

# Applications of Autonomous Computational Methods and In Online Learning

Andrew Pownuk

Department of Mathematical Sciences  
University of Texas at El Paso, El Paso, Texas, USA  
ampownuk@utep.edu, <https://andrew.pownuk.com>

26th Joint UTEP/NMSU Workshop on Mathematics,  
Computer Science, and Computational Sciences  
November 6, 2021

# Outline

- 1 Education
- 2 Mathematics
- 3 Engineering
- 4 Generalizations
- 5 Conclusions

# Summary

Education

Mathematics

Engineering

Generalizations

Conclusions

- Education
- Mathematics
- Engineering
- Generalization

# Online Learning

Education

Mathematics

Engineering

Generalizations

Conclusions

**Blackboard**

Institution Page

Andrew Powruk

Activity Stream

Courses

Organizations

Calendar

Messages

Grades

Tools

Sign Out

**Blackboard Help**

Enter Blackboard Central's virtual office | Mon - Fri 8AM - 5PM

24/7 Blackboard Support | 915-747-HELP option # 2

Blackboard Help Website

Provide your Blackboard feedback

**Technology Support Helpdesk**

MON-THU 7AM-9PM   FRI 7AM-8PM   SAT 9AM-2PM   SUN 12PM-5PM
Visit us at <a href="https://helpdesk.utep.edu">helpdesk.utep.edu</a>
Submit a request
Chat with us
Contact us: 915-747-HELP, <a href="mailto:helpdesk@utep.edu">helpdesk@utep.edu</a>

**TECHNOLOGY Support**

# Early Examples of Distance Learning

Education

Mathematics

Engineering

Generalizations

Conclusions



# Online Learning

Education

Mathematics

Engineering

Generalizations

Conclusions

## User Main Page

	Logoff
Username:	80080081
First Name:	Pownuk
Last Name:	Andrew
Group:	2020-Fall-MATH-1312-CRN-12219
	Change password
	Change e-mail

Show my grades

Files (notes, syllabus etc.)

# Online Learning (List of Students, Data Storage)

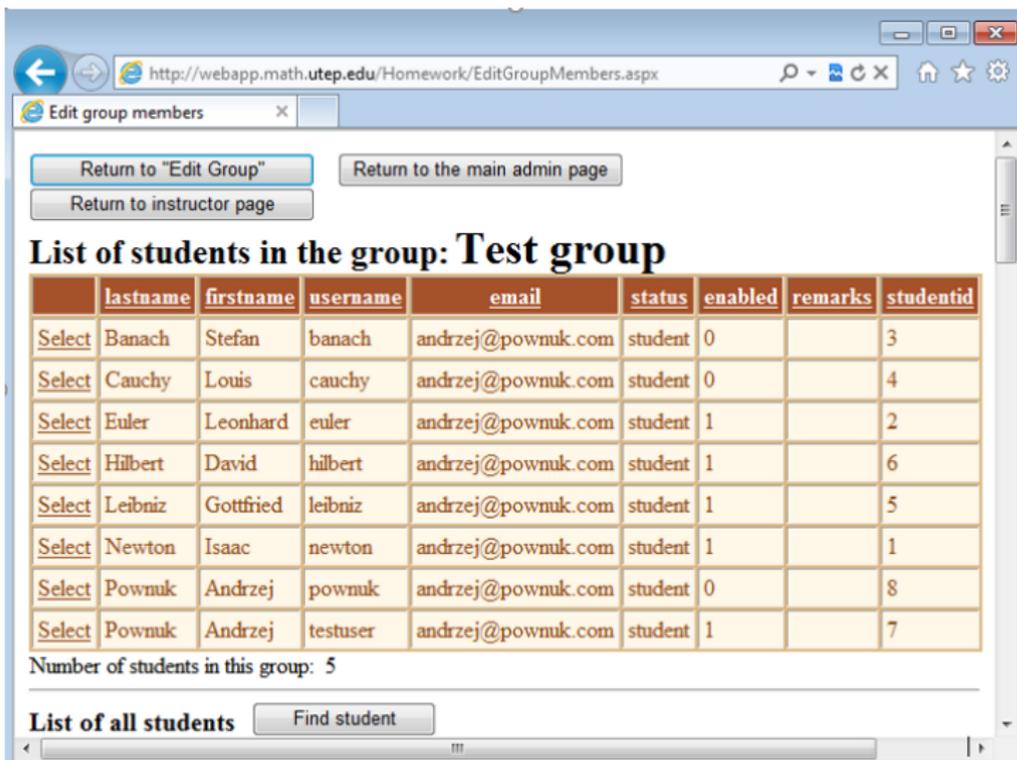
Education

Mathematics

Engineering

Generalizations

Conclusions



The screenshot shows a web browser window with the URL `http://webapp.math.utep.edu/Homework/EditGroupMembers.aspx`. The page title is "Edit group members". At the top, there are three buttons: "Return to 'Edit Group'", "Return to the main admin page", and "Return to instructor page". Below these is the heading "List of students in the group: Test group". A table lists the students with columns for selection, last name, first name, username, email, status, enabled, remarks, and student ID. Below the table, it says "Number of students in this group: 5". At the bottom, there is a link "List of all students" and a "Find student" search box.

	lastname	firstname	username	email	status	enabled	remarks	studentid
<a href="#">Select</a>	Banach	Stefan	banach	andrzej@pow nuk.com	student	0		3
<a href="#">Select</a>	Cauchy	Louis	cauchy	andrzej@pow nuk.com	student	0		4
<a href="#">Select</a>	Euler	Leonhard	euler	andrzej@pow nuk.com	student	1		2
<a href="#">Select</a>	Hilbert	David	hilbert	andrzej@pow nuk.com	student	1		6
<a href="#">Select</a>	Leibniz	Gottfried	leibniz	andrzej@pow nuk.com	student	1		5
<a href="#">Select</a>	Newton	Isaac	newton	andrzej@pow nuk.com	student	1		1
<a href="#">Select</a>	Pownuk	Andrzej	pownuk	andrzej@pow nuk.com	student	0		8
<a href="#">Select</a>	Pownuk	Andrzej	testuser	andrzej@pow nuk.com	student	1		7

Number of students in this group: 5

[List of all students](#)

# Online Learning (Information about the Student)

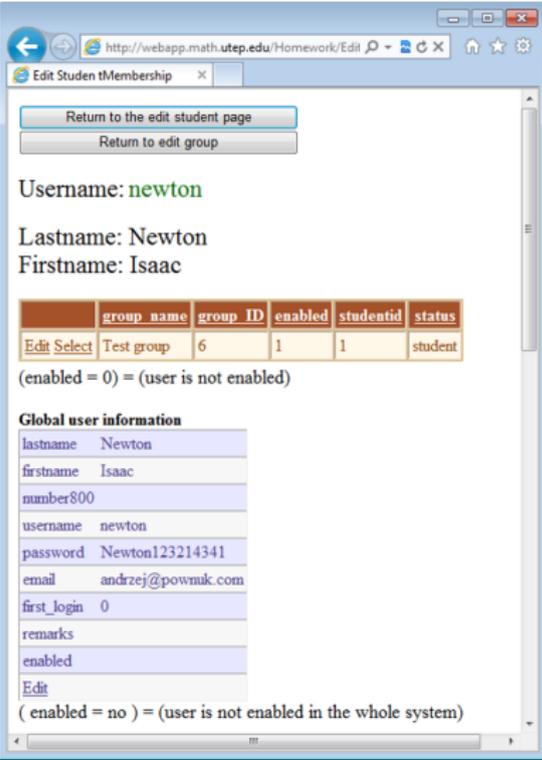
Education

Mathematics

Engineering

Generalizations

Conclusions



The screenshot shows a web browser window with the address bar containing the URL `http://webapp.math.utep.edu/Homework/Edit`. The page title is "Edit Student Membership". At the top, there are two buttons: "Return to the edit student page" and "Return to edit group". Below these, the student's information is displayed: Username: newton, Lastname: Newton, and Firstname: Isaac. A table with columns "group\_name", "group\_ID", "enabled", "studentid", and "status" shows one entry for "Test group" with "enabled" set to 1. Below the table, a note states "(enabled = 0) = (user is not enabled)". The "Global user information" section lists fields like lastname, firstname, number800, username, password, email, first\_login, remarks, and enabled. An "Edit" link is provided at the bottom of this section, with a note "(enabled = no) = (user is not enabled in the whole system)".

