Wrocław University of Technology

Application of an evolution algorithm and cluster computing for parameters identification of the Friderick-Armstrong's material model

Ł. MACIEJEWSKI, W. MYSZKA, G. ZIĘTEK
Institute of Materials Science and Applied Mechanics

Wrocław 2006

Introduction

Internal parameters

Processes observed in cyclicly loaded material could be modelled by internal parameters introduced into constitutive relation. Following parameters could be distinguished:

- kinematic hardening,
- isotropic hardening,
- damage parameter and others.

Therefore constitutive relation becomes differential or integral equation, very often without a solution as an elementary function.

Material model

General assumptions

- elastic-plastic material model with Frederick-Armstrong's kinematic hardening parameter,
- Hubera-Mises yield condition:

$$\sqrt{(s-X)(s-X)} - R = 0, \qquad (1)$$

where s - stress tensor deviator, X - back stress, R - yield point.

Material model

Constitutive equation

Applying appropriate evolution equation one may achieve

$$\dot{X}_{ij} = C^{(k)}\dot{\varepsilon}^{p}_{ij} - \gamma^{(k)}\dot{\lambda}X^{(k)}_{ij}, \quad (2)$$

where $\dot{\lambda} = \sqrt{\frac{2}{3}\dot{\varepsilon}^{p}_{ij}\dot{\varepsilon}^{p}_{ij}}$ - change of plastic strain trajectory, $C^{(k)}$ i $\gamma^{(k)}$ - parameter that may be functions of internal variables, ε^{p}_{ij} - plastic strain tensor.

Material model

Constitutive equation

In 1D case, for one plasticity surface ($k = 1$) and linear Frederick-Armstrong's evolution equation:

$$\dot{X} = C\dot{\varepsilon}_p - \gamma|\dot{\varepsilon}_p|X \qquad (3)$$

and

$$\sigma = X \pm R. \qquad (4)$$

For constant C i γ equation (3) is a linear equation with a solution as an elementary function

$$\sigma = (R - \sigma_a - \frac{C}{\gamma})e^{-\gamma(\varepsilon_p + \varepsilon_{ap})} + \frac{C}{\gamma}. \qquad (5)$$

Material model

Constitutive equation

If the γ is a function of plastic strain trajectory

$$\gamma = \gamma(\lambda), \quad \lambda = \int_0^t |\dot{\varepsilon}_p| dt. \tag{6}$$

$$\dot{X} = C\dot{\varepsilon}_p - \gamma(\lambda)|\dot{\varepsilon}_p|X \tag{7}$$

Experiment

- austenitic steel AISI 304 ($E = 1,93e5[MPa]$),
- material exhibit strong hardening under cyclic loads,
- round specimen,
- material test system by MTS company,
- load – constant amplitude total strain signal.

Experiment

Modelling material cyclic hardening implies taking:

- variable yield point R,
- variable relaxation parameter γ.

Second option was selected because parameter γ defines relaxation that decreases due to increase of martensitic phase.

