(7)}$$



Hermitte approximation



$$\underline{y}^{approx} = min\{y^{approx}(t, p^{(0)}, ..., p^{(n)}, p) : p \in \mathbf{p}\},$$

$$\overline{y}^{approx} = max\{y^{approx}(t, p^{(0)}, ..., p^{(n)}, p) : p \in \mathbf{p}\}.$$

First order approximation

$$T_1(t, p^{(k)}, p) = y(t, p^{(k)}) + \sum_i \frac{\partial y(t, p^{(k)})}{\partial p_i}(p_i - p_i^{(k)})$$

$$L_{T_1} = \{T_1(t, p^{(0)}, p), \ldots, T_1(t, p^{(n)}, p)\}$$

Data for the calculations

$$L_p = \{p^{(0)}, ..., p^{(n)}\}$$

$$L_y = \{y(t, p^{(0)}), y(t, p^{(1)}), ..., y(t, p^{(n)})\}$$

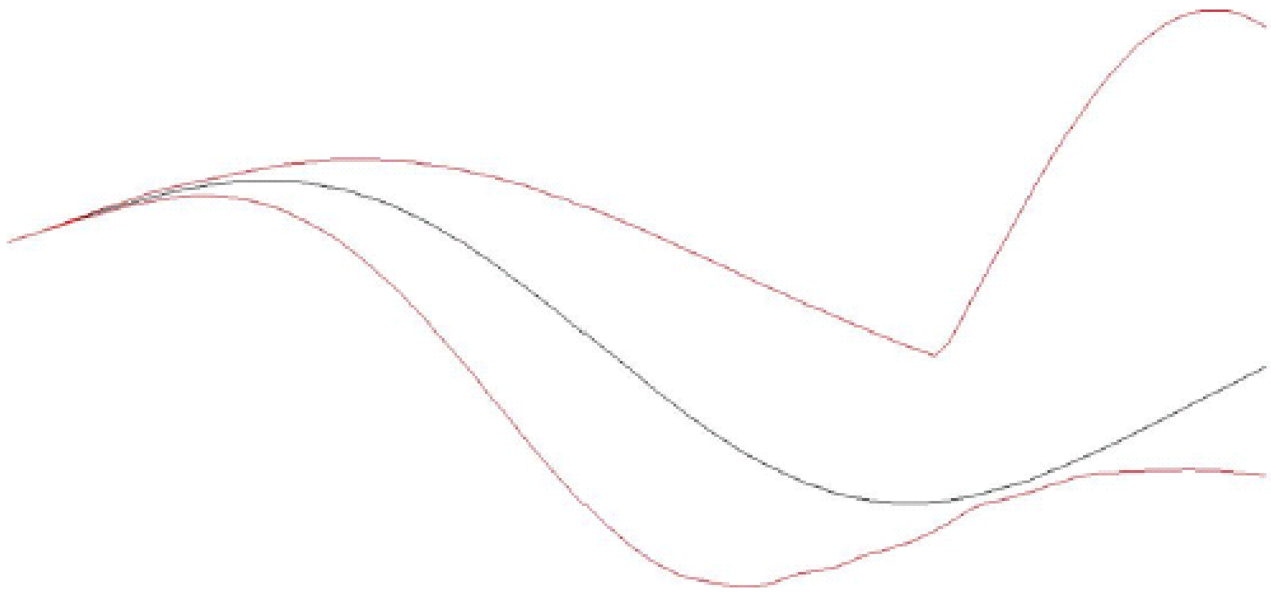
$$L_{\frac{\partial y}{\partial p_i}} = \left\{ \frac{\partial y(t, p^{(0)})}{\partial p_i}, \frac{\partial y(t, p^{(1)})}{\partial p_i}, ..., \frac{\partial y(t, p^{(n)})}{\partial p_i} \right\}$$

Adaptivity

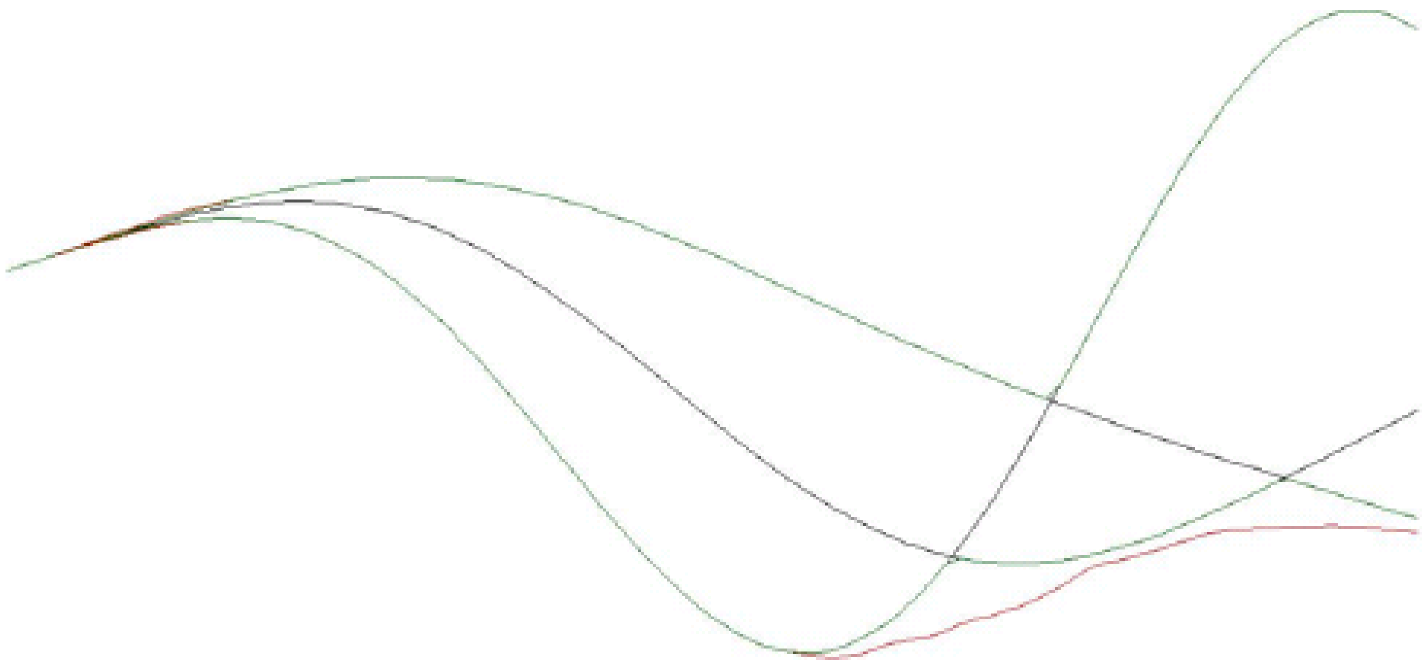
$$k_{min} = \underset{k \in 0,\ldots,n}{argmin} \|p - p^{(k)}\|$$