Return to the edit student page

Return to edit group

Username: newton

Lastname: Newton

Firstname: Isaac

	group_name	group_ID	enabled	studentid	status
<a href="#">Edit</a> <a href="#">Select</a>	Test group	6	1	1	student

(enabled = 0) = (user is not enabled)

**Global user information**

lastname	Newton
firstname	Isaac
number800	
username	newton
password	Newton123214341
email	andrzej@powmuk.com
first_login	0
remarks	
enabled	

[Edit](#)

(enabled = no) = (user is not enabled in the whole system)

# Online Learning (Online Homework)

Education

Mathematics

Engineering

Generalizations

Conclusions

The screenshot shows a web browser window with the address bar containing `http://webapp.math.utep.edu/Homework/EditHomeworkList.aspx`. The page title is "Untitled Page". Below the browser window is a table with the following data:

<a href="#">Edit</a>	<a href="#">Select</a>	42	TG-Homework-2	/HomeworkDir/TG-Homework-2.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	43	Cal-III-Homework-13	/HomeworkDir/Cal-III-Homework-13.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	44	Cal-II-Homework-12	/HomeworkDir/Cal-II-Homework-12.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	45	AA-I-Homework-12	/HomeworkDir/AA-I-Homework-12.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	46	Cal-III-Homework-14	/HomeworkDir/Cal-III-Homework-14.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	47	Cal-II-Homework-13	/HomeworkDir/Cal-II-Homework-13.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	48	AA-I-Homework-13	/HomeworkDir/AA-I-Homework-13.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	49	AA-I-Homework-14	/HomeworkDir/AA-I-Homework-14.aspx
<a href="#">Edit</a>	<a href="#">Select</a>	50	AA-I-Homework-15	/HomeworkDir/AA-I-Homework-15.aspx

Below the table is a navigation bar with the numbers 1 2 3 4. The main content area contains the following text:

Find Laurent series of the function

$$f(z) = \frac{\sin(2z)}{z^2}$$

at

$$z_0 = 0$$

$a_{-1} =$

$a_0 =$

# Online Learning (Online Homework)

Education

Mathematics

Engineering

Generalizations

Conclusions

Untitled Page - Windows Internet Explorer

http://localhost:54348/Homework/HomeworkDir/Cal-III-Homework-4.aspx

Live Search

Untitled Page

Calculate the equation of plain which pass through three points

A=(1,0,0)  
B=(0,1,0)  
C=(0,0,-2)

$\vec{n} = \overline{AB} \times \overline{AC} = [ \text{input} , \text{input} , \text{input} ]$

Equation of plain (for example  $x+y-z-2=0$ )

=0

(symbolic formula)

Submit the answer

Done Local intranet | Protected Mode: On 100%

# Online Learning (Online Visualization)

Education

Mathematics

Engineering

Generalizations

Conclusions

Lx	10	
Ly	10	
ax	30	
ay	30	
k	1	
q	100	
dt	0.05	
T <sub>Initial</sub>	10	
T <sub>min</sub>	10	
T <sub>max</sub>	50	

Start calculations

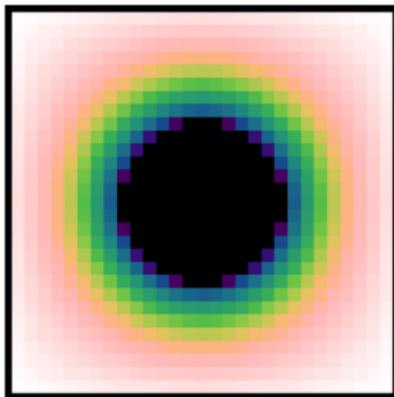


Figure: Solution of the heat transfer equation

# Online Learning (Online Visualization)

Education

Mathematics

Engineering

Generalizations

Conclusions

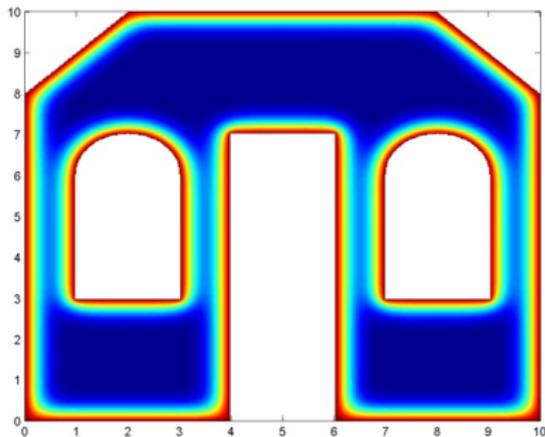


Figure: Solution of the heat transfer equation

$$\lambda \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} \right) + \dot{q} = \rho c_p \frac{\partial T}{\partial t}$$

# Online Learning (Online Visualization)

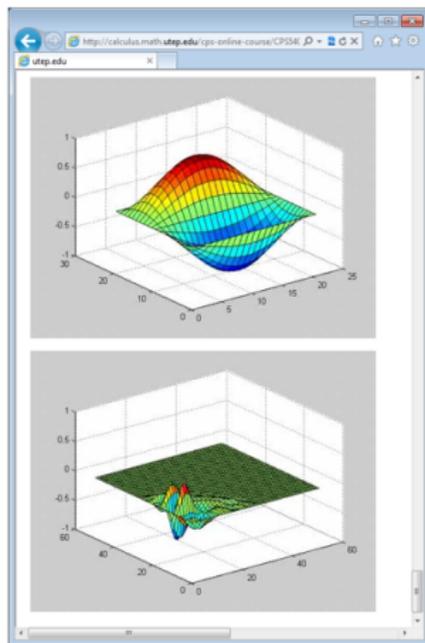
Education

Mathematics

Engineering

Generalizations

Conclusions



$$\begin{cases} -c \left( \frac{\partial^4 u}{\partial x^4} + \frac{\partial^4 u}{\partial y^4} \right) + g = \frac{\partial^2 u}{\partial t^2} \\ u(x, y) = 0, \quad \text{for } (x, y) \in \partial\Omega \\ \frac{\partial^2 u}{\partial x^2}(0, y, t) = \frac{\partial^2 u}{\partial x^2}(L, y, t) = 0 \\ \frac{\partial^2 u}{\partial y^2}(x, 0, t) = \frac{\partial^2 u}{\partial y^2}(x, L, t) = 0 \\ u(x, y, 0) = u^*(x, y) \end{cases}$$

Figure: Vibration of plates

# Sample Problem

Education

Mathematics

Engineering

Generalizations

Conclusions

- Find area of the parallelogram for  $\bar{a} = [1, 2, 3]$ ,  $\bar{b} = [3, 2, 1]$ . Answer:  $A = |\bar{a} \times \bar{b}| = 4\sqrt{6}$ .
- How to input  $4\sqrt{6}$  into the system in order to provide the answer?
- It is possible to use text description of the expression. For example:
  - $4 * \text{sqrt}(6)$
  - $4 * \text{Sqrt}[6]$
  - $4\text{sqrt}(6)$
  - $4\text{sqrt}6$
  - $4 \cdot \text{sq}6$
  - etc.

