A R C H I V E S O F M E T A L L U R G Y A N D M A T E R I A L S
Volume 59 2014 Issue 3
DOI: 10.2478/amm-2014-0146

T. DĘBIŃSKI*, M. GŁOWACKI*, M. HOJNY*, A. GUMUŁA*, D. WOŹNIAK*

WEB SYSTEM DEDICATED TO PARALLEL COMPUTATION FOR MODELING OF MUSHY STEEL DEFORMATION

SYSTEM INTERNETOWY POŚWIĘCONY OBLICZENIOM RÓWNOLEGŁYM W MODELOWANIU STREFY PÓŁCIEKŁEJ STALI

The paper presents web base system for an application of parallel object-oriented programming technique in modelling of rolling process of steel plates with semi-solid zone. It also throws light on the problem of semi-solid steels yield stress relationship, one of the main input data of the simulation, and on application of inverse solution, the only possible method of development of the stress-strain curves at extremely high temperatures. Due to limitations of available computer resources a very accurate computation can sometimes be impossible or the time performance can be a barrier for practical application of complex sequential models. Taking advantage of parallel computing the authors have developed an algorithm allowing for fast computation using multiple processors, which is the main subject of the presented paper.

Keywords: parallel computation, semi-solid state, FEM modelling, inverse analysis, solidification

W pracy zaprezentowano system internetowy dla aplikacji wykorzystującej obliczenia równoległe oraz obiektowe techniki programowania do modelowaniu procesu walcowania blach z występowaniem strefy półciekłej. Określenie granicy plastyczności dla stali w stanie półciekłym jest niezwykle trudne, niezbędne dane w postaci krzywych naprężenie-odkształcenie w ekstra wysokich temperaturach można uzyskać jedynie przy zastosowaniu analizy odwrotnej. Ze względu na ograniczenie dostępnych zasobów komputerowych, a także złożoność modeli obliczeniowych, bardzo dokładne analizy mogą być bardzo czasochłonne, a w pewnych przypadkach nawet niemożliwe. Autorzy, wykorzystując obliczenia równoległe, opracowali algorytm pozwalający na szybkie obliczenia przy użyciu wielu procesorów, co jest głównym przedmiotem niniejszej pracy.

1. Introduction

The mechanical properties of steel depend on the temperature of the metal. These data significantly affect the field of temperature, strain and stress, so they should be under consideration during semi-solid material processing.

At very high temperature it is difficult to determine plastic parameters of steels and physical metal investigation needs additional support of numerical methods. Mechanical properties obtained from computer aided semi-solid steel testing can be used as the input data for simulation of rolling of species with semi-solid zone.

Due to limitations concerning computer resources which are available to technologists of metal forming processes a very accurate computation can sometimes be impossible or the computation time can be a barrier for practical application of complex sequential models. Sudden changes of strip temperature and significant temperature dependence of steel mechanical properties are sources of optimization difficulties. There are many phenomena which have to be taken into consideration in the model and which finally affect the computation process. The resulting computer programs require very long processing time and parallelization can be one of the solutions of the problem.

Sequential numerical algorithms using scalar calculations do not allow easy migration to a parallel computer system with shared memory due to their high sensitivity to the data which they work with. An attempt to transfer this type of algorithms to parallel computers needs serious modifications. A key issue that allows for the shortest possible computation time is to minimize the communication between cluster nodes because of the high cost of CPU time and network latency.

The program presented in the current contribution enables the analysis of phenomena accompanying the material deformation during rolling of slabs with mushy zone. The proposed algorithm, which has been implemented in the C++ language, provides such an arrangement of code segments working on each cluster node which guaranties even distribution of workload among all workstations. For control of calculation projects web based system was created.

2. Prediction of mechanical properties in semi-solid state

The computer aided procedure [1-6] leading to stress-strain relationship has been developed at AGH-University of Science and Technology in Krakow Poland. Its experimental part has been carried out in the Institute for Ferrous Metallurgy in Gliwice Poland using the GLEEBLE 3800 simulator

* AGH UNIVERSITY OF SCIENCE AND TECHNOLOGY, FACULTY OF METALS ENGINEERING AND INDUSTRIAL COMPUTER SCIENCE, AL. A. MICKIEWICZA 30, 30-059 KRAKÓW, POLAND

(Fig. 1). The material used for the purposes of the current work was the BW11 grade steel [1,2].