Identification procedures

Identification criterion

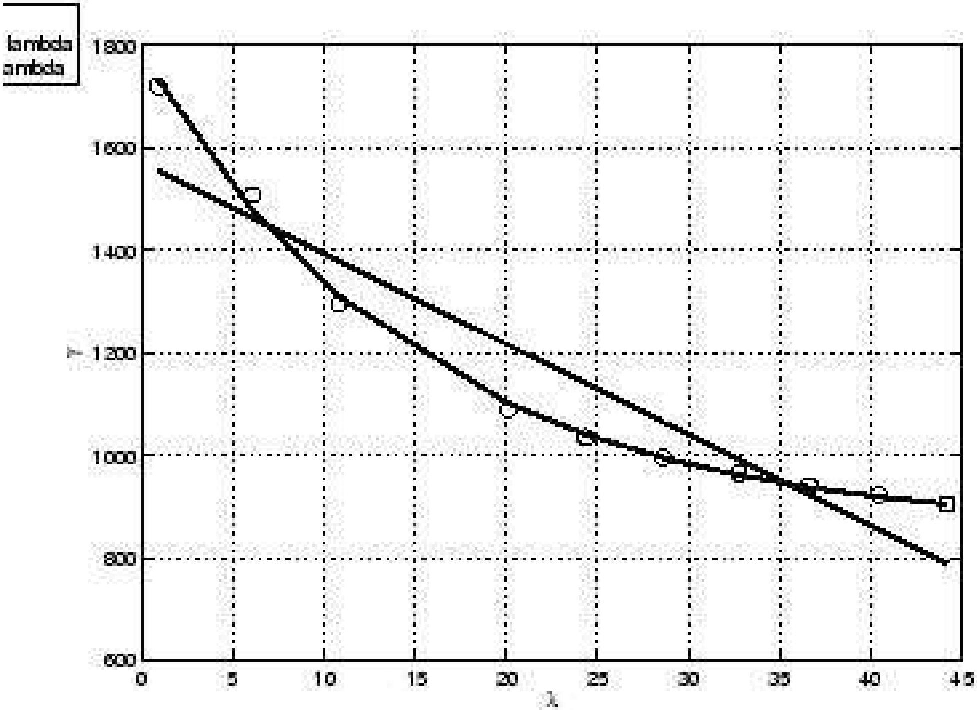
Following identification criterion was assumed

$$Q = \sum_{i=0}^{N} (\sigma(t_i) - \tilde{\sigma}(t_i))^2 \qquad (8)$$

where $\sigma(t_i)$ - measured material response (stress signal),
$\tilde{\sigma} = \tilde{\sigma}(t_i; C, R, X_0, \gamma(\lambda(t_i)))$ - computed material response.