# Parse Tree

Education

Mathematics

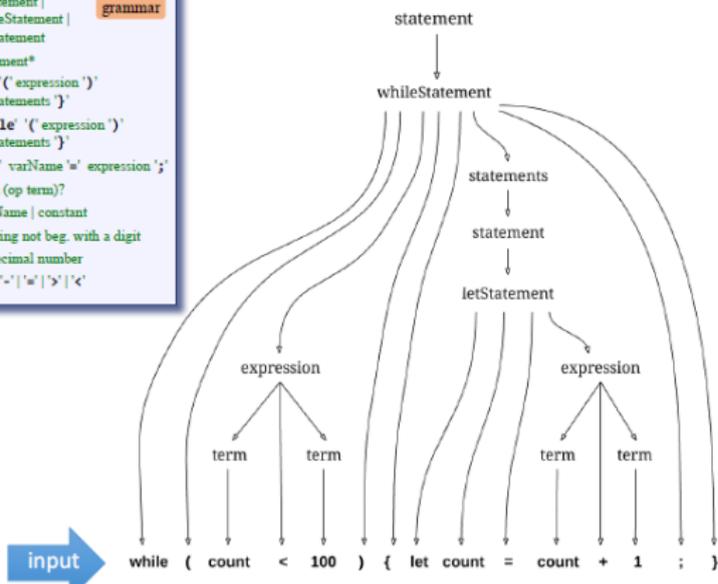
Engineering

Generalizations

Conclusions

## Parse tree

statement:	ifStatement   whileStatement   letStatement	grammar
statement:	statement*	
ifStatement:	'if' '(' expression ')' '{' statements '}'	
whileStatement:	'while' '(' expression ')' '{' statements '}'	
letStatement:	'let' varName 'w' expression ';' ;	
expression:	term (op term)?	
term:	varName   constant	
varName:	a string not beg. with a digit	
constant:	a decimal number	
op:	'+' '-' '*' '/' '<' '>'	



# XML Parse Tree

Education

Mathematics

Engineering

Generalizations

Conclusions

## Parse tree

```
statement: ifStatement |
          whileStatement |
          letStatement |
          grammar
statement: statement*
ifStatement: 'if' '(' expression ')'
            '(' statements ')'
whileStatement: 'while' '(' expression ')'
              '(' statements ')'
letStatement: 'let' varName '=' expression ';'
expression: term (op term)?
term: varName | constant
varName: a string not beg. with a digit
constant: a decimal number
op: '+' '-' '*' '/' '>' '<'
```

grammar

```
<whileStatement>
<keyword> while </keyword>
<symbol> ( </symbol>
<expression>
  <term>
    <identifier> count </identifier>
  </term>
  <symbol> + </symbol>
  <term>
    <IntConstant> 100 </IntConstant>
  </term>
</expression>
<symbol> ) </symbol>
<symbol> { </symbol>
<statements>
  <letStatement>
    <keyword> let </keyword>
    <identifier> count </identifier>
    <symbol> = </symbol>
    <expression>
      <term> <identifier> count </identifier> </term>
      <symbol> + </symbol>
      <term> <IntConstant> 1 </IntConstant> </term>
    </expression>
    <symbol> ; </symbol>
  </letStatement>
</statements>
<symbol> } </symbol>
</whileStatement>
```

parser output

Same parse tree,  
in XML

# Grammar

Education

Mathematics

Engineering

Generalizations

Conclusions

<b>Lexical elements:</b>	The Jack language includes five types of terminal elements (tokens):
keyword:	<code>class</code>   <code>constructor</code>   <code>function</code>   <code>method</code>   <code>field</code>   <code>static</code>   <code>var</code>   <code>int</code>   <code>char</code>   <code>boolean</code>   <code>void</code>   <code>true</code>   <code>false</code>   <code>null</code>   <code>this</code>   <code>let</code>   <code>do</code>   <code>if</code>   <code>else</code>   <code>while</code>   <code>return</code>
symbol:	<code>{</code>   <code>}</code>   <code>(</code>   <code>)</code>   <code>[</code>   <code>]</code>   <code>;</code>   <code>,</code>   <code>'</code>   <code>"</code>   <code>`</code>   <code>~</code>   <code>^</code>   <code>%</code>   <code>&amp;</code>   <code>*</code>   <code>/</code>   <code> </code>   <code>&lt;</code>   <code>&gt;</code>   <code>=</code>   <code>!</code>   <code>~</code>
integerConstant:	A decimal number in the range 0 .. 32767.
StringConstant:	"" A sequence of Unicode characters not including double quote or newline ""
identifier:	A sequence of letters, digits, and underscore ('_') not starting with a digit.
<b>Program structure:</b>	A Jack program is a collection of classes, each appearing in a separate file. The compilation unit is a class. A class is a sequence of tokens structured according to the following context free syntax:
class:	<code>class</code> 'className' '{' classVarDec* subroutineDec* '}'
classVarDec:	<code>(static</code>   <code>field</code> ) type varName (',' varName)* ';'
type:	<code>int</code>   <code>char</code>   <code>boolean</code>   className
subroutineDec:	<code>(constructor</code>   <code>function</code>   <code>method</code> ) <code>(void</code>   type) subroutineName '(' parameterList ')' subroutineBody
parameterList:	( (type varName) (',' type varName)* )
subroutineBody:	'{' varDec* statements '}'
varDec:	<code>var</code> type varName (',' varName)* ';'
className:	identifier
subroutineName:	identifier
varName:	identifier
<b>Statements:</b>	
statements:	statement*
statement:	letStatement   ifStatement   whileStatement   doStatement   returnStatement
letStatement:	<code>let</code> varName '(' (' expression ')' ) <sup>1</sup> '=' expression ';'
ifStatement:	<code>if</code> '(' expression ')' '{' statements '}' ( <code>else</code> '{' statements '}' )?
whileStatement:	<code>while</code> '(' expression ')' '{' statements '}'
doStatement:	<code>do</code> subroutineCall ';'
ReturnStatement:	<code>return</code> expression? ';'
<b>Expressions:</b>	
expression:	term (op term)*
term:	integerConstant   stringConstant   keywordConstant   varName   varName '(' expression ')'   subroutineCall   '(' expression ')' unaryOp term
subroutineCall:	subroutineName '(' (expressionList)   (className varName) ',' subroutineName '(' expressionList ')'
expressionList:	(expression (',' expression)* )?
op:	<code>+</code>   <code>-</code>   <code>*</code>   <code>/</code>   <code>%</code>   <code>&amp;</code>   <code>*</code>   <code>/</code>   <code>&lt;</code>   <code>&gt;</code>   <code>!</code>   <code>~</code>
unaryOp:	<code>-</code>   <code>!</code>
KeywordConstant:	<code>true</code>   <code>false</code>   <code>null</code>   <code>this</code>

# Infix notation, Prefix notation, Postfix notation

Education

Mathematics

Engineering

Generalizations

Conclusions

Different notation for arithmetic expressions.