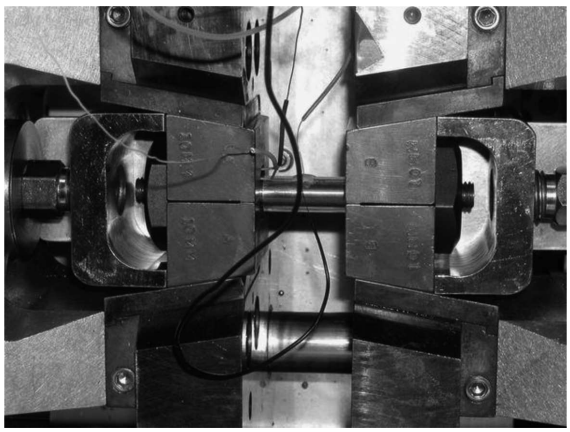

Fig. 1. Equipment before the experiment

The numerical part of the program was written in both Fortran and C++ languages. The main purpose of the program is numerical support for very high temperature testing of mechanical steel properties. It allows for avoiding problems which arise by traditional testing procedures, caused by extremely strong in homogeneity of strain existing in samples subjected to compression at temperatures exceeding 1400°C.

During experiments consisting of sample melting and subsequent deformation some quantities were recorded. The most important ones are: die displacement, force and temperature changes in the heating zone. Simultaneously the computer simulations were running in order to obtain optimal values of the material parameters, first of all coefficients of the strain-stress curve. The liquidus and solidus temperatures of the BW11 grade steel are 1523°C and 1482°C, respectively [2]. The strain-stress curves, which are necessary for the mechanical model, were constructed on basis of a series of experiments [3].

For the temperature range under 1420°C traditional testing method may be applied giving good results. The temperature range was divided into two sub-ranges: lower, below 1420°C and higher, above that temperature. The usual testing procedure was applied for the lower temperature range. The curves were described by Voce'a formula in the temperature range between 1200°C and 1420°C. It is not easy to carry out isothermal experiments for temperatures higher than 1420°C. Several serious experimental problems arise. First of all, keeping such a high temperature constant during the whole experimental procedure is extremely difficult. There are also severe difficulties concerning interpretation of the measurement results. The significant in homogeneity in the strain distribution in the deformation zone and the distortion of the central part of the sample lead to poor accuracy of the stress calculated using traditional methods, which are good for lower temperatures. At temperature 1400°C and lower the lack of the liquid phase has been observed [1] and traditional analysis of experimental results is valid. However, the results of tests conducted at temperatures higher than 1420°C require inverse analysis [2]. Apart from different kind of analysis the curves were described by Voce'a equation in both the cases. The inverse analysis is time consuming. Hence, due to required long calculation time parallel computation is very helpful.

The inverse method is based on comparison between the calculated and measured loads, which for temperature of 1460°C is presented in Fig. 2. The experimental graph shows high level of noise because of very low material strength and the nature of the deformation. Despite this the model is able to compute the real values of the strain-stress curve. The objective function used for the experimental purposes was defined as a norm of discrepancies between calculated (f^c) and measured (f^m) loads in a number of subsequent stages of the compression test (1):

$$\varphi(x) = \sum_{i-1}^{n} \left[f_i^c(x,\, p) - f_i^m \right]^2 \tag{1}$$

The coefficients, which are results of the inverse analysis, allow for drawing stress-strain curves, which are presented in Fig. 3. The figure shows curves for three different temperature values: 1425°C, 1450°C and 1460°C. Very low level of stress can be observed in all the graphs.