$$y^{approx}(p) = y^{approx}(t, p^{(k_{min})}, p)$$

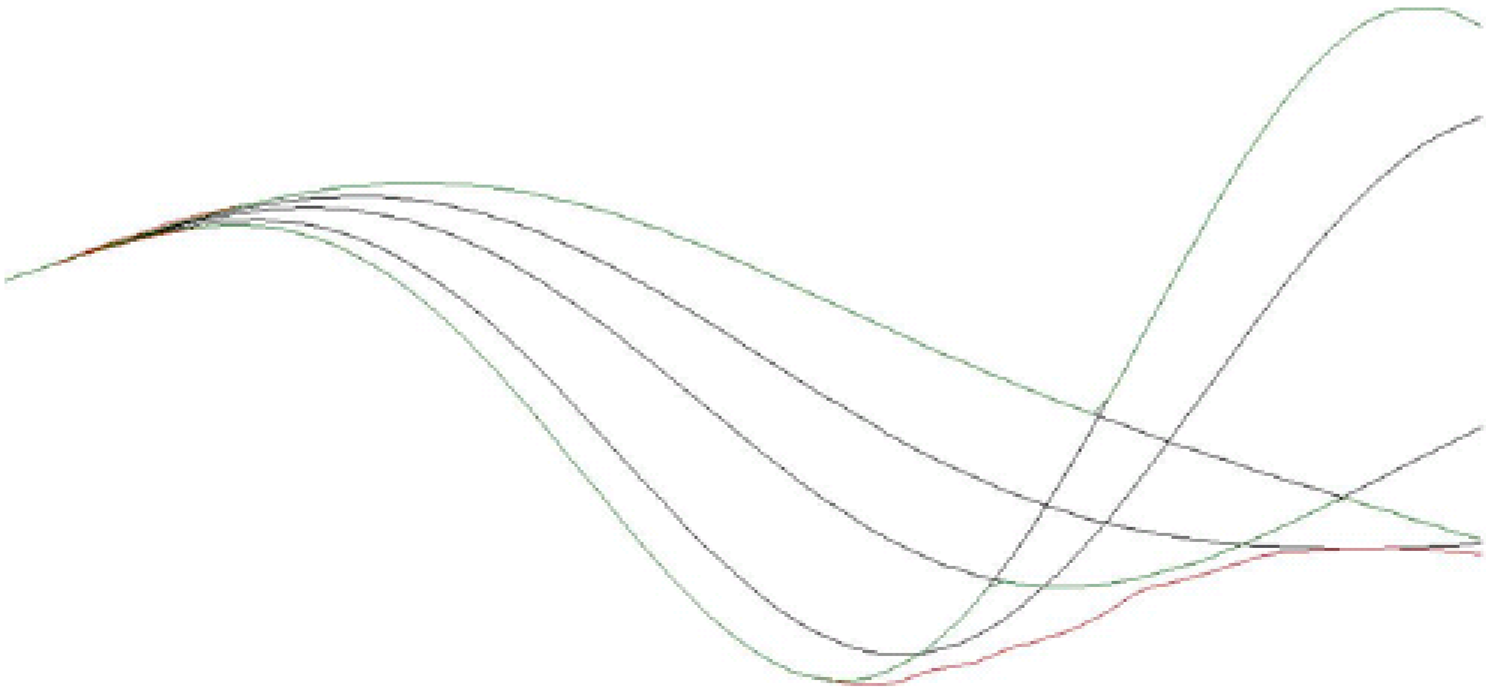
Example (step 1 – 1 solution)



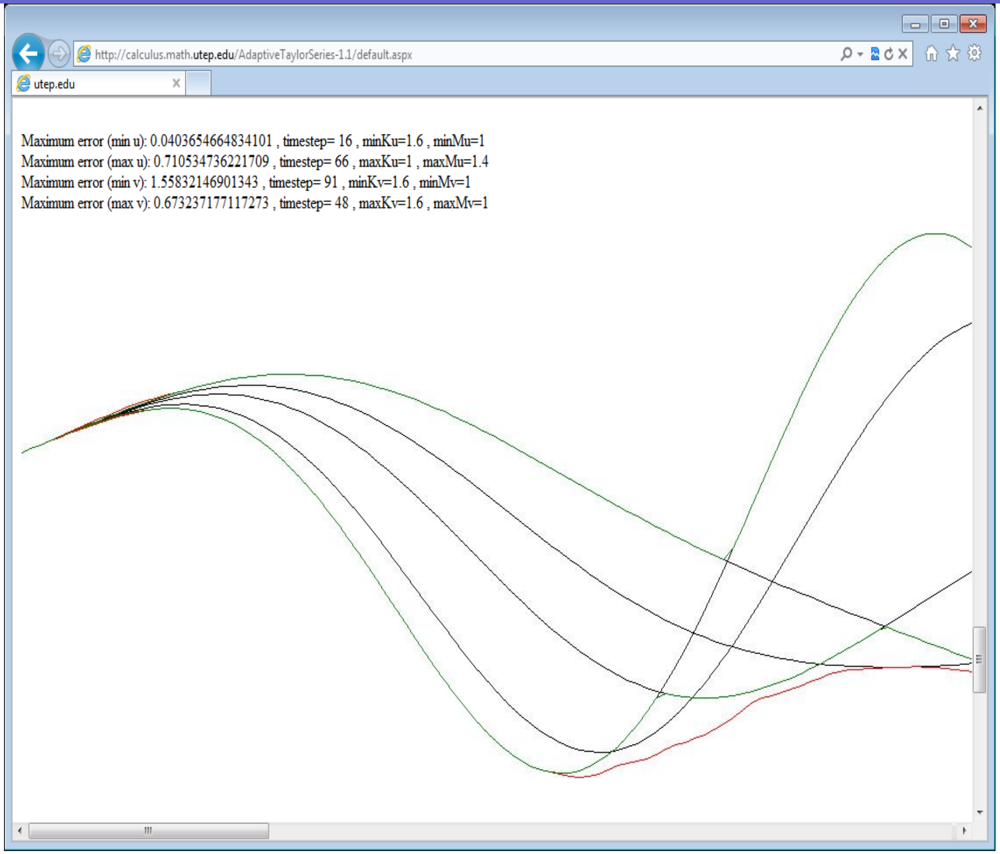
Example (step 2 – 3 solutions)



Example (step 3 – 5 solutions)