Identification procedures

Function $\gamma(\lambda)$



Identification procedures

Function $\gamma(\lambda)$

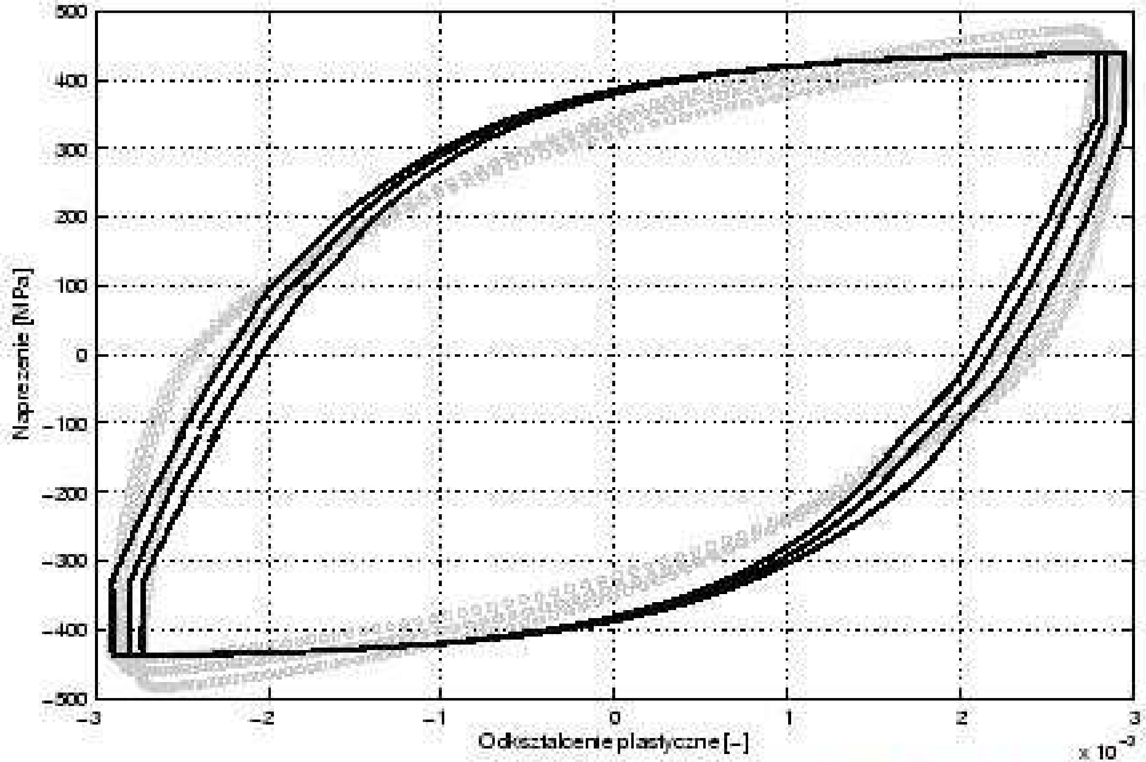
Relaxation function

$$\gamma(\lambda) = e^{a\lambda+b} + c \tag{9}$$

was selected and therefore identification parameters vector takes following form $x_i = [C, R, X_0, a, b, c]$.

