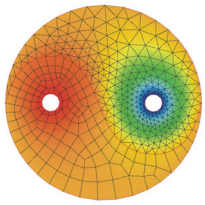
Publishing House
AKAPIT

COMPUTER METHODS IN MATERIALS SCIENCE

Informatyka w Technologii Materiałów

Vol. 14, 2014, No. 1



COMPUTER SYSTEM DEDICATED TO OPTIMIZATION OF PRODUCTION PROCESSES AND CYCLES IN METAL FORMING INDUSTRY

ŁUKASZ RAUCH*, MATEUSZ SKIBA, JAN KUSIAK

AGH – Akademia Górniczo-Hutnicza, Mickiewicza 30, 30-059 Kraków, Poland
**Corresponding author: lrauch@agh.edu.pl*

Abstract

The paper presents design and implementation of the computer system dedicated to optimization of production processes and cycles. The system is proposed as a software responsible for flexible integration of various external computer programs for numerical simulations, libraries of optimization methods, sensitivity analysis, metamodels and material database. The subsequent chapters describe the main idea of the system, its implementation details and the case studies of real industrial processes or cycles. The examples of obtained results are presented, as well.

Key words: computer system, optimization, production processes, production cycles, numerical simulations, metamodelling

1. INTRODUCTION

Modern design of production cycles in metal forming industry is based on sophisticated numerical procedures integrated with advanced optimization methods. The software used for numerical simulations is delivered in a form of different computer systems, e.g. Abaqus, Forge, Adina, or in a form of in-house codes usually implemented by different academic and research institutions. Each of existing systems, dedicated to simulation of various production processes, uses different formats of input and output files, different operating systems and requires different resources. Therefore, integration of these systems is in the most cases very difficult or even impossible without additionally implemented communication interfaces. Subsequent problem occurs at the moment of application of optimization methods. A lot of open source and freeware solutions exist, which are implemented by using the most popular programming languages like C, C++, C#, Java. Nevertheless, introduction of such libraries into practice requires automatic execution of objective function, which includes performance of external computer systems for numerical simulations. This task is one of the most difficult issues, which require automatic preparation of input data file and postprocessing of output files, while many of external computer programs are placed on remote hosts with specific protocols of authorization and files exchange. All these problems cause that only few software solutions possess the functionality allowing integration both external computer programs and optimization libraries. Currently, the most universal software is iSight supplied by Simulia and dedicated to integrate Abaqus, Matlab, MS Office and many other computer programs (Miyata et al., 2003; Wang et al., 2011). It is also the most advanced solutions equipped with Design of Experiment (DoE) procedures. Other available software is focused only on optimization of selected industrial processes like rolling, stamping, extrusion or forging. This paper presents solution of presented problems in a form of specialized computer system integrating various

ISSN 1641-8581

external programs, optimization library, metamodels and material database. The second chapter of the paper presents the main idea of the system, while the third one shows design and some implementation details. The foregoing chapter discusses application issues in the form of the use cases based on real industrial processes and cycles.

2. THE MAIN IDEA OF THE SYSTEM

The proposed system aims at support of production technology design through optimization methods, which execute computing procedures based on numerical simulations or metamodels. The most sophisticated part of the system is the functionality of the production cycles design, which are usually composed of two or more processes. Each process can be simulated by different computer programs, which finally have to be connected and used as a single process. Therefore, the crucial objective was implementation of mechanism offering functionality of automatic join and execution of such programs. This functionality allows to close numerical simulations in optimization loop and use such procedures to design complex production cycles or to apply sensitivity analysis to determine parameters of the process having the highest influence on a final product. Moreover, various metamodels based on Artificial Neural Network (ANN) can be used instead of simulation tools to avoid very time consuming multi-iterative optimization procedures.

2.1. Design of production cycles

To facilitate creation and parameterization of production cycles through the Graphical User Interface (GUI), the system gathers information about single processes, which can be flexibly joined together by using specially prepared converters. The main idea of the cycle design is presented in figure 1 (in the presented case Abaqus and in-house software are used in rods production optimization).

Presented converters are computer programs responsible for management of input/output data, used by metamodels and external software for numerical simulations. The converters, covering functionality of mentioned above mechanism of automatic performance of complex calculations, can be divided into four groups:

– Input converters – programs responsible for preparation of input data. System is equipped with built-in module for preprocessing of specially prepared text files, which are used as input data for external simulation tools. The files contain names of variables in curly brackets, which before execution of external programs are replaced with specific values. The parsing of input files allows not only to submit variable values for simulation, but also to change these values automatically for optimization and sensitivity analysis purposes. The values are submitted to simulation tools, as well as to metamodels, in the same way. The main difference lays in execution of both programs. Metamodels are executed by built-in functional module, while the simulation tools run as separated threads.