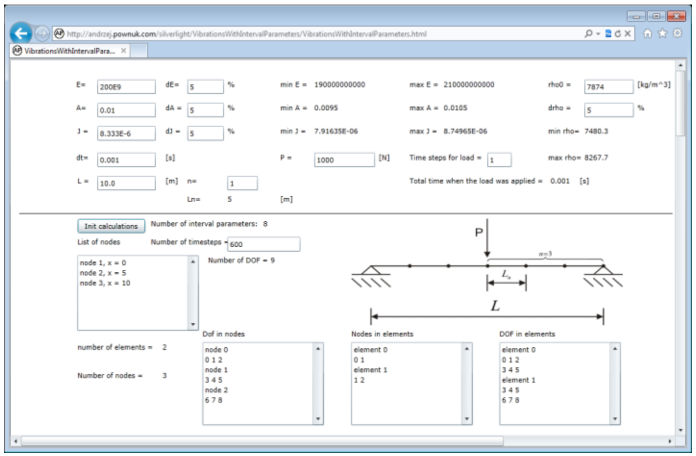


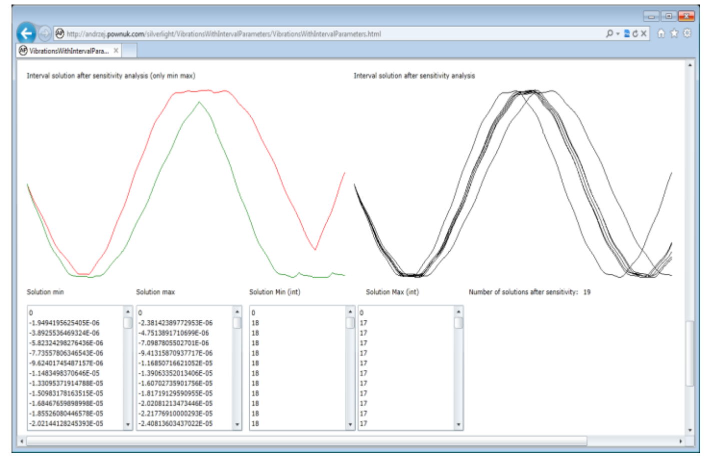
Adaptive Taylor series



http://webapp.math.utep.edu/AdaptiveTaylorSeries-1.1/

Adaptive Taylor series





http://andrzej.pownuk.com/silverlight/VibrationsWithIntervalParameters/VibrationsWithIntervalParameters.html

Epistemic uncertainty

H – set of horses

This is a horse.

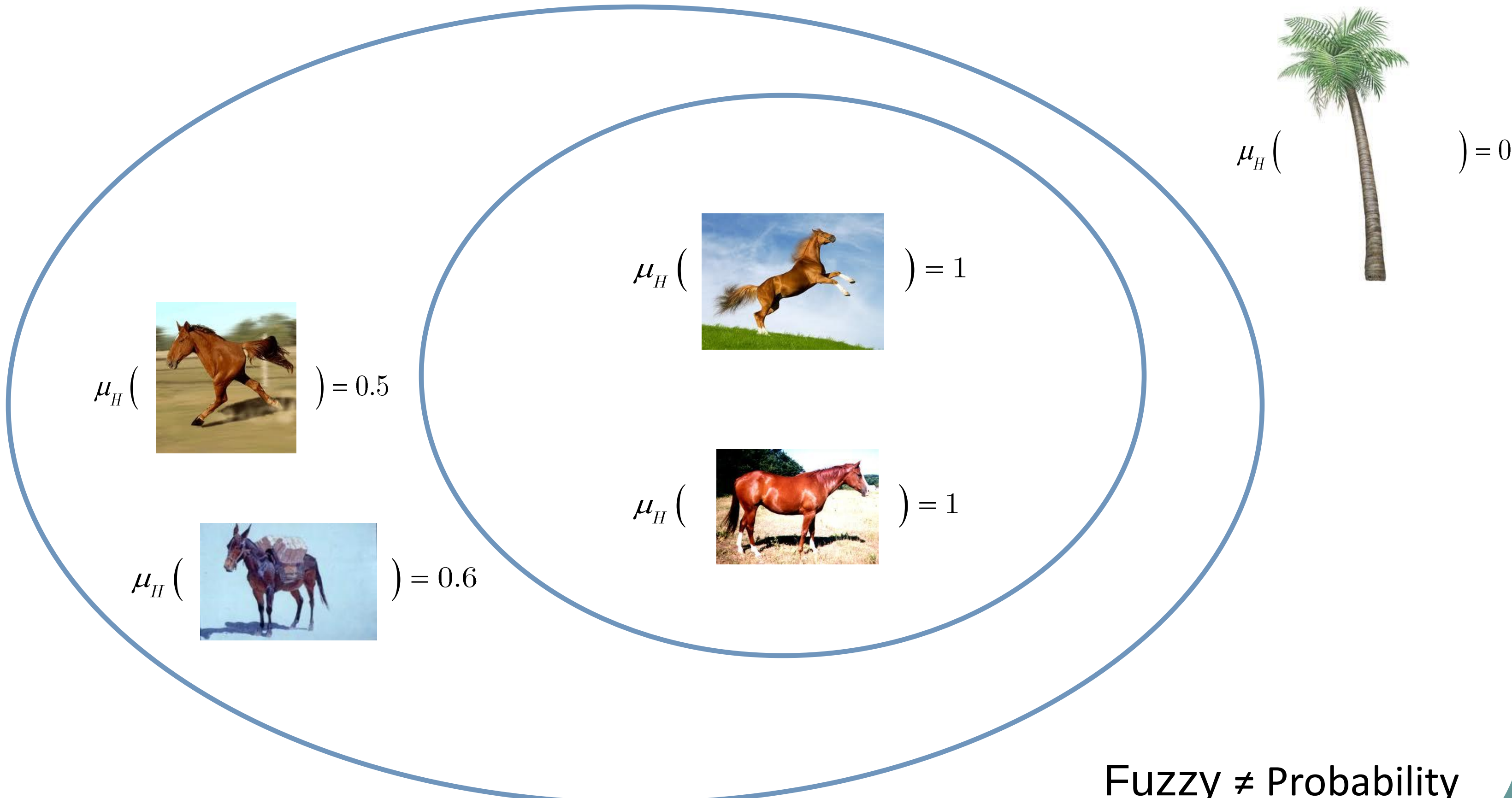
$\in H$

Is this a horse?

$\overset{?}{\in} H$

Fuzzy sets

H – set of horses



Fuzzy ≠ Probability

Fuzzy concept of safety

$$\frac{P_{\max}}{P_{design}} = \gamma$$

$$g(x) = \gamma$$

$$P_f = P\left\{g(x) < 0\right\} < P_f^0$$

Problems with binary logic

- Is it possible to find in the real world statements which are absolutely true?

(L. Wittgenstein, Tractatus Logico-Philosophicus, Annalen der Naturphilosophie, 14, 1921)

?

Modus ponens can be applied if $P \Rightarrow Q$ and Q are true.

uncertainty.wmv

$$\frac{P \Rightarrow Q, P}{Q}$$

When modus ponens can be applied?