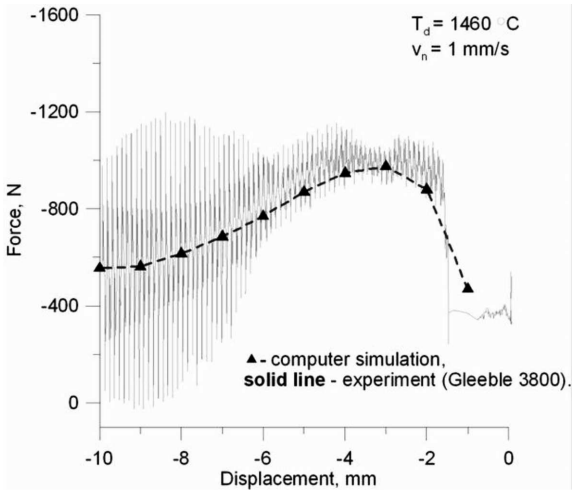

Fig. 2. Comparison between measured and calculated loads at temperature 1460°C for the quasi – static process

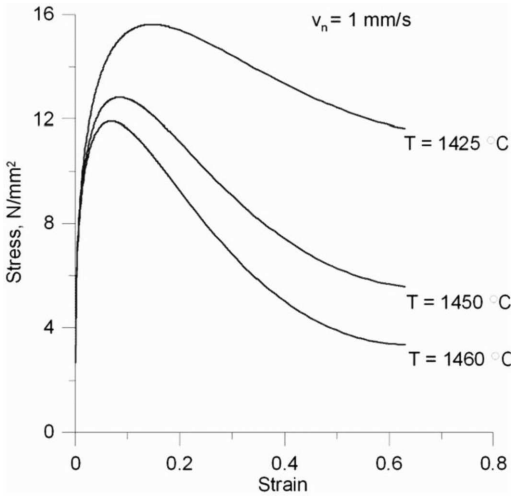

Fig. 3. Flow stress vs. strain at temperature 1425°C, 1450°C and 1460°C

3. Sequential model of rolling - thermal solution

Heat transfer is one of the main phenomena accompanying the hot rolling process, which results in formation of temperature gradients inside the rolling zone and even outside of it. In the presented model, heat flow is considered in a particular area of the specimen. The discretization process leads to selecting a finite number of points inside the body. Certain temperature is attributed to each of the points (nodes). The set of all values of temperature at all given points creates a space and time dependent temperature field T = f(x, y, z, t). Heat transfer models were based on the solution of Fourier-Kirchhoff heat conduction equation [7, 8]. The solution is based on heat flux functional minimization, which includes relevant boundary conditions.

$$\chi = \int_V \left\{ \frac{1}{2}\left[k_x \left(\frac{\partial T}{\partial x}\right)^2 + k_y \left(\frac{\partial T}{\partial y}\right)^2 + k_z \left(\frac{\partial T}{\partial z}\right)^2 \right] - QT \right\} dV + \int_S \left(qT + \frac{1}{2}\alpha (T - T_0)^2 \right) dS \tag{2}$$

In (2) λ_i are anisotropic heat transfer coefficients, Q – heat generation, V – control volume, S –body surface, α – heat transfer coefficient and q – heat generated by friction. In [9] one can find solution of this problem in two steps. The first one is based on the Fourier-Kirchhoff equation for steady flow of heat using the finite element method. The second step is a generalization of the resulting stationary matrix equations obtained for the steady-state process with the help of the Galerkin residual method. In both cases the system of equations to be solved can be represented in matrix form as (3):

$$KT = p \tag{3}$$

where **K** is heat capacity matrix, **T** – parameters vector and p – the right hand vector.

4. Mechanical model

Another model, the most important one for metal forming processes, is the material plastic behaviour model. In the presented approach a three dimensional rigid-plastic model of rolling process has been applied, which in the case of large plastic deformations at very high temperatures can give good results. The power functional resulting from variational formulation of the problem is non-linear in all cases and so the solution requires strong computing power. Application of finite element discretization to analysis of spatial rigid-plastic model involves processing a large number of variational parameters. Hence, the calculation requires long computation time but results in a solution which is consistent with experimental data for both simple and complex deformation zones.

The variational approach of the rolling process requires optimization of a power functional, which in general can be expressed in the form of equation (4):

$$W = W_\sigma + W_\lambda + W_t \tag{4}$$

where W_σ is the power of plastic deformation, W_λ is the incompressibility condition (penalty power) and W_t is the friction power.

The deformation process of steel in semi-solid state depends on material density changes. In this case the condition of incompressibility, which is sufficient for deformation at lower temperatures, has to be replaced with a more general condition of mass conservation [11].