- Infix notation  $(5 + 6) \times 7$
- Prefix notation  $x + 567$
- Postfix notation  $756 + x$

Typical evaluation process of arithmetic expressions.

- $\text{InfixToPrefix}((5 + 6) \times 7) = x + 567$
- $\text{EvaluatePrefix}(x + 567) = 77$

# Expression Tree

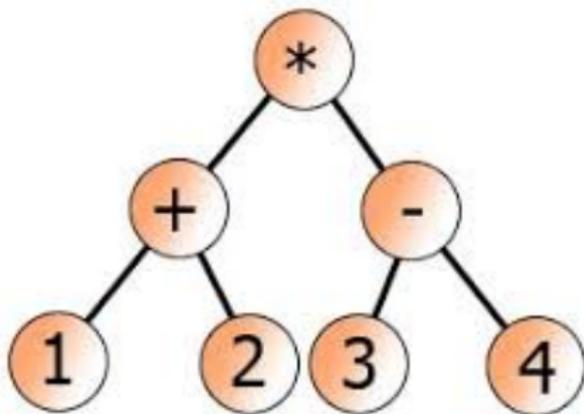
Education

Mathematics

Engineering

Generalizations

Conclusions



$((1+2)*(3-4))$

# How to Evaluate Mathematical Expression Given as a String?

Education

Mathematics

Engineering

Generalizations

Conclusions

## Example

- Expression: `sqrt(2)*6`
- Value:  $\sqrt{2} \cdot 6$

## Example

- Expression: `sqrt2*6`
- Value:  $\sqrt{2} \cdot 6$

## Example

- Expression: `Sqrt[2]6`
- Value:  $\sqrt{2} \cdot 6$

## Example

- Expression: `SQRT[2]6`
- Value:  $\sqrt{2} \cdot 6$

# Automated Generation of Homework Assignments

Education

Mathematics

Engineering

Generalizations

Conclusions

Differentiation, vector algebra, numerical integration, etc.

- Automatically generated list of formulas with given level of difficulty.
- Latex representation of given formula.
- Evaluation of formulas and generation of tests.
- Appropriate HTML code which implements all elements.
- Upload to server and add integrate with the grading system for appropriate group of students, due dates.

# Theoretical Aspects of Online Learning

Education

Mathematics

Engineering

Generalizations

Conclusions

*Journal of Uncertain Systems*

Published by print and online quarterly, England, UK

ISSN: 1752-8909 ( print )  
ISSN: 1752-8917 (online)

**ACADEMIC**  
World Academic Union

World Academic Press, World Academic Union

---

[Home](#)   [Authors Guide](#)   [Subscription](#)   [Online Journal](#)   [Link](#)

---

[Home](#)

- A. Pownuk, Mathematical aspects of grading student's homework in on-line web applications, *Journal of Uncertain Systems*, 5(2), 141-153, 2011.

# COCONUT Project (Vienna University)

Education

Mathematics

Engineering

Generalizations

Conclusions

AMPL (A Mathematical Programming Language) is an algebraic modeling language to describe and solve high-complexity problems for large-scale mathematical computing.

```
#VARIABLE DEFINITIONS
```

```
var x_1;
```

```
var x_2;
```

```
#OBJECTIVE FUNCTION (maximize or minimize)
```

```
maximize value: x_1 + 2*x_2;
```

```
#CONSTRAINTS
```

```
subject to condition_1: x_1 + 3*x_2 <= 20;
```

```
subject to condition_2: 3*x_1 + x_2 <= 20;
```

```
subject to condition_3: x_1 >= 0;
```

```
subject to condition_4: x_2 >= 0;
```

# COCONUT Project (Vienna University)

Education

Mathematics

Engineering

Generalizations

Conclusions

The optimization problems stored in the work nodes, which are passed to the various inference engines, are kept as directed acyclic graphs (DAG), as well. This representation has big advantages. Hereby, a complete optimization problem is always represented by a single DAG. The vertices of the graph represent operators similar to computational trees. Constants and variables are sources, objective and constraints are sinks of the DAG.

<https://www.mat.univie.ac.at/~neum/glopt/coconut/>

# COCONUT Project (Vienna University)

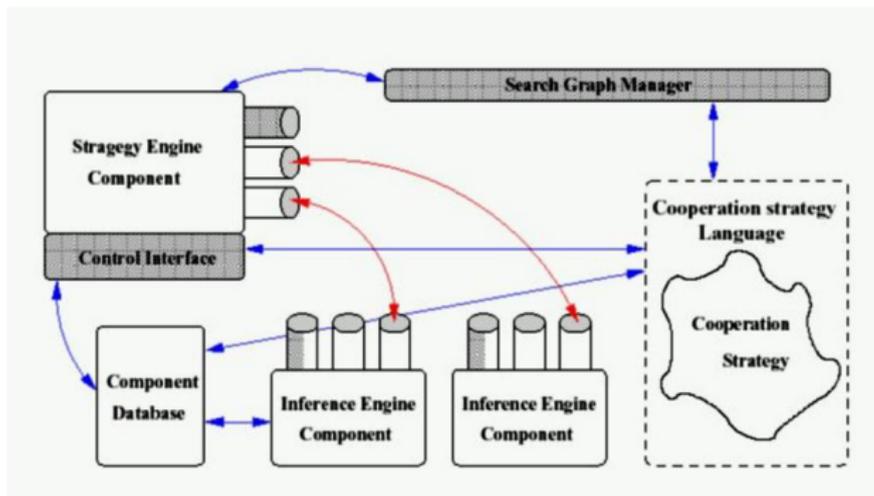
Education

Mathematics

Engineering

Generalizations

Conclusions



<https://www.mat.univie.ac.at/~neum/glopt/coconut/>

# COCONUT Project (Vienna University)

Education

Mathematics

Engineering

Generalizations

Conclusions

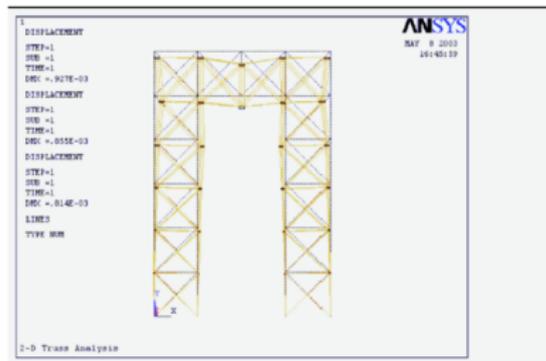


Figure: Modeling of engineering problems with uncertainty

A. Neumaier and A. Pownuk, Linear Systems with Large Uncertainties, with Applications to Truss Structures, Journal of Reliable Computing, 13(2), 149-172, 2007.

# SAGA - Scientific Computing with Algebraic and Generative Abstractions

Education

Mathematics

Engineering

Generalizations

Conclusions

Algebraic software methodologies are a result of the last 20-30 years investigation into abstract data types and algebraic development techniques. The algebraic concepts also abstract modern program structuring mechanisms like classes and generic (or template) modules of object-oriented programming languages such as C++, Generic Java and Fortran-2000.