Tools

- Approach without tools

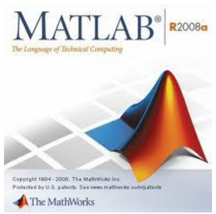
5 years of training

Final result

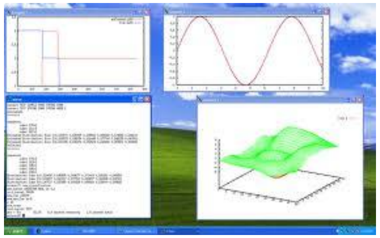
- Approach with tools

Mathematical tools

Mathematica



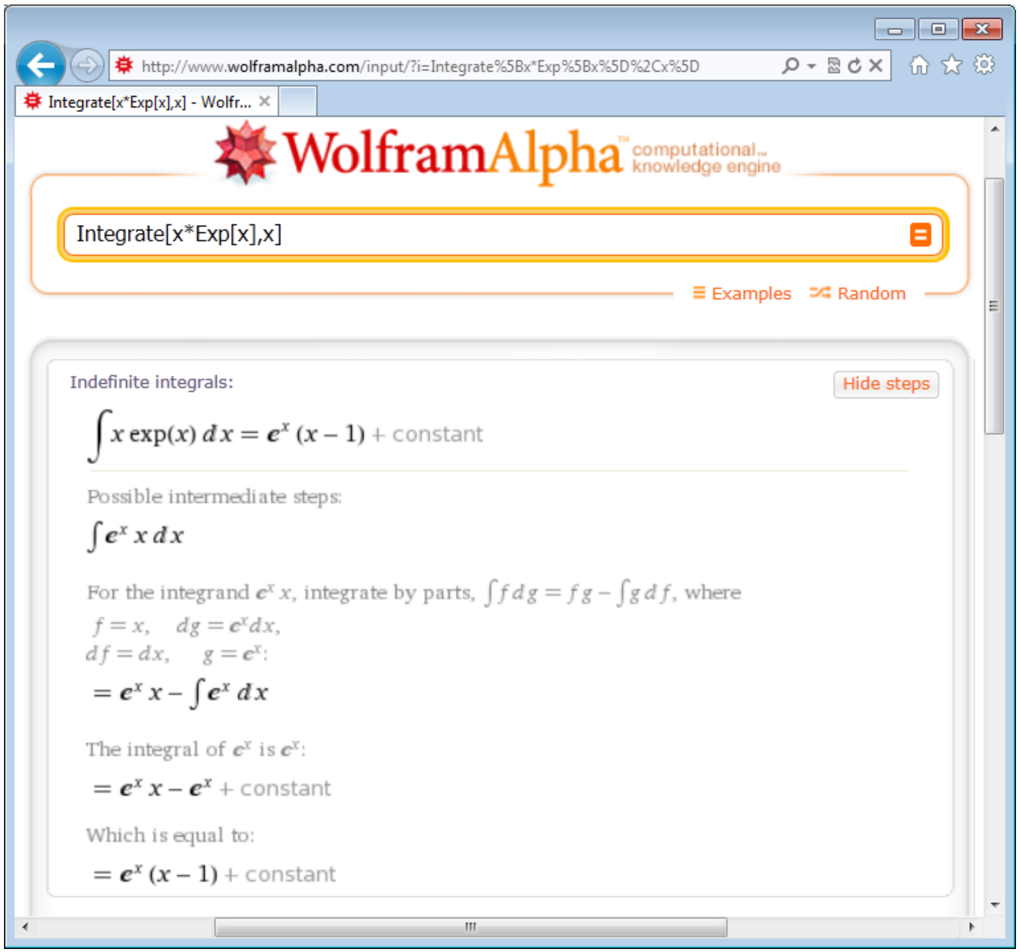
Matlab



Octave

Etc.

Example: http://www.wolframalpha.com

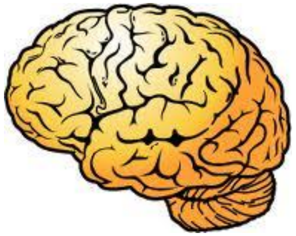


It is possible to calculate not only the result but also intermediate steps in the calculations

Mathematics and programming

Mathematics

mathematical method



results

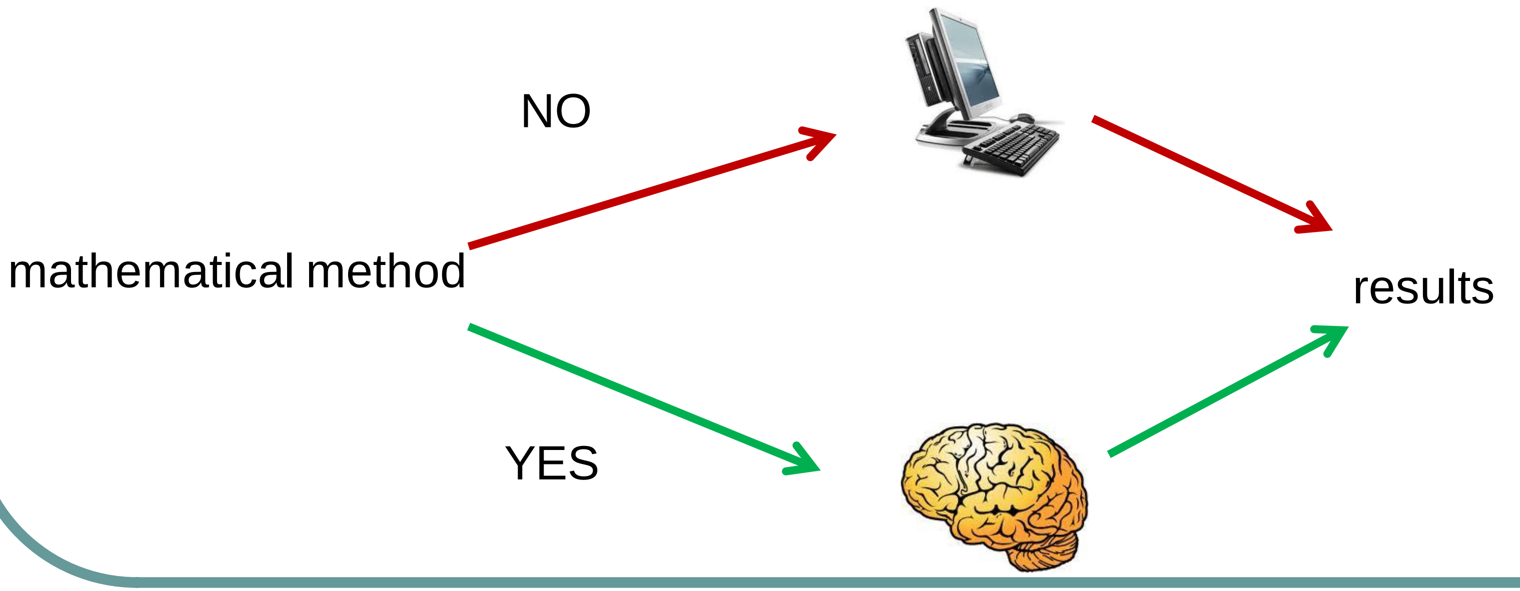
Programming

program

results

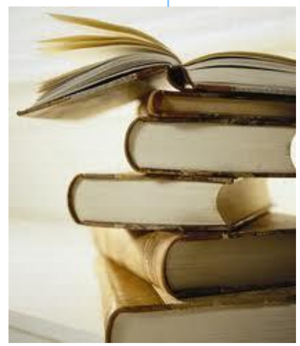
Main problem

- At this moment it is not possible perform general mathematical research automatically without human input.