The application of finite element method in case of metal forming processes results in a finite set of velocity values. All of them are parameters of the body deformation field. Spatial discretization allows for optimization of deformation field for a certain time step. At the same time discretization of time is necessary to perform series of time steps under the assumption of constant strain field in each step. The incremental solution of the problem has to be constructed iteratively. Each iteration involves solving linearized system which can be expressed as a matrix equation (5):

$$Kv = f \tag{5}$$

where K is the structure stiffness matrix, v – the nodal velocity vector and f – the right hand vector. The process is divided into time steps. In each step Ąt the final shape of the body is calculated from its initial shape using the velocity field resulting from optimization of the functional given by (4).

5. The parallel algorithm for a cluster of workstations

The main idea of parallel computation is a decomposition of more complex issues into smaller, simpler sub problems. Such decomposition encounters numerous obstacles concerning the criteria and method of task division. There is no universal silver bullet for decomposition method. As a consequence individual analysis and selection of the optimal method [12] is required for any particular case. The finite element parallel solution requires a uniform domain decomposition between the cluster nodes. In case of model with distributed memory an appropriate division of tasks is required. The division should minimize the communication overhead, therefore geometric decomposition of the finite element mesh was chosen.

Decomposition of the problem imposes both the way of storing the equation system and the method of exchanging data between processors. The equation set for thermal and mechanical model has to be stored in sub-matrices in a form reflecting the selected geometric distribution. Each processor stores the allocated part of matrix and right hand vector. Communication between cluster nodes with neighbouring sub domains is essential for maintaining consistency of data common to the machines.

It is necessary to store data belonging to the neighbouring subdivisions. This problem has been solved by the use of temporary data vectors. The same rules may be applied to both heat capacity and stiffness matrices. The stiffness matrix has larger number of variables and so is able to store more data. Principles of data exchange and completion of local matrices are the same, the only difference is the amount of data.

6. Preconditioning

The iterative methods for solving systems of linear equations are theoretically effective and efficient, characterized by poor flexibility compared to direct methods. In the case of

solving problems in the field of fluid dynamics, simulation of electronic circuits and simulation of metal forming processes are characterized by slow convergence.

These disadvantages make it difficult to use iterative methods in industrial applications. In order to increase the efficiency and flexibility of Krylov subspace methods, techniques to improve the initial conditions called preconditioning are commonly used. Special methods to improve convergence, transforms original linear system to another having the same solution, but more useful for solving iterative methods. Reliability iterative techniques depends on the quality of the preconditioner.

In linear algebra and numerical analysis matrix to improve conditions for the initial matrix A is a matrix such as $M^{-1}A$ characterized by a smaller number of degrees of freedom. Such arrays are commonly used in iterative methods for solving large systems of linear equations with sparse matrices [15] of the type (6):

$$A \cdot x = b \tag{6}$$

Convergence of solutions for a large number of iterative solvers for linear systems of equations decreases with increasing conditions of **A**. Therefore, the above equations can be replaced by system (7):

$$M^{-1}A \cdot x = M^{-1}b \text{ or } AM^{-1}u = b,\ x = M^{-1}u \tag{7}$$

From a practical point of view, the solution of set of equations with preconditioning should be characterized by a small computational effort, because the algorithms with improved initial conditions based on the calculations with the matrix **M** in each iteration.

The main task of preconditioning is reduction of conditions of the matrix. Very important for speed of calculations is good choice of preconditioner matrix. The best preconditioner for the matrix inversion is (8):

$$M = I \text{ or } M^{-1} = I \tag{8}$$

where:I – identity matrix.

Unfortunately, this type of matrix does not speed up the solution. The second case is a system (9):

$$M = A \text{ or } M^{-1}A = AM^{-1} = I \tag{9}$$

Where the number of conditions is 1, and the solution can be obtained after a single iteration. But for this approach (10):

$$M^{-1} = A^{-1} \tag{10}$$

calculating of the inverse matrix for preconditioner is just as difficult and time consuming as in the original matrix A.

Given the above, the applied matrix M must be between two extremes – solutions for the minimum number of iterations and minimal computations for determining the matrix $\mathbf{M^{-1}}$.