<https://www.ii.uib.no/saga/>

# SAGA - Scientific Computing with Algebraic and Generative Abstractions

Education

Mathematics

Engineering

Generalizations

Conclusions

- **Sapphire:** For the quick prototyping of mathematical models an **algebraic programming language** and a compiler that translates recursive functions into non-recursive, imperative code was developed. This allows us to code the recursive equations of the mathematical formulation of a solver directly as recursive functions and compile this for both sequential and parallel HPC computers.
- **Sophus:** This is a software library written in C++ and carefully designed to mimic the abstract structure of the PDE mathematics.
- **CodeBoost:** This is a software transformation system being developed to address the gap between well formed code (from a software engineering point of view) and efficient code (from a run-time point of view).

# Fuzzy/Interval Calculator

Insert a description of interval and fuzzy expressions and press "Calculate" button. [\[USER'S MANUAL\]](#)

```
[ [ 0, 6, 14 ] [ 0.25, 6.25, 13.5 ] [ 0.5, 6.5, 13 ] [ 0.75, 6.75, 12.5 ] [ 1, 7, 12 ] ]
```

Calculate The result is:

```
1+2*((0,1)+1)
{[0,1,5], [1,2,3]} + [1,2]*2

#####
#
# The program evaluate the value of expressions
# which contain floating-point, interval and fuzzy numbers.
#
# 1+2*(2+5)           Floating-point expression.
# [1,2]+[2,3]         Interval numbers are defined using upper (Xmax) and lower (Xmin) bounds e.g. [Xmin, Xmax].
#
# By default the + operator is assumed between the lines.
# These two lines are equivalent to the expression 1+2*(2+5) + [1,2]+[2,3].
#
# {[0,1,5], [1,2,3]}   Fuzzy number is a collection of alpha-cuts.
#                       Each alpha-cut is a triple [alpha,Xmin,Xmax]
# {[0,1,5], [1,2,3]}+[1,2]*2 Composite expression can have floating-point, interval and fuzzy numbers.
#
# It is possible to add an operator between the lines.
#
# 1+2
# *
# [1,2]+[2,3]
#
# These two lines are equivalent to the expression 1+2 * [1,2]+[2,3].
#
# The final result is a value of the expression which is created
# from all the data in the data file.
```

[http://www.math.utep.edu/Faculty/ampownuk/php/fuzzy\\_calculator/](http://www.math.utep.edu/Faculty/ampownuk/php/fuzzy_calculator/)

# Interval Arithmetic

Education

Mathematics

Engineering

Generalizations

Conclusions

An binary operation  $\star$  on two intervals, such as addition or multiplication, is defined by

$$[x_1, x_2] \star [y_1, y_2] = \{x \star y \mid x \in [x_1, x_2] \wedge y \in [y_1, y_2]\}$$

- Interval addition  $[x_1, x_2] + [y_1, y_2] = [x_1 + y_1, x_2 + y_2]$

- Interval multiplication

$$[x_1, x_2] \star [y_1, y_2] = [z_1, z_2] \text{ where}$$

$$z_1 = \min\{x_1 \star y_1, x_1 \star y_2, x_2 \star y_1, x_2 \star y_2\},$$

$$z_2 = \max\{x_1 \star y_1, x_1 \star y_2, x_2 \star y_1, x_2 \star y_2\}.$$

- Interval division

$$\frac{[x_1, x_2]}{[y_1, y_2]} = [x_1, x_2] \cdot \frac{1}{[y_1, y_2]}$$

# Fuzzy/Interval Calculator

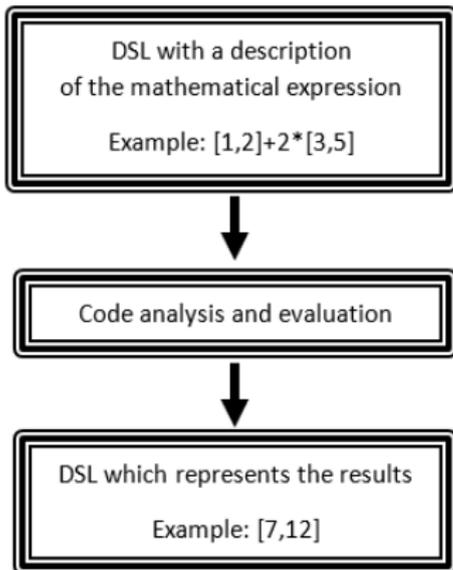
Education

Mathematics

Engineering

Generalizations

Conclusions



DSL - Domain Specific Language

# Fuzzy/Interval Calculator

Education

Mathematics

Engineering

Generalizations

Conclusions

DSL for description of fuzzy and interval numbers.

Input:

$$1+2*([0,1]+1)$$

$$\{[0,1,5], [1,2,3]\} + [1,2]*2$$

Output:

$$\begin{array}{l} \{ [ 0, 6, 14 ] \quad [ 0.25, 6.25, 13.5 ] \\ [ 0.5, 6.5, 13 ] \quad [ 0.75, 6.75, 12.5 ] \\ [ 1, 7, 12 ] \quad \} \end{array}$$

# Fuzzy Random Variables

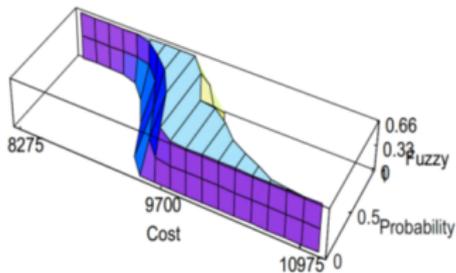
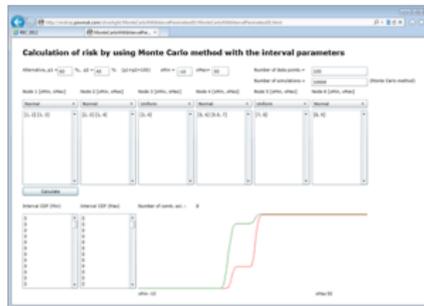
Education

Mathematics

Engineering

Generalizations

Conclusions



# Differential Equations with Uncertain Parameters

Education

Mathematics

Engineering

Generalizations

Conclusions



Figure: Second order differential equation

# Differential Equations with Uncertain Parameters

The screenshot shows a web browser window with the URL `http://andrei.gownik.com/3d-vlight/VibrationsWithIntervalParameters/VibrationsWithIntervalParameters.html`. The interface is titled "VibrationsWithIntervalPara...".

**Input Parameters:**

- $E = 200E9$ ,  $dE = 5$  %
- $A = 0.01$ ,  $dA = 5$  %
- $J = 8.333E-6$ ,  $dJ = 5$  %
- $dI = 0.001$  [s]
- $L = 10.0$  [m],  $n = 1$
- $L_n = 5$  [m]
- $\min E = 19000000000$ ,  $\max E = 21000000000$
- $\min A = 0.0095$ ,  $\max A = 0.0105$
- $\min J = 7.91635E-06$ ,  $\max J = 8.74965E-06$
- $P = 1000$  [N]
- Time steps for load = 1
- $\rho = 7874$  [kg/m<sup>3</sup>]
- $d\rho = 5$  %
- $\min \rho = 7480.3$ ,  $\max \rho = 8267.7$
- Total time when the load was applied = 0.001 [s]

**Init calculations:** Number of interval parameters = 8

List of nodes: Number of timesteps = 600

- node 1,  $x = 0$
- node 2,  $x = 5$
- node 3,  $x = 10$

Number of Dof = 9

number of elements = 2

Number of nodes = 3

Dof in nodes:

- node 0: 0 1 2
- node 1: 3 4 5
- node 2: 6 7 8

Nodes in elements:

- element 0: 0 1 2
- element 1: 3 4 5

Dof in elements:

- element 0: 0 1 2
- element 1: 3 4 5

The diagram shows a beam of length  $L$  supported at both ends. A downward point load  $P$  is applied at a distance  $a=3$  from the right end. The beam is divided into two elements of length  $L_n=5$  by a node at  $x=5$ .