Science

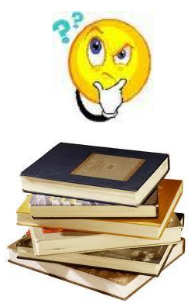
Experiment



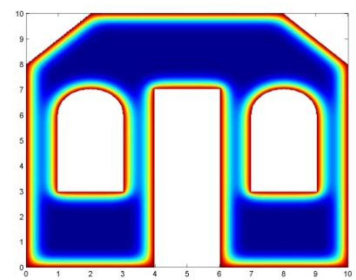
Theory

$$\rho\left(\frac{\partial u}{\partial t}+u\frac{\partial u}{\partial x}+v\frac{\partial u}{\partial y}+w\frac{\partial u}{\partial z}\right)=$$
$$\rho g_x-\frac{\partial p}{\partial x}+\frac{\partial}{\partial x}\left[2\mu\frac{\partial u}{\partial x}+\lambda\nabla\cdot\mathbf{V}\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right]$$
$$\rho\left(\frac{\partial v}{\partial t}+u\frac{\partial v}{\partial x}+v\frac{\partial v}{\partial y}+w\frac{\partial v}{\partial z}\right)=$$
$$\rho g_y-\frac{\partial p}{\partial y}+\frac{\partial}{\partial y}\left[2\mu\frac{\partial v}{\partial y}+\lambda\nabla\cdot\mathbf{V}\right]+\frac{\partial}{\partial z}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial u}{\partial y}+\frac{\partial v}{\partial x}\right)\right]$$
$$\rho\left(\frac{\partial w}{\partial t}+u\frac{\partial w}{\partial x}+v\frac{\partial w}{\partial y}+w\frac{\partial w}{\partial z}\right)=$$
$$\rho g_z-\frac{\partial p}{\partial z}+\frac{\partial}{\partial z}\left[2\mu\frac{\partial w}{\partial z}+\lambda\nabla\cdot\mathbf{V}\right]+\frac{\partial}{\partial x}\left[\mu\left(\frac{\partial w}{\partial x}+\frac{\partial u}{\partial z}\right)\right]+\frac{\partial}{\partial y}\left[\mu\left(\frac{\partial v}{\partial z}+\frac{\partial w}{\partial y}\right)\right]$$

Mathematical model



Scientific hypothesis



Simulations (predictions)

HPC computing

New tool

- Self adaptive computational methods

Symbolic calculations

- Interval arithmetic example

- 2*[1,2]*x-x=1
- (2*[1,2]-1)x=1
- ([2,4]-1)x=1
- [1,3]x=1
- x=1/[1,3]

It is possible to solve more complicated equations and get justification of each step of the calculations.

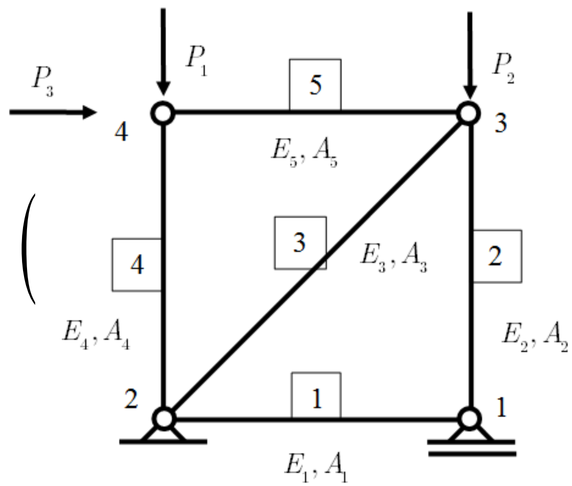
Realistic example

```
double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 )
double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z
+ y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 -
sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y
1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double(
40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double
( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y
double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) (
x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z
( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) do
uble( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin(
double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) )
) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500
) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z
( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 +
y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin(
 double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 )
double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) do
uble( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + (
z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 -
sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z +
y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40
) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double(
40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double
( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double(
500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y
/ z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin(
double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 -
sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 )
double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500
) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 )
x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y
1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1
+ y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double
( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin(
double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) +
( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) )
( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z
+ y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 -
sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( doubl
e( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double(
500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double(
500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z +
y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin(
 double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z
( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) do
uble( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 )
double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2
) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) )
) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z (
x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin(
double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( d
ouble( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 )
double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) )
) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x (
z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 -
sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1
- sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500
) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double(
40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z +
y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) (
x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y /
z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin(
double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 4
0 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) )
) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500
) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z (
z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( doubl
e( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1
+ y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double(
500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 )
double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x (
z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) )
( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 +
 y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) + ( z + y / z ( z + y / z ( x1 + y1 - sin( double( 40 ) double( 500 ) ) ) ) ( x1 + y1 - sin( double(
40 ) double( 500 ) ) ) ) y double( 2 ) x ( z + y / z ( z + y / z ( x1 + y1 - sin( doubl
```

Automatically generated equations of the truss structures

symbolic calculations

$\Phi\Bigg($



$\Bigg) = \qquad K(p)u = Q(p)$

$$\frac{d}{dx}\left(EA\frac{du}{dx}\right) + n = 0$$

Efficiency of rewriting systems

- In average scientist can write less than 10 pages per day.
- Automated systems can generate 1 000 000 pages in 1 hour.

http://andrzej.pownuk.com/publications/test-0-5000.docx
(part of 1 000 000 pages document)

Example output

Table 260: Interval data (uncertainty 37%)

Name	Value	Units
E	[126000000000, 274000000000]	$[\frac{N}{m^2}]$
A	0.0001	$[m^2]$
P	[630, 1370]	$[N]$
L	1	$[m]$
H	1	$[m]$

Table 261: Interval displacements (uncertainty 37%).

u	$\underline{u}[m]$	$mid(\mathbf{u})[m]$	$\overline{u}[m]$
1	1.21305600352052E-05	0.000100990965484165	0.000252372472465261
2	-6.85996385258009E-05	-1.05605747310197E-05	4.59064536921089E-05
3	1.3919764776875E-05	9.04303907531449E-05	0.000230202461510371
4	-0.000147816134050454	-6.05605747310197E-05	-2.12951211825158E-05

Table 262: Combinations (uncertainty 37%, -1 - lower bound, 1 - upper bound, 0 - mid point solution).

u	E_1	E_2	E_3	E_4	E_5	P_1	P_2	P_3
1 (inf)	-1	1	-1	-1	1	-1	1	-1
1 (sup)	-1	-1	-1	-1	-1	1	-1	1
monotone	no	yes	no	no	yes	yes	yes	yes
2 (inf)	-1	1	1	1	-1	-1	1	-1
2 (sup)	-1	-1	-1	-1	1	1	-1	1
monotone	no	yes	yes	yes	yes	yes	yes	yes
3 (inf)	-1	1	1	-1	1	-1	1	-1
3 (sup)	-1	-1	1	-1	-1	1	-1	1
monotone	no	yes	no	no	yes	yes	yes	yes
4 (inf)	-1	-1	1	1	-1	1	-1	1
4 (sup)	-1	1	1	1	1	-1	1	-1
monotone	no	yes	no	no	yes	yes	yes	yes