In this solution a block version of the matrix to improve conditions initial Jacobi was used. Block version considered the division into sub-tasks. If the S={1, ...,n}will be divided into sub $S=U_iS_i$ where the sets S_i are mutually exclusive, the block version of the Jacobi matrix can be written (11):

$$m_{i,j} = \begin{cases} a_{i,j} & \text{for the same subset} \\ 0 & \text{otherwise} \end{cases}$$

$$m_{i,j} = \begin{cases} \hat{m}_{ii} & 0 \\ 0 & \hat{m}_{jj} \end{cases} \tag{11}$$

Where $\hat{m}_{ii}$ and $\hat{m}_{jj}$ are diagonal matrixes.

Most of decompositions is performed in cases of:

- problems with many variables per node, blocks of data are created by grouping the unknowns at the nodes,
- structural matrix where the division of tasks corresponding physical areas, for example, the division can be done along the line in case of problems or 2D planes for 3D problems,
- for parallel computers natural way to decompose the task is to divide the variables between the available processors.

7. Productivity analysis of modeling the steel rolling process in the semi-solid state

The developed parallel algorithm was tested on a cluster of two nodes. Each node had two Dual-Core AMD Opteron 2.2 GHz processors, 2048 MB of memory and disk matrix RAID 0 RAID CORE HT 1000, 2×400 GB and the operating system OPEN SUSE 10.2. Communication between cluster nodes was based on MPICH message-passing environment [13] over a 1 Gb/s Ethernet Network.

In order to determine efficiency and speedup of calculations several tests have been conducted. The acceleration quality was calculated using the following indicators(12,13)

- relative speedup

$$S_p = \frac{t_1}{t_p} \tag{12}$$

where t_1 is the execution time on one processor and t_p is the execution time on p processors,

- efficiency

$$E_p = \frac{S_p}{p} \tag{13}$$

where $S_p$is the speedup and p is the number of processors.

The main part of computational tests was performed for parallel iterative methods for solving set of equations, CG and MINRES [14]. Program efficiency analysis was performed as well as tests of the solvers for 40,000 data nodes for two categories of parallel computing: shared memory and distributed memory models. Although many basic operations (object creation, memory allocation, mesh generation, solving the set of equations, results collection) could not be performed in parallel due to data dependence, conducted tests show the computational efficiency of the solution. In order to verify the performance of networks for computational process tests were conducted for both the mentioned memory models. The results of the tests are presented in pictures 4, 5 and 6.