Figure: Input parameters

# Online Learning (Numerical Analysis, 1998)

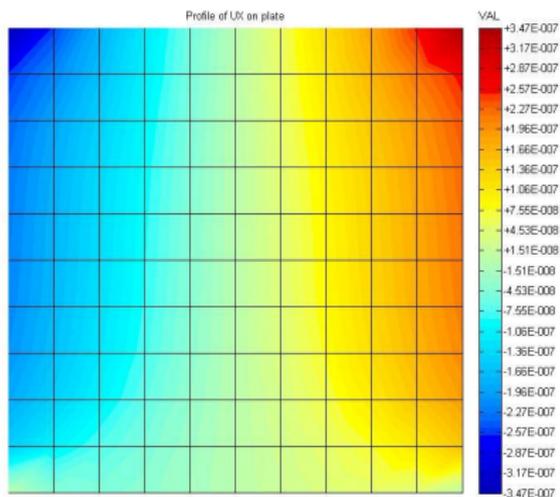
Education

Mathematics

Engineering

Generalizations

Conclusions



**Figure:** Web application for teaching of the finite element method. Description of the problem was given in some DSL.

# Teaching (Computer Methods in Mechanics)

Education

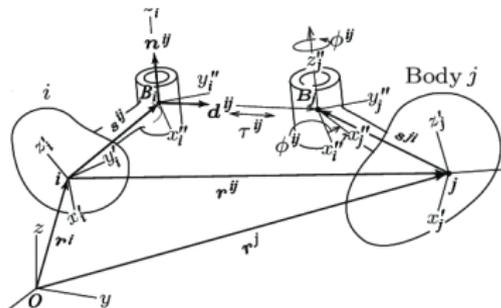
Mathematics

Engineering

Generalizations

Conclusions

## Lagrangian Mechanics



$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_j} \right) = \frac{\partial L}{\partial q_j}$$

Computer methods for finding analytical formulation of the equations of motion in multibody dynamics.

# Teaching (Computer Methods in Mechanics)

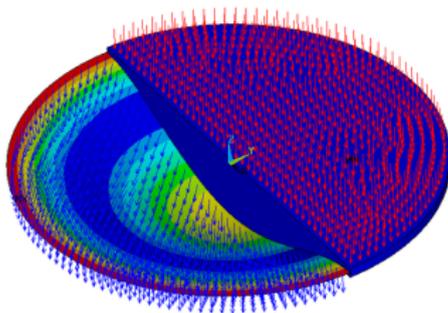
Education

Mathematics

Engineering

Generalizations

Conclusions



Computer algebra software (Mathematica, Derive, etc.)  
for numerical methods in the theory of plates.

$$\frac{2Eh^3}{3(1-\nu)} \left( \frac{\partial^4 w}{\partial x_1^4} + 2 \frac{\partial^4 w}{\partial x_1^2 \partial x_2^2} + \frac{\partial^4 w}{\partial x_2^4} \right) + q + 2\rho h \frac{\partial^2 u}{\partial t^2} = 0$$

# Teaching (Computer Methods in Mechanics)

Education

Mathematics

Engineering

Generalizations

Conclusions

Computer algebra software (Mathematica, Derive, etc.)  
for numerical methods in the theory of linear elasticity.

$$\frac{1}{2(1-\nu)(1-2\nu)} \left( 2(1-\nu) \frac{\partial^2 u_x}{\partial x^2} + \frac{\partial^2 u_y}{\partial x \partial y} + (1-2\nu) \frac{\partial^2 u_x}{\partial y^2} \right) + b_x = 0$$

$$\frac{1}{2(1-\nu)(1-2\nu)} \left( 2(1-\nu) \frac{\partial^2 u_y}{\partial y^2} + \frac{\partial^2 u_x}{\partial x \partial y} + (1-2\nu) \frac{\partial^2 u_y}{\partial x^2} \right) + b_y = 0$$

# Teaching (Computer Methods in Mechanics)

Education

Mathematics

Engineering

Generalizations

Conclusions

- Partial differential equations of elasticity.
- Partial differential equations of plasticity.
- Partial differential equations of viscoelasticity.
- Partial differential equations of the theory of shells and appropriate theory in curvilinear coordinate systems.
- The theory of thin-walled structures.
- Adaptive mesh refinement.
- The theory of variational equations related to the contact mechanics.
- The theory of crack mechanics (fracture mechanics).
- The theory of heat transfer and multiphysics problems.
- etc.

# Calculation of the Interval risk by Using Petri Networks and interval Probability

Education

Mathematics

Engineering

Generalizations

Conclusions



Figure: DSL for description of the engineering problem

M. Betkowski and A. Pownuk, Calculating risk of cost using Monte Carlo simulation with fuzzy parameters in civil engineering, Proceeding of the NSF Workshop on Reliable Engineering Computing, Savannah, Georgia, USA, 179-192, September 15-17, 2004.

# Interval Finite Element Method for the Truss Structures

Education

Mathematics

Engineering

Generalizations

Conclusions

## Description of the problem:

```
/PREP7
ET,1,LINK1
N, 1, 0, 0
N, 2, 1, 0
N, 3, 2, 0
N, 4, 3, 0
N, 5, 0, 1
N, 6, 1, 1
N, 7, 2, 1
N, 8, 3, 1
N, 9, 0, 2
N, 10, 1, 2
N, 11, 2, 2
N, 12, 3, 2
MP, EX, 1, 2.1e+11
R, 1, 0.0025
MAT 1
REAL 1
```