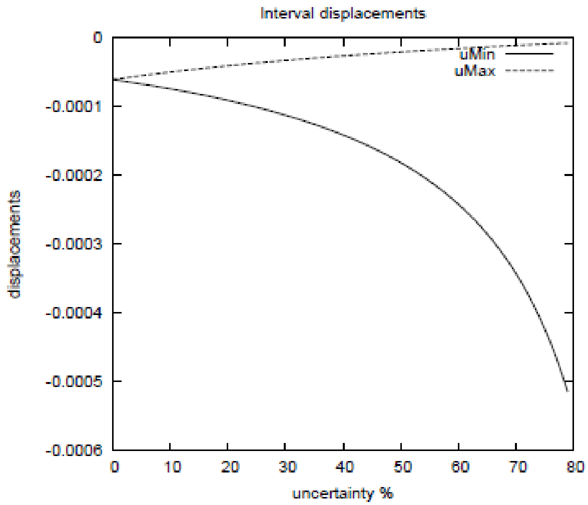
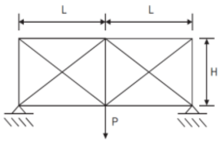


Figure 4: Uncertainty in displacements: u_2

appendix-2-truss-11-bar-551p.pdf

Equations with the interval parameters

- Automatically generated examples.



- Automatically generated methods of solution.
- Automatic research on how these methods are related.
- Automatically generated reports.
- How many examples? Thousands, millions … … as many as you want.

Automated theorem proving

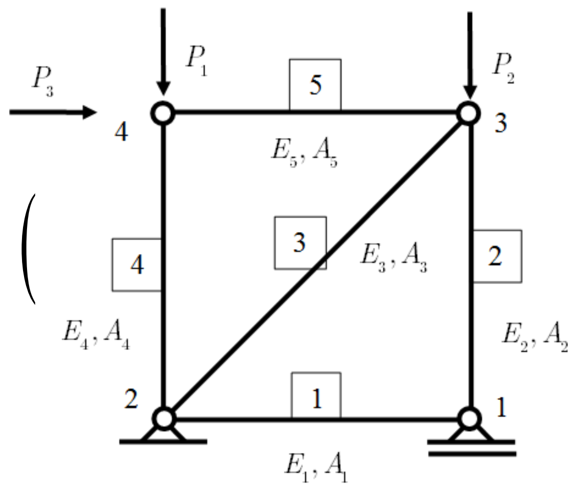
- The method generate not only the final result of the calculations but also all intermediate steps of the calculations.

?

Example application

symbolic calculations

$\Phi\Bigg($



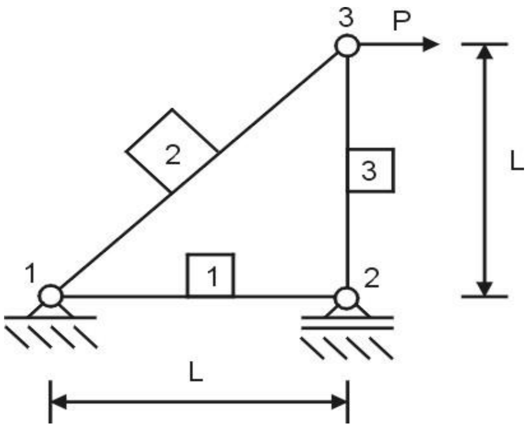
$\Bigg) =$

$$K(p)u = Q(p)$$

$$\frac{d}{dx}\left(EA\frac{du}{dx}\right) + n = 0$$

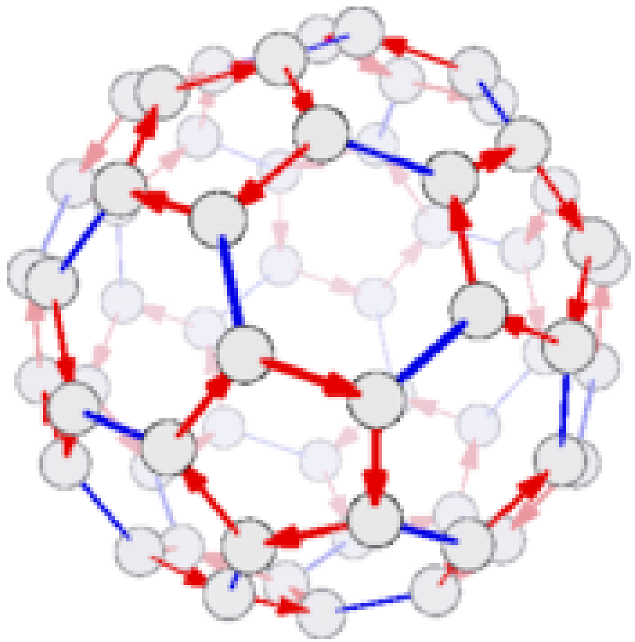
Example application

```
/PREP7
ET,1,LINK1

N, 1, 0, 0
N, 2, 1, 0
N, 3, 1, 1

MP, EX, 1, 2.1e+11
R, 1, 0.0025
MAT  1
REAL 1

E, 1, 2
E, 1, 3
E, 2, 3

F, 3, FX, 10000

D, 1, UX, 0
D, 1, UY, 0
D, 2, UX, 0
```



Example applications

- Automated reasoning
  on Sobolev Spaces, group theory etc.