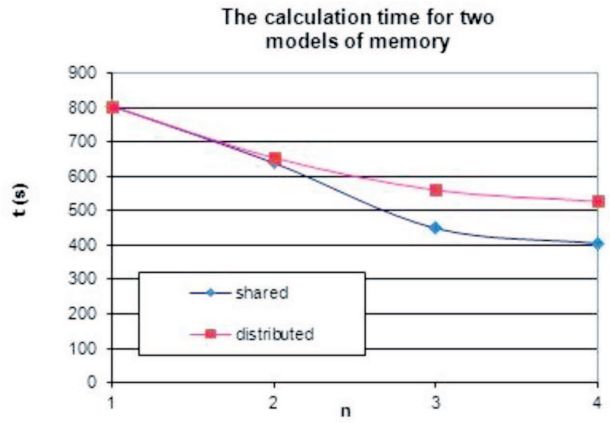

Fig. 4. Time profile for shared and distributed memory model

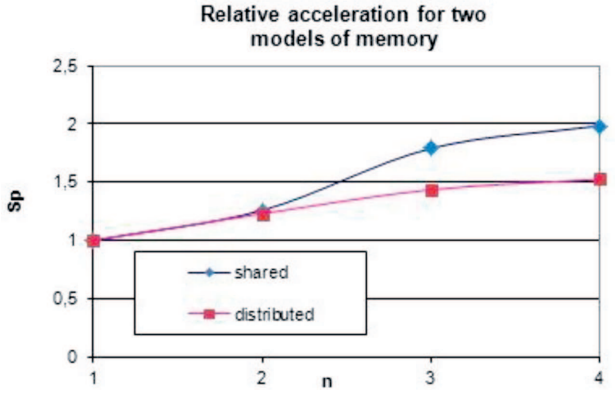

Fig. 5. Acceleration profile for shared and distributed memory model

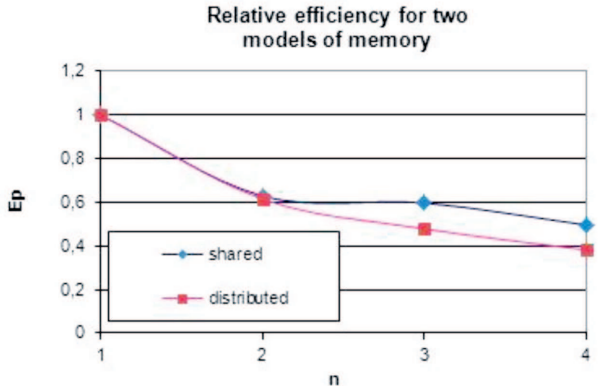

Fig. 6. Efficiency profile for shared and distributed memory model

8. Web system

Because system was located on server the user must have access and control in a simple and accessible way. Commonly used technique for this is an interface based on Web technology [16], [17], [18], Also in this work this type of solution was used. This technology allows to modify the input data and perform remote calculations using any computer with access to network. Because it is a multi access system, users verification is required. This project was implemented in PHP [19, 20, 21] to allow execution of scripts on the server side. The user fills in the login form, the data is sent to the server for verification in a MySQL database [22, 23, 24] using the SQL query language.After successful verification, full functionality is available. The system operates on separate projects, where the user can store all the input process parameters and simulation results. All process data are stored in a central database, so it allow to safe and fast access to the stored information.Setting the parameters of the process, based on the forms updated from the database. This solution allows for quick and clear way to enter and correct data process. The first step is to create a new project. Entering of all data is in a few steps (Fig. 7):

- parameters of rolled slab
- parameters of rollers
- material properties

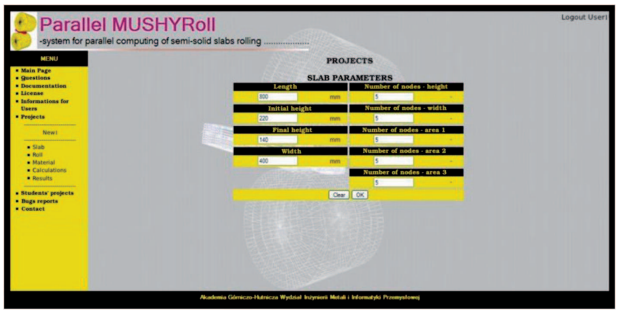

Fig. 7. Window for parameters of rolled slab

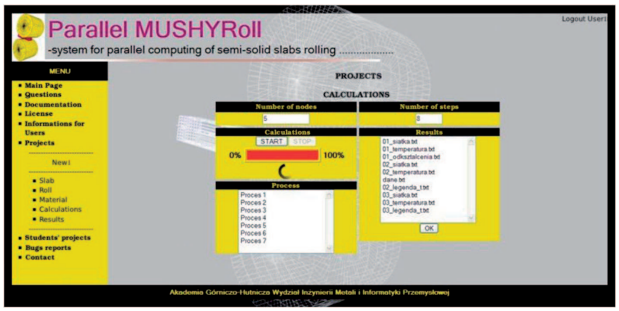

Fig. 8. Window for final process parameters and calculations

Next, user is redirected to the final process parameters and can start the calculations (Fig. 8). Finally, after the correct calculations, postprocessor window allows to read and display the results. There are areas for visualization of the results:

- The area of graphic visualization of object in VRML format
- The area of the files with the results

Three-dimensional visualization window allows to an interactive analyze of the results by operating on model in 3D space (Fig. 9).

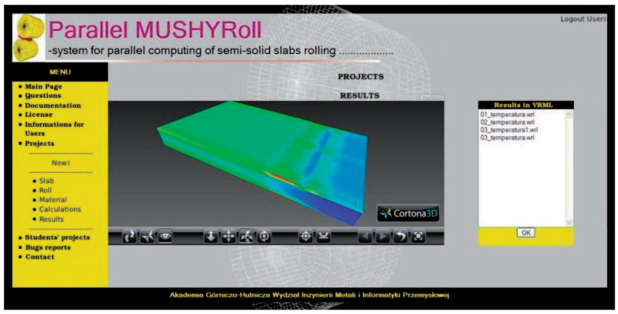

Fig. 9. Postprocessor window

9. Conclusions

Parallelization of iterative methods for solving systems of equations was performed for the following operations: construction of a parallel matrix, the matrix-vector product, scalar product of vectors, vector's norm, preconditioning and implementation. Because all these operations work on independent parts of the matrix and vectors their parallelization is very effective. Finally, a parallel version of the software simulating the rolling process of steel in semi-solid state was built. For improve convergence of set of equations preconditioning techniques was used. Calculation was done for four nodes and two models of parallel computing techniques. The acquired results allow to conclude that the proposed solution is appropriate. Speedup for four nodes was about 2x. For better improvement of efficiency modification of communication is needed or using other matrix or method for preconditioning.