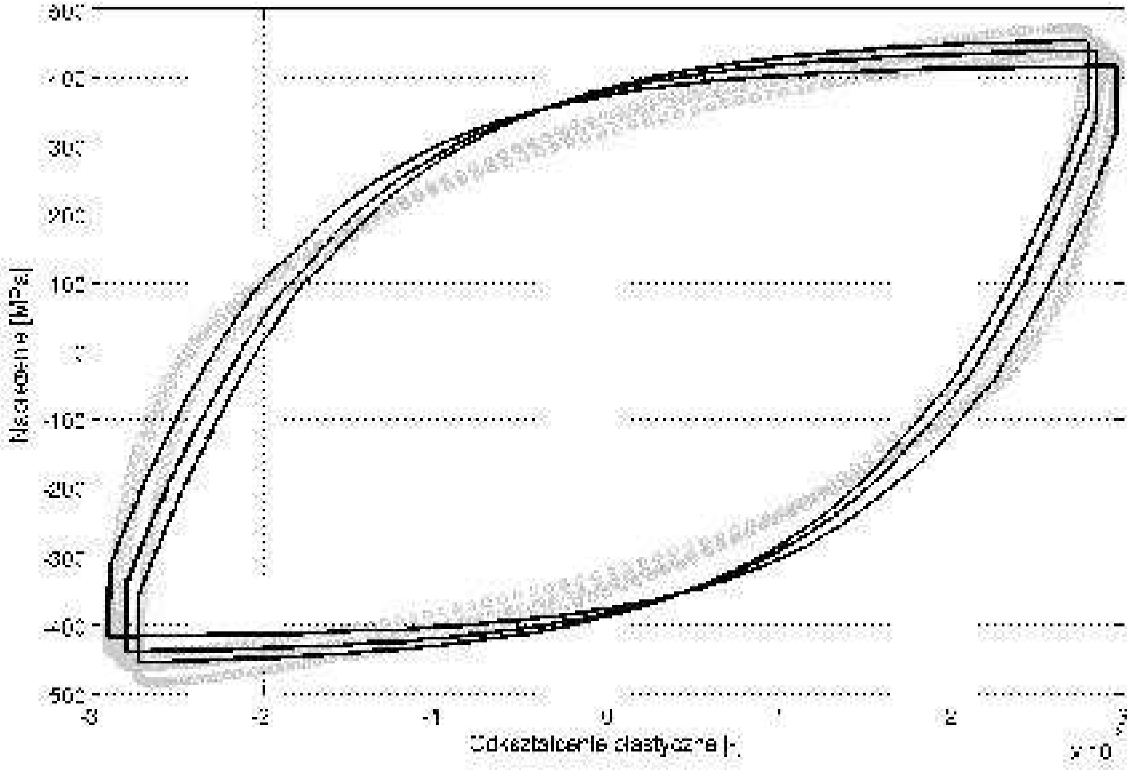
Identification procedures

Typical plots for constant γ



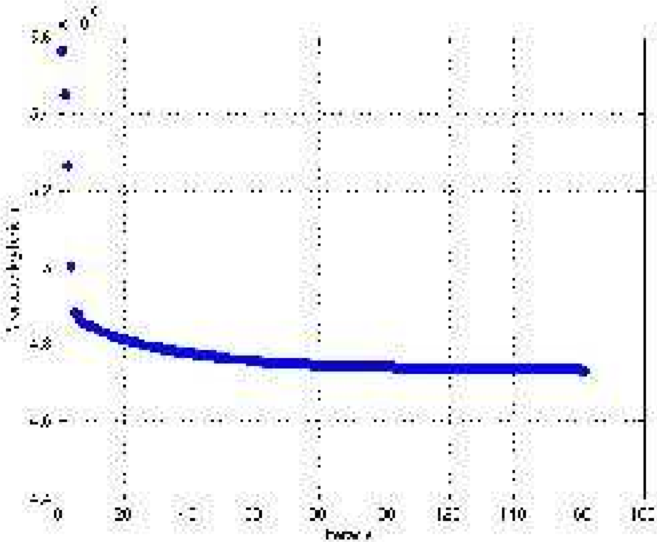
Identification procedures

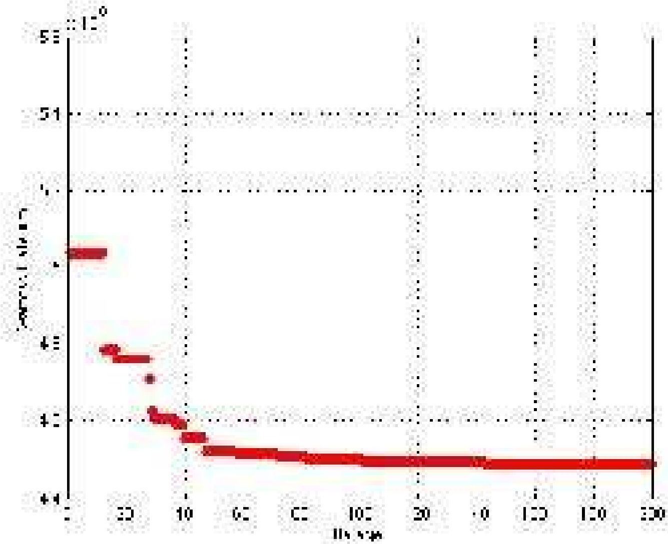
Typical plots for variable γ



Identification procedures

Typical plots of criterion value changes

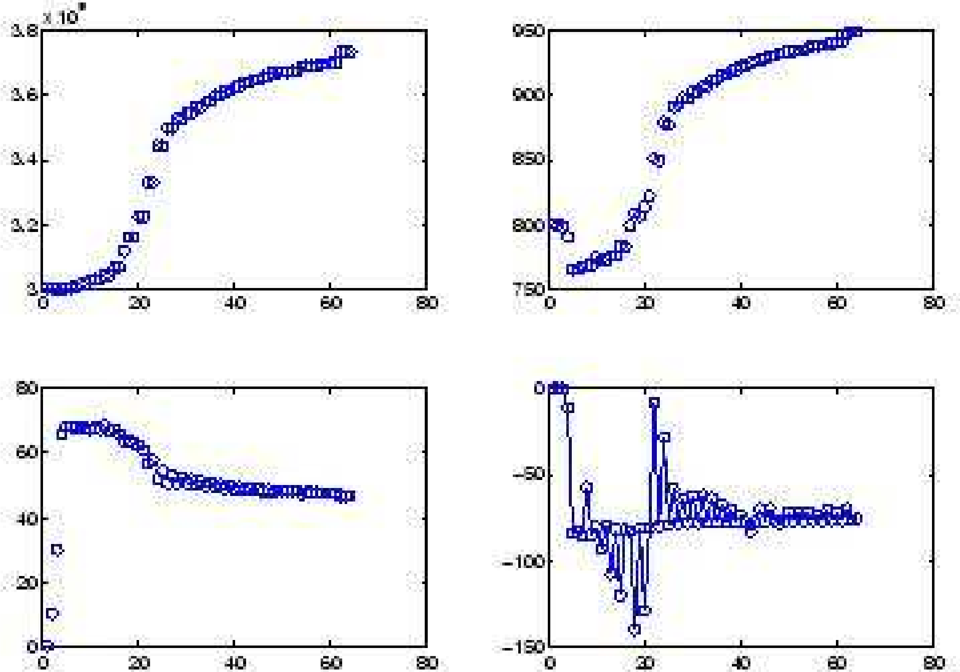




$$x_{CG} = [377079; 49; -324; -0,067; 6,92; 874]$$
$$x_{DE} = [431892; 38; -193; -0.062; 7.02; 971]$$

Identification procedures

Typical plots of identified parameters' values changes



Behind the scenes

Differential evolution algorithm

- One of the world fastest optimisation algorithm.