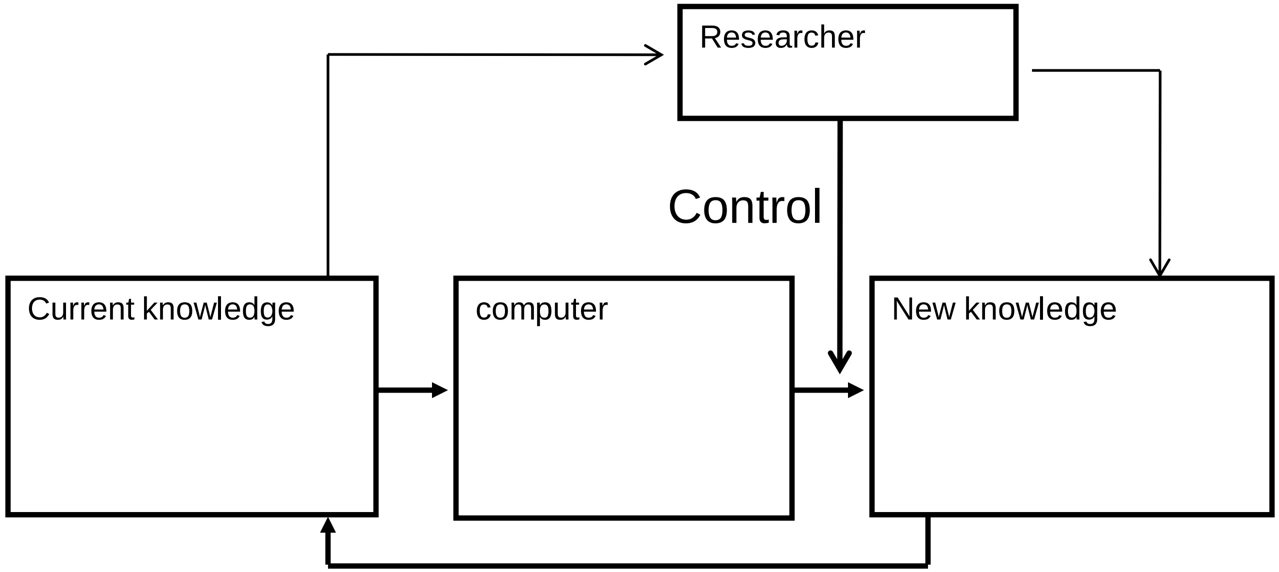


Computational creativity

- Depending on the amount of background information and the context in which this background information is applied, it is possible to get new scientific conclusions with complete justification (proofs).
- How many proofs/theorems … thousands, millions ... as many as you want.

Science today

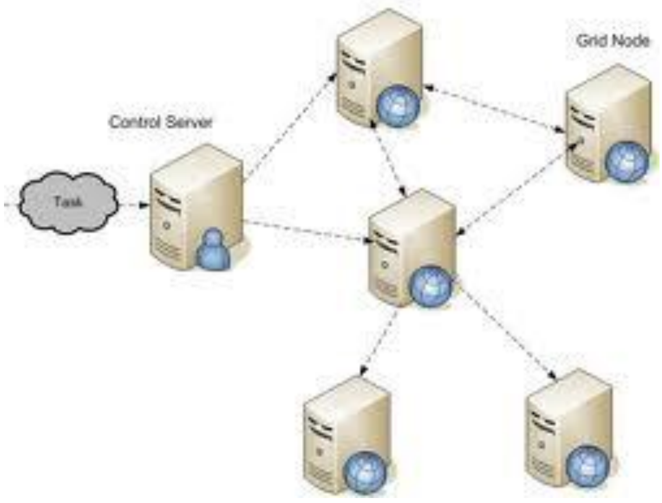
Science tomorrow



HPC Computing

- Background algorithms are embarrassingly parallel and can be significantly speed up
by using HPC computing.

HPC Computing



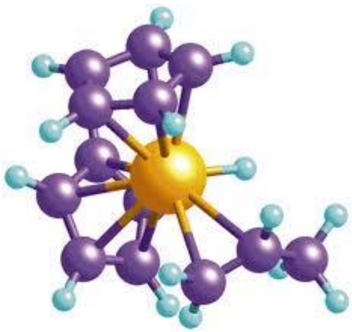
Total amount of cores from all Top500 supercomputers (June 2011) is 7 779 924.

There are between 900 million and one billion personal computers in the world right now.

Interdisciplinary science

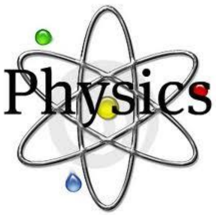
- The method can be applied in any scientific areas which can be described by abstract mathematical concepts.

Chemistry



Biology

Physics

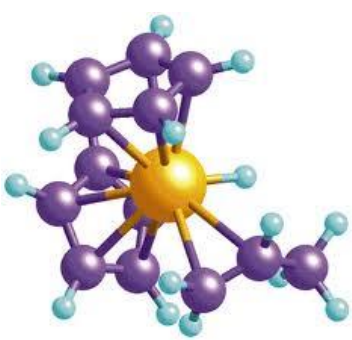


Interdisciplinary science

- The method can be applied in any scientific areas which can be described by abstract mathematical concepts.