Web-based system allows to control input data and calculations. Write parameters to the database allows to build a library of materials for later use. Interactive three-dimensional visualization of the results, facilitates the analysis of the results.

Acknowledgements

The project has been supported by the Polish National Science Centre. Decision number: DEC-2011/03/D/ST8/04041.

REFERENCES

[1] M. H o j n y, M. G ł o w a c k i, Computer modelling of deformation of steel samples with mushy zone, Steel Research Int. 868-874 (2008).
[2] M. G ł o w a c k i, M. H o j n y, Inverse analysis applied for determination of strain-stress curves for steel deformed in semi-solid state, Inverse Problems in Science and Engineering, 159-174 (2009).
[3] M. H o j n y, M. G ł o w a c k i, The physical and computer modelling of plastic deformation of low carbon steel in semi-solid state, Journal of Engineering Materials and Technology, (2009).
[4] M. H o j n y, M. G ł o w a c k i, The methodology of strain – stress curves determination for steel in semi-solid state, Archives of Metallurgy and Materials, 475-483 (2009).
[5] M. H o j n y, Thermo-mechanical model of a TIG welding process for the aircraft industry, Archive of Metallurgy nad Materials **58**, 4, 1125-1130 (2013).
[6] M. H o j n y, M. G ł o w a c k i, Modelling of Strain-Stress Relationship for Carbon Steel Deformed at Temperature Exceeding Hot Rolling Range, Journal of Engineering Materials and Technology, (2011).
[7] J. T a l e r, P. D u d a, Direct and inverse heat transfer problems, WNT, Warszawa, (2003) – in Polish.
[8] E. K o s t k o w s k i, Heat flow, Silesian Technical University , Gliwice, (2000).
[9] M. G ł o w a c k i, Thermo-mechanical and microstructural model of shape rolling, Dissertations and Monographs, Kraków, 1998 – in Polish.
[10] M. G ł o w a c k i, The mathematical modelling of thermo-mechanical processing of steel during multi-pass shape rolling, J. Mater. Proc. Technol. 336-343 (2005).
[11] M. G ł o w a c k i, Computer simulation of rolling of semi-solid steel slabs, Proc. 15th International symposium on Plasticity & its current applications, St. Thomas, U. S. Virgin Islands, January 10-12 (2009).
[12] B. B o r o w i k, Parallel programming in applications, MIKOM, Warszawa (2001) – in Polish.
[13] MPICH2 home page, http://www-unix.mcs.anl.gov/mpi/mpich/
[14] Y. S a a d, Iterative Methods for Sparse Linear Systems, (2000).
[15] M. B e n z i, Preconditioning Techniques For Large Linear Systems: A Survey, Journal of Computational Physics 182, 418-477 (2002).
[16] HTML and JavaScript for Visual Learners, Visibooks, Computer books for Visual Learners.
[17] B. P f a f f e n b e r g e n, M. S c h a f e r, Ch. W h i t e, B. K a r o w, HTML, XHTML, and CSS Bible, Canada 2004.
[18] R. F u l l e r, A. U r l i c h, HTML in 10 Simple Steps or Less, Canada 2004.
[19] J. M e l o n i, PHP, MySQL and Apache, 2003.
[20] M. R u t l e d g e, PHP Game Programming, 2004.
[21] J. C a s t a n g n e t t o, H. R a w a t, Ch. S c o l l o, D. V e l i a t h, Professional PHP Programming, 1999.
[22] T. C o n v e r s e, J. P a r k, C. M o r g a n, PHP5 and MySQL Bible, 2004.
[23] M. D e l i s e, Creating your MySQL Database: Practical Design Tips and Techniques, 2006.
[24] W. G i l m o r e, Beginning PHP and MySQL, 2008.

Received: 10 January 2014.