- Developed by Price & Storn for Chebyschev Polynominal fitting problem in 1995.
- Since that applied in many fields like:
  - digital filter design,
  - neural networks learning,
  - electricity market simulation,
  - wide range of optimisation problems.

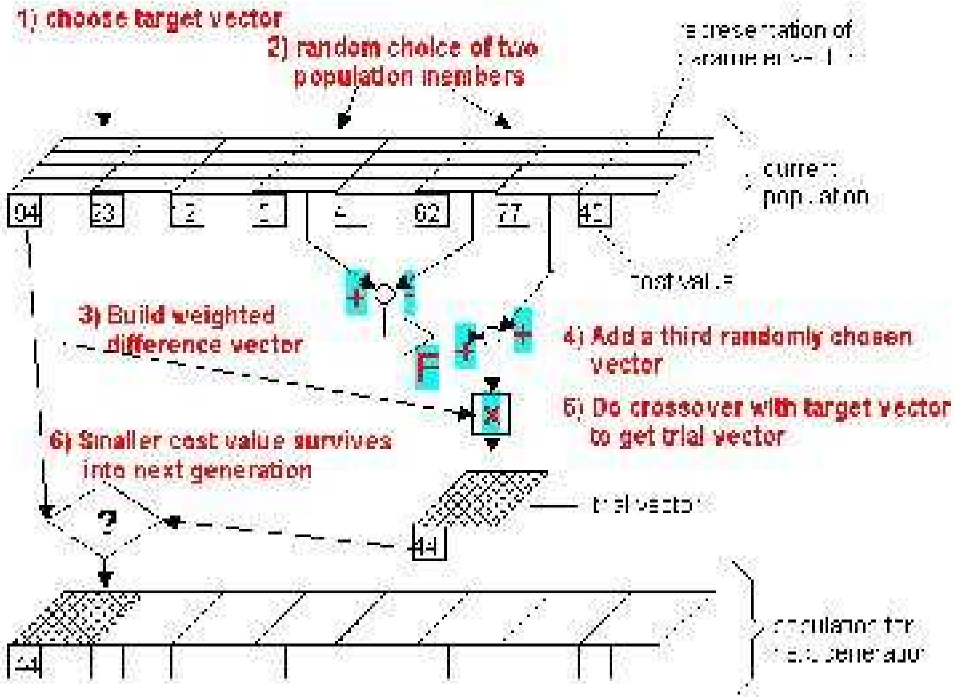
Behind the scenes

Differential evolution algorithm outline

- random choice of initial population of parameter vectors from defined range,
- computation criterion value for each vector,
- cross, according to selected strategy, two vectors and generate third one,
- build second generation of parameter vectors.