Description of interval parameters ([help](#))

```
MP, EX, 1, 5
R, 1, 5
```

Sensitivity analysis method

Calculate

<http://www.math.utep.edu/Faculty/ampownuk/php/ansys2interval/ansys-code.php>

# Interval Finite Element Method for the Truss Structures

Education

Mathematics

Engineering

Generalizations

Conclusions

## Results:

```
Time of calculation: 0.004996 [sec]
Number of DOF:      16
Number of elements: 26
Number of nodes:    12

u[ 0]= [ 2.54206368927894e-05, 2.70758977991233e-05, 2.88890319829459e-05] node= 5 dof= 1
u[ 1]= [ -2.41613231842201e-06, -1.45525064589709e-06, -5.5936232302321e-07] node= 5 dof= 2
u[ 2]= [ 1.89488493026688e-05, 2.03244670585942e-05, 2.18240888299457e-05] node= 6 dof= 1
u[ 3]= [ -1.18336781275183e-05, -1.07203077121679e-05, -9.68801242678198e-06] node= 6 dof= 2
u[ 4]= [ 1.74375666485017e-05, 1.86853353510165e-05, 2.00368684309889e-05] node= 7 dof= 1
u[ 5]= [ -1.53016570105788e-05, -1.40293414211092e-05, -1.28438219917361e-05] node= 7 dof= 2
u[ 6]= [ 2.23883755090784e-05, 2.38816715072828e-05, 2.5532229372461e-05] node= 8 dof= 1
u[ 7]= [ -2.43184098360924e-05, -2.27501214588593e-05, -2.13175562611172e-05] node= 8 dof= 2
u[ 8]= [ 4.47984203980532e-05, 4.76520021045755e-05, 5.07482294189415e-05] node= 9 dof= 1
u[ 9]= [ -1.25873042500698e-05, -1.0800778851294e-05, -9.13828295995457e-06] node= 9 dof= 2
u[ 10]= [ 3.58319463043394e-05, 3.83064738991786e-05, 4.09641151999668e-05] node= 10 dof= 1
u[ 11]= [ -2.03184368590144e-05, -1.88999001199072e-05, -1.75638790058709e-05] node= 10 dof= 2
u[ 12]= [ 3.30408793908687e-05, 3.54230615860712e-05, 3.79901925037356e-05] node= 11 dof= 1
u[ 13]= [ -2.87524495626644e-05, -2.70377395621771e-05, -2.54594638852595e-05] node= 11 dof= 2
u[ 14]= [ 3.51831538994549e-05, 3.77051247175134e-05, 4.04232862441163e-05] node= 12 dof= 1
u[ 15]= [ -4.18322390326742e-05, -3.95037800489603e-05, -3.7394527683613e-05] node= 12 dof= 2
```

<http://www.math.utep.edu/Faculty/ampownuk/php/ansys2interval/ansys-code.php>

# Interval Finite Element Method for the Truss Structures

Education

Mathematics

Engineering

Generalizations

Conclusions

## ANSYS Parametric Design Language (APDL)

```
MP, EX, 1, 2.1e+11
```

```
R, 1, 0.0025
```

```
MAT 1
```

```
REAL 1
```

```
...
```

Extension of the ANSYS Parametric Design Language (APDL)  
which describes uncertainty of parameters.

```
MP, EX, 1, 5
```

```
R, 1, 5
```

```
...
```

# Interval Finite Element Method for the 2D Linear Elasticity Problems

Education

Mathematics

Engineering

Generalizations

Conclusions

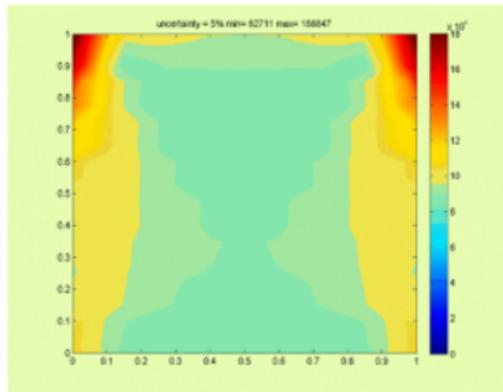
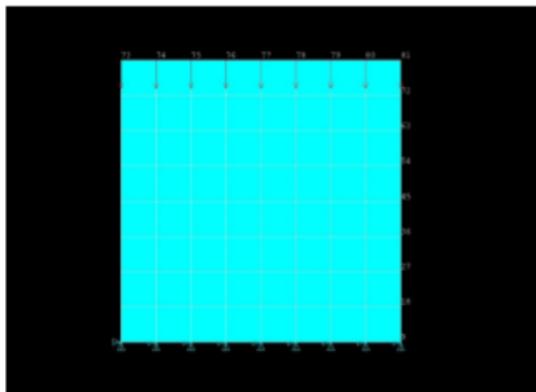


Figure: Solution of system of partial differential equations

# Interval Finite Element Method for the Truss Structures

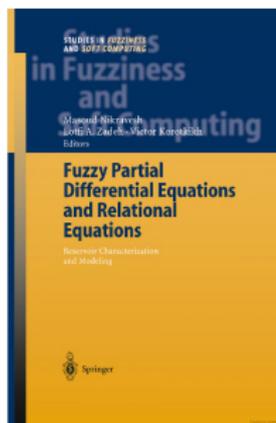
Education

Mathematics

Engineering

Generalizations

Conclusions



A. Pownuk, Numerical solutions of fuzzy partial differential equation and its application in computational mechanics, In: M. Nikravan, L. Zadeh and V. Korotkikh, (eds.), Fuzzy Partial Differential Equations and Relational Equations: Reservoir Characterization and Modeling, 308-347, Springer 2004.