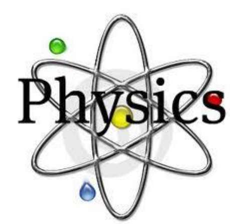
Mathematics is the queen of sciences

Chemistry

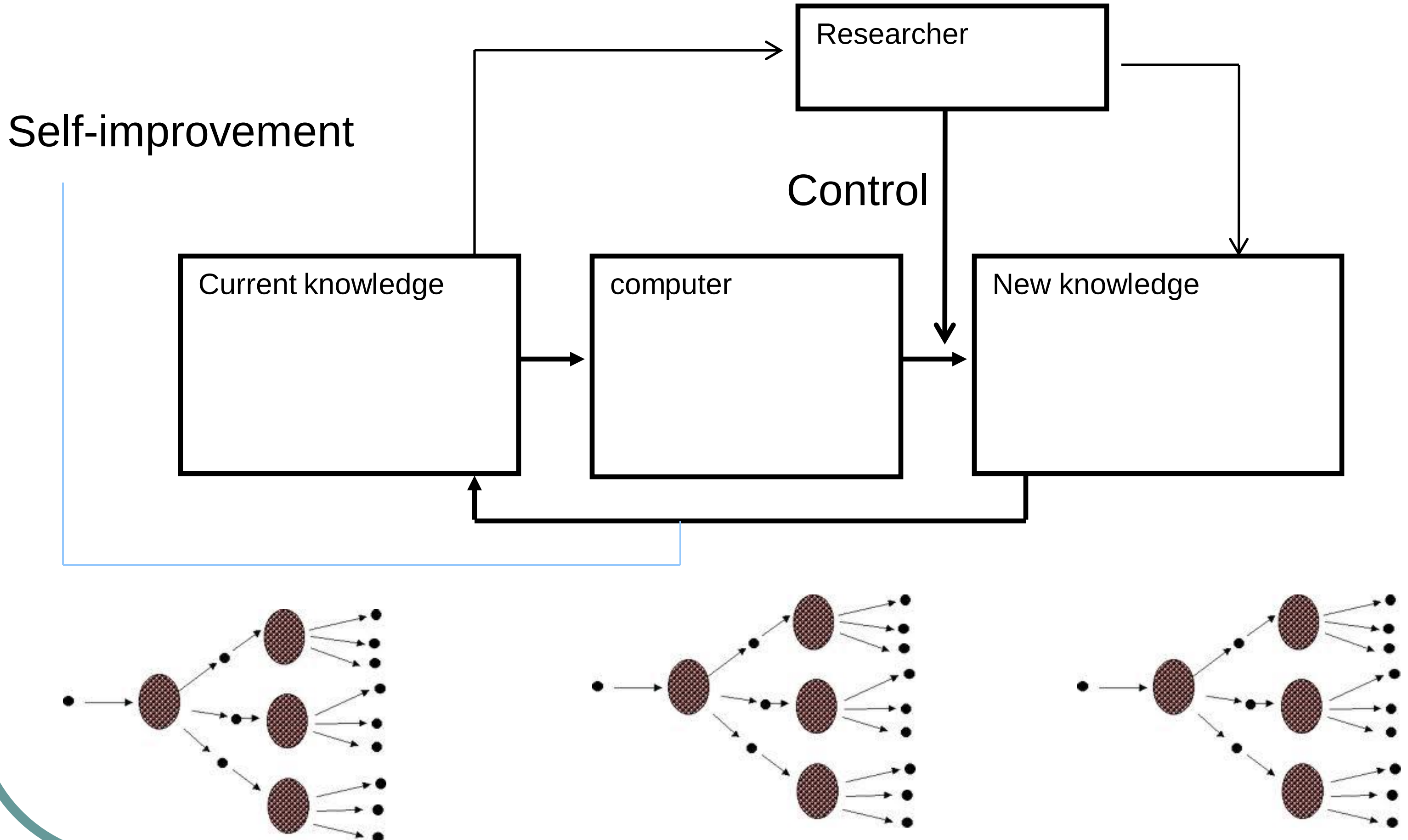


Biology

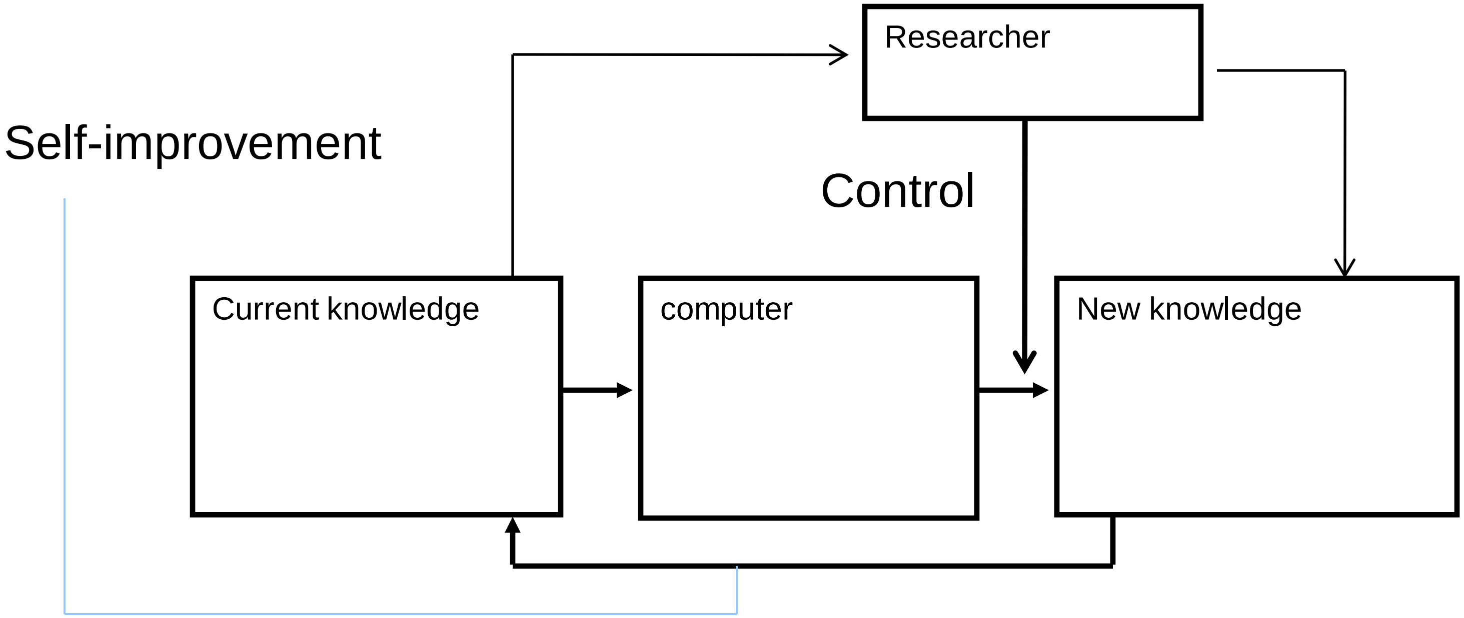
Physics



“Chain reaction” of knowledge



"Chain reaction" of knowledge



According to my research very little amount of background knowledge may generate very big data set of conclusions. New conclusions may lead to even bigger amounts of new knowledge. This process can be continued practically forever.

What kind of language?

- C, C++, FORTRAN
- ALGOL, R, Matlab, Mathematica, FORTH, Cobol, C#, F#, Scala, Lisp, Java, Assembler, Miranda, OCaml, Perl, Prolog, Objective-C, Pascal, PHP, HTML, ASP.NET etc.

Mathematics - The Language of Science

- “Philosophy is written in this grand book, the universe which stands continually open to our gaze. But the book cannot be understood unless one first learns to comprehend the language and read the letters in which it is composed. It is written in the language of mathematics, and its characters are triangles, circles and other geometric figures without which it is humanly impossible to understand a single word of it; without these, one wanders about in a dark labyrinth.”

Galileo Galilee in Assayer

What kind of language?

- Mathematics can be treated as a programming language
- Any text written in a natural or artificial language can be treated as a programming language

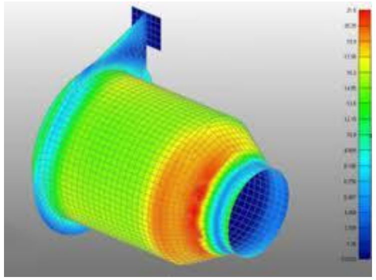
What kind of problems it is possible to investigate by using this tool?

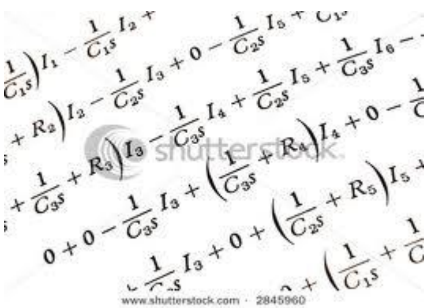
- Almost all currently known mathematical areas
(at this moment there are problems with geometry
but … these problems are solvable).

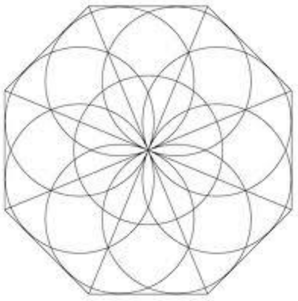
- **Important remark**
Reserch of the problem
does not always lead to solution of that problem.

How it is possible to get background knowledge for the system?

Mathematics is a science of abstract concepts







You are providing an input

More specific examples …

- All examples in this presentation …

and almost anything else …

Platonism

- It is possible to say that in the framework of *SelfNet* system mathematical ideas exist outside of human brain.
- Abstract ideas may exist as a part of computer program.

Limitation of human brain

- It takes 20 years of training to get MS in Mathematics.
- Computer system can process complex problems after several minutes of training.

but …it is hard to compete with the flexibility of human brain

Automated Science

- After retirement many scientific ideas are forgotten.

- Once the idea is added to the system it will never be forgotten and it can improve itself without interactions with humans.

Thank you very much