Behind the scenes

Differential evolution algorithm outline



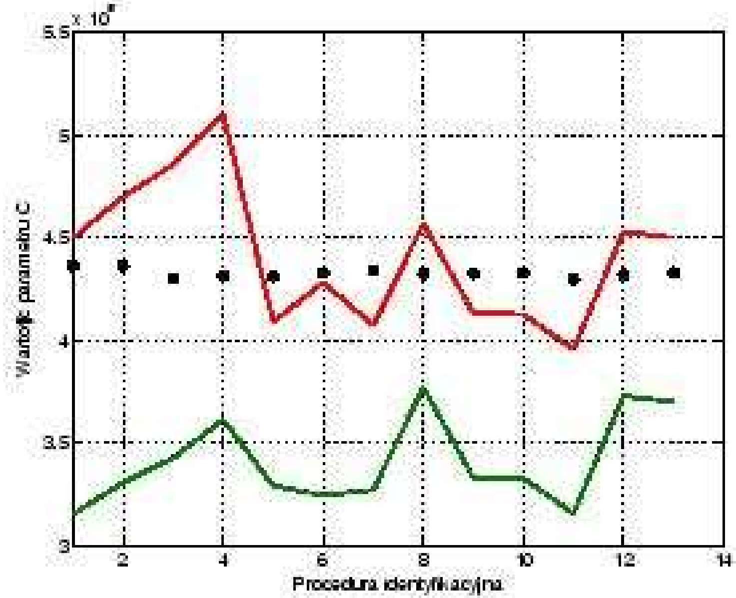
Behind the scenes

Example of cluster application

- 13 nodes were used,
- 13 ranges for initial vector parameters population were randomly chosen,
- 13 values for every parameter were computed in time needed for computation of 1 value on a single computer.

Behind the scenes

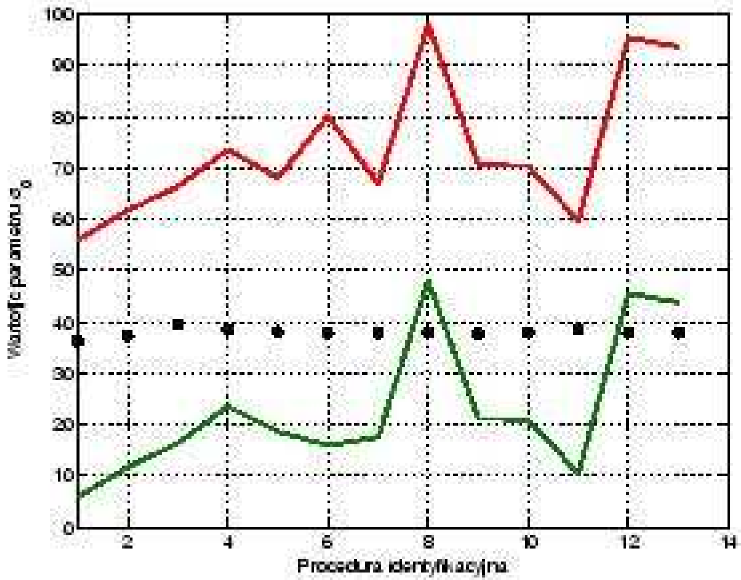
Example of cluster application



Parameter C obtained on cluster nodes and initial vector parameter range

Behind the scenes

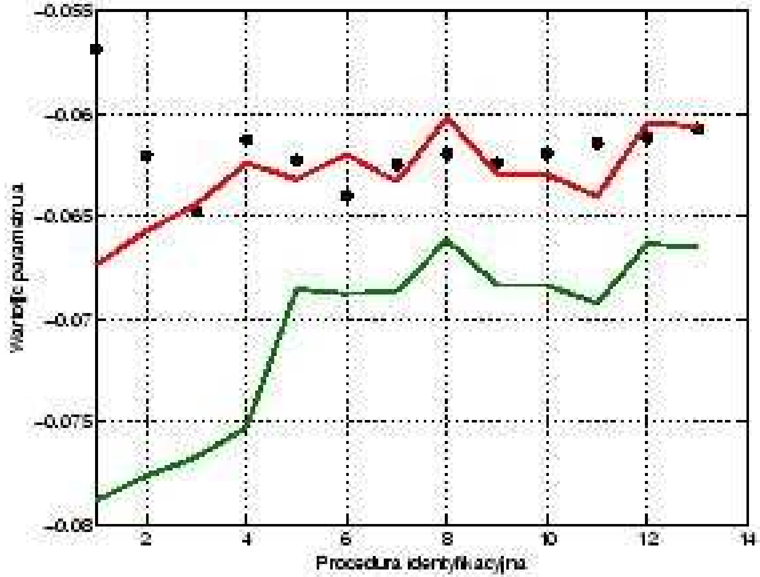
Example of cluster application



Yield point *R* value obtained on the cluster nodes and initial vector parameter range