# Chevron Oil Company

Education

Mathematics

Engineering

Generalizations

Conclusions

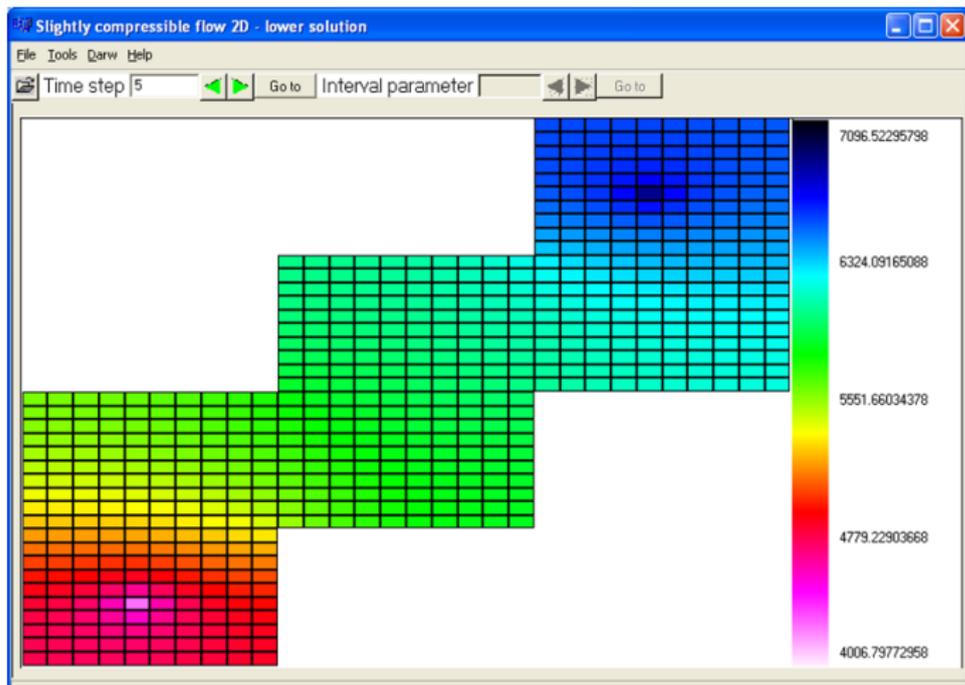


Figure: Research for Chevron Oil Company

# Commercial FEM Software for Designing Truss and Frame Structures

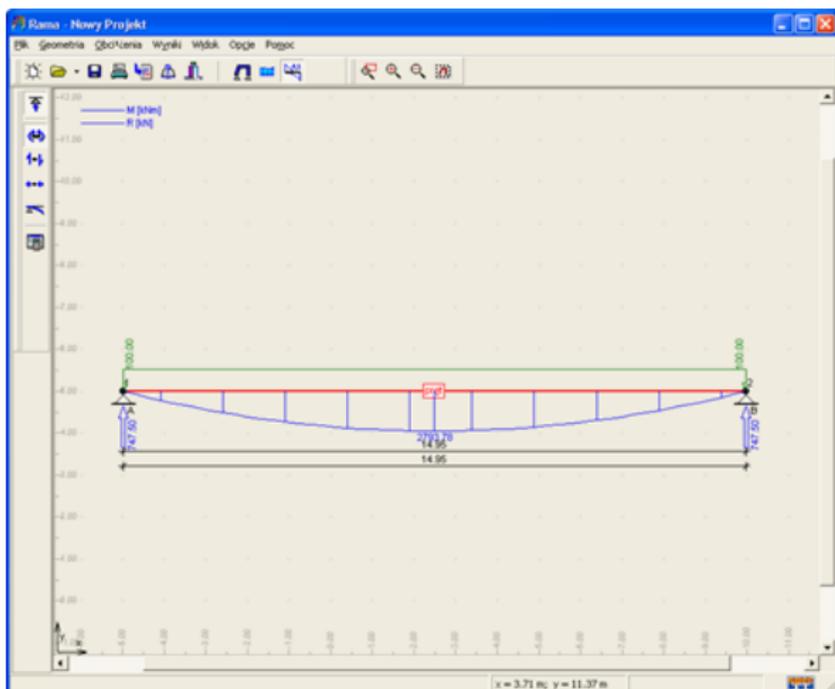
Education

Mathematics

Engineering

Generalizations

Conclusions



# FEM Equations form APDL

Education

Mathematics

Engineering

Generalizations

Conclusions

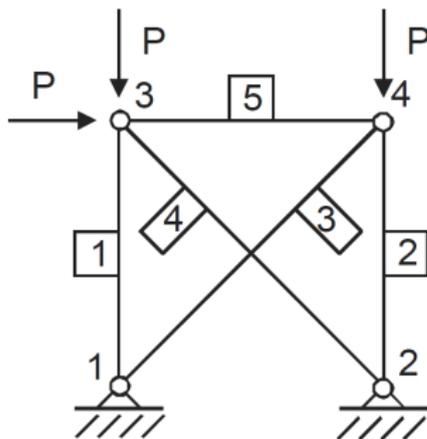


Figure: Sample engineering structure

<http://www.math.utep.edu/Faculty/ampownuk/php/fem-equations/fem-equations.php>

# FEM Equations form APDL

Education

Mathematics

Engineering

Generalizations

Conclusions

```
N 1 0 0
N 2 1 0
N 3 0 1
N 4 1 1

E 1 1 3 MP 1 R 1
E 2 2 4 MP 2 R 2
E 3 1 4 MP 3 R 3
E 4 2 3 MP 4 R 4
E 5 3 4 MP 5 R 5

BC 1 UX UY
BC 2 UX UY

F 3 FX 1000
F 3 FY -1000
F 4 FY -1000
```

● Generate Equations

Calculate

Figure: APDL description of engineering problem

<http://www.math.utep.edu/Faculty/ampownuk/php/fem-equations/fem-equations.php>

# FEM Method

Education

Mathematics

Engineering

Generalizations

Conclusions

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

$$\int_0^L \frac{d}{dx} \left( EA \frac{du}{dx} \right) v dx + \int_0^L n v dx = \int_0^L 0 v dx, u(0) = 0, u(L) = 0$$

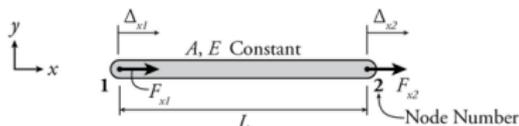
$$\int_0^L \frac{d}{dx} \left( EA \frac{du}{dx} \right) v dx =$$

$$= \int_0^L EA \frac{du}{dx} \frac{dv}{dx} dx + EA \frac{du(0)}{dx} v(0) - EA \frac{du(L)}{dx} v(L)$$

etc.

# Local Stiffness Matrix

1D TRUSS ELEMENT



1D element

$$K = \begin{bmatrix} \frac{EA}{L} & -\frac{EA}{L} \\ -\frac{EA}{L} & \frac{EA}{L} \end{bmatrix}$$

2D element

$$K = \begin{bmatrix} \frac{EA}{L} & 0 & -\frac{EA}{L} & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{EA}{L} & 0 & \frac{EA}{L} & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

etc.

# Global Stiffness Matrix

Education

Mathematics

Engineering

Generalizations

Conclusions

$$\begin{aligned}K[1][1] &= ((E[4]*A[4])/1.414214)*0.500000+(E[5]*A[5])/1.000000 \\K[1][2] &= ((E[4]*A[4])/1.414214)*(-0.500000) \\K[1][3] &= ((E[5]*A[5])/1.000000)*(-1.000000) \\K[1][4] &= 0.000000\end{aligned}$$

$$\begin{aligned}K[2][1] &= ((E[4]*A[4])/1.414214)*(-0.500000) \\K[2][2] &= (E[1]*A[1])/1.000000+((E[4]*A[4])/1.414214)*0.500000 \\K[2][3] &= 0.000000 \\K[2][4] &= 0.000000\end{aligned}$$

$$\begin{aligned}K[3][1] &= ((E[5]*A[5])/1.000000)*(-1.000000) \\K[3][2] &= 0.000000 \\K[3][3] &= ((E[3]*A[3])/1.414214)*0.500000+(E[5]*A[5])/1.000000 \\K[3][4] &= ((E[3]*A[3])/1.414214)*0.500000\end{aligned}$$

$$\begin{aligned}K[4][1] &= 0.000000 \\K[4][2] &= 0.000000 \\K[4][3] &= ((E[3]*A[3])/1.414214)*0.500000 \\K[4][4] &= (E[2]*A[2])/1.000000+((E[3]*A[3])/1.414214)*0.500000\end{aligned}$$

# How to Efficiently use Available Tools?

Education

Mathematics

Engineering

Generalizations

Conclusions

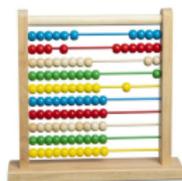


Figure: Tools in the past



Figure: Tools now

# Advantages of the Automated Computational Methods

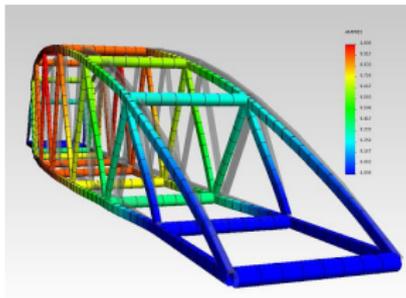
Education

Mathematics

Engineering

Generalizations

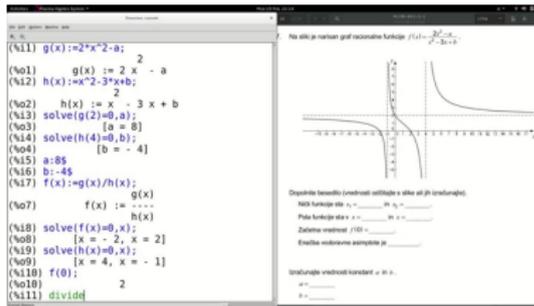
Conclusions



Automation of some part of the engineering computational process:

- faster design,
- more optimal products,
- cheaper engineering structures.

# Advantages of the Automated Computational Methods



Some part of the the computational algorithms can be automated. There are several benefits of this process:

- calculations are faster,
- it is possible to analyse more results,
- it is possible to solve some problems with high complexity.

# Conclusions

Education

Mathematics

Engineering

Generalizations

Conclusions

- Syntax and grammar analysis of the mathematical statements can improve online learning systems.
- Some optimization problems and some aspects of theory of partial differential equations can be can be solved automatically by using special software methodologies.
- Automated development of mathematical models speeds up calculations and and software development.

# Thank You