Behind the scenes

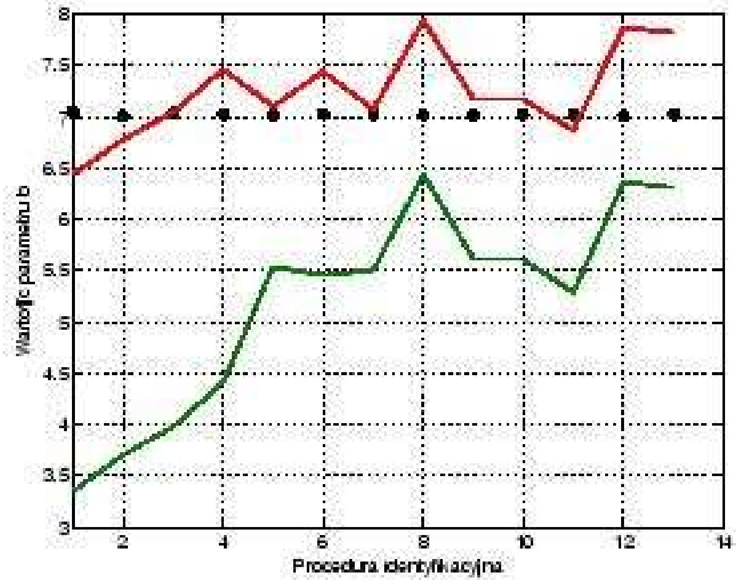
Example of cluster application



Parameter *a* obtained on the cluster nodes and initial vector parameter range

Behind the scenes

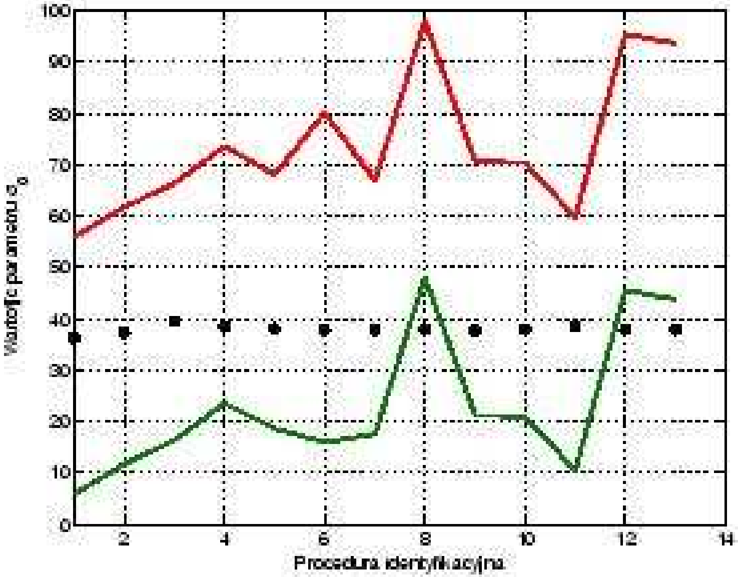
Example of cluster application



Parameter *b* obtained on the cluster nodes and initial vector parameter range

Behind the scenes

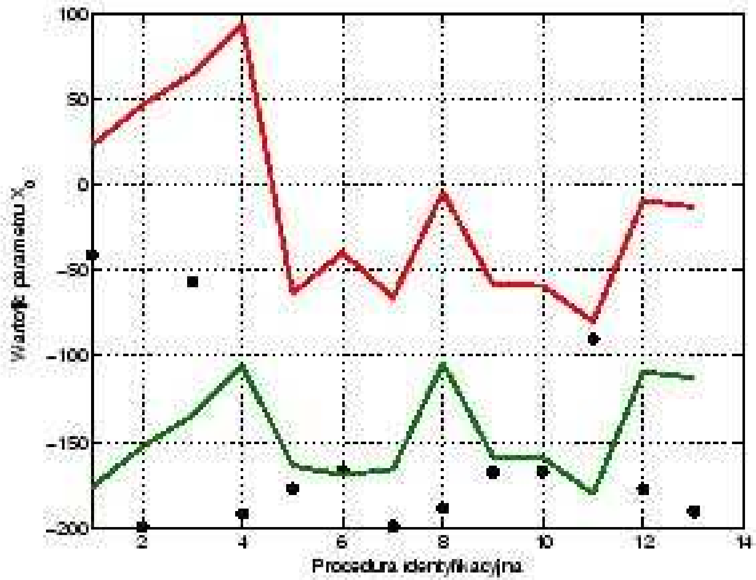
Example of cluster application



Parameter *c* obtained on the cluster nodes and initial vector parameter range

Behind the scenes

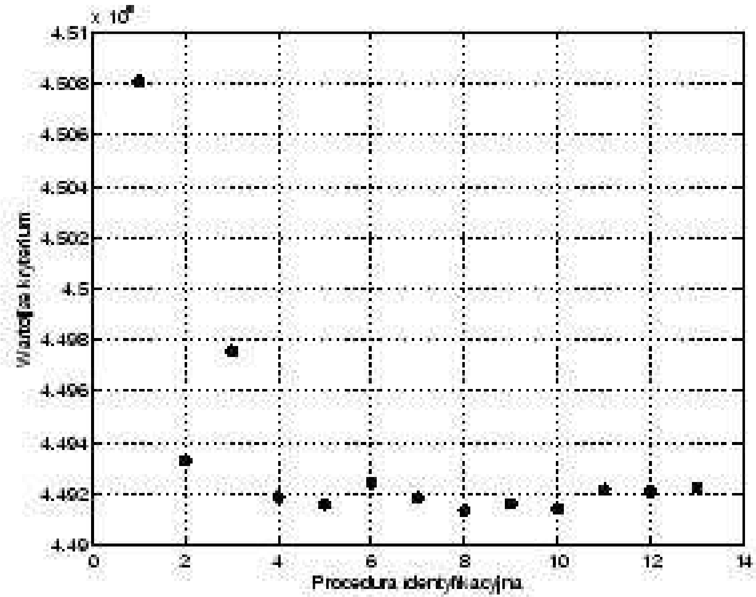
Example of cluster application



Parameter X_0 obtained on the cluster nodes and initial vector parameter range

Behind the scenes

Example of cluster application



Criterion value computed on the cluster nodes

Behind the scenes

Cluster facts

- Built upon computers regularly utilised in the Institute's computer laboratory.
- Consists of virtually the same 17 64-bit AMD Athlon machines (nodes) with 0,5GB RAM installed and over 100GB HDD.
- The nodes run Debian GNU/Linux operating system, Torque cluster software and Maui queuing application.
- Approximate total computing power is c.a. 18GFLOPS.

Summary

- differential evolution algorithm applied to the presented optimisation problem is faster and gives better fits than standard gradient algorithm,
- cluster computing significantly decreases computation time,
- the drawback of the presented approach is large amount of data to deal with,
- the drawback may be partially overcome by keeping clear structure and use of object-oriented programming techniques in optimisation and data manipulation procedures.

Literature

[1] Maciejewski, Ł., Myszka, W. and Ziętek G.: Estymacja parametrów nieliniowego równania konstututywnego przy obciążeniach cyklicznych. W: XXI Sympozjum Zmęczenie i Mechanika Pękania, Bydgoszcz 2006

[2] Storn, R. and Price, K.: Differential Evolution - a Simple and Efficient Adaptive Scheme for Global Optimization over Continuous Spaces. Technical Report TR-95-012, ICSI, March 1995.

Summary

Thank you for your attention!