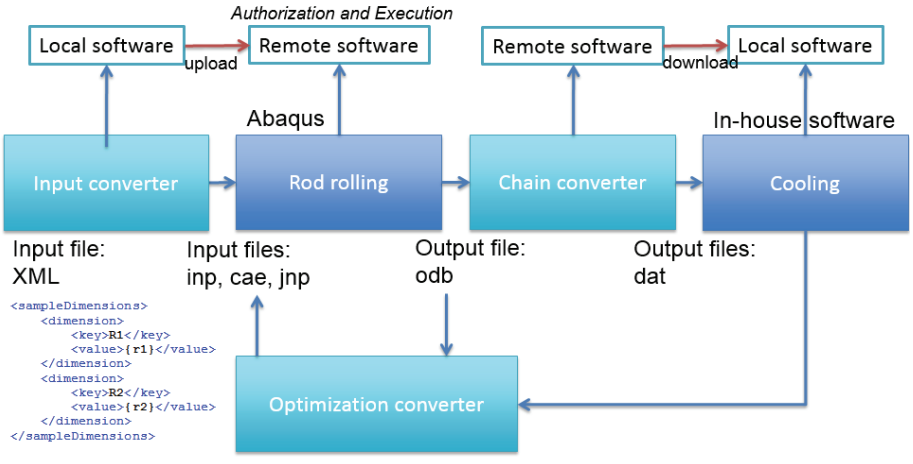


Fig. 1. *Idea of the setup and optimization of production processes and cycles.*

– Output converters – the results obtained from simulation tools and metamodels are often saved in specific binary files, which are very difficult in the analysis, while users of the system expect simple readable formats. The output converters

analyse difficult result files and transform the most important values to a text or images, which can be easily interpreted by users.

- Chain converters – the basic idea of the industrial cycle design is based on connection of single processes into a more sophisticated chain. The connection between two processes is possible only in the case, when output files from the foregoing process can be converted into input files of the subsequent process. The availability of a chain converter responsible for this functionality unblocks the possibility of a connection of two processes.
- Optimization converters – very similar functionality as in the case of chain converters is required for optimization procedure. The results from the last process have to be analysed and passed to selected optimization method, which calculates objective function and chooses new values for numerical simulations. Then, the values are submitted to built-in input converted responsible for start of new simulations of the whole industrial cycle.

The external simulation tools are divided into local and remote software (figure 1). Local software is deployed on the same computer as the proposed system. The *Process* component is used for execution of local externals in the separated thread. The remote software needs more sophisticated procedure, which requires authorization through SSH protocol and upload of input files by using SFTP server. The system uses login and password stored for this purposes in the database. Uploaded input files are passed to remote externals, which are further executed and closed. The obtained results are saved on the remote machine, where the output, chain or optimization converters are placed as well. One of the converters is executed, allowing to download obtained files. Then, the files are used as input data for subsequent simulation tools or metamodels.

2.2. Optimization

The functionality of optimization is covered by numerical modules, which can be directly imported into the system in a form of dynamically linked libraries (*dll* files). The main requirement is that each library has to possess the class called *Main,* which supports listing of available optimization methods and their parameters through the *listing* method. Such mechanism is fully supported by reflection methodology allowing to keep heterogeneity of libraries implementation and flexibility of their import to the system.

Beside described functionality the system is equipped with built-in library, implementing conventional optimization methods as well as nature inspired algorithms. The first group of methods contains e.g. Nelder-Mead and Hooke-Jeeves algorithms, while the latter group is represented by Evolutionary Algorithm (EA) and Particle Swarm Optimization (PSO) procedures. Additionally, the library is equipped with the strategy of the optimization, which allows to select the best method and configure its parameters depending on the analysed problem. The detailed description of this strategy is presented by Kuś and Mucha (2014).

3. COMPUTER SYSTEM DESIGN

3.1. System deployment

The system is implemented in the Client-Server architecture with so called thin Client responsible mainly for the management of the two databases. Internal database gathers information about data used for the purposes of production cycles design, external simulation tools, metamodels, input files, parameters of processes and input files. External database is dedicated to storing the material data, e.g. chemical composition, properties, models and their parameters. The system integrates local and remote simulation tools, where the latter group is deployed on distributed computing servers including e-infrastructures equipped with authorization modules, monitoring software and the third party computer applications. The main deployment diagram is presented in figure 2.

The assumed deployment realizes flexible development of the system by implemented mechanism of adding new external simulation tools. These tools are added to internal database as subsequent records describing IP address of the server, user name, password, destination folder and command line allowing direct execution of the program. The address and authorization data allow automatic upload and download of input files, which create the main data flow between personal computer and remote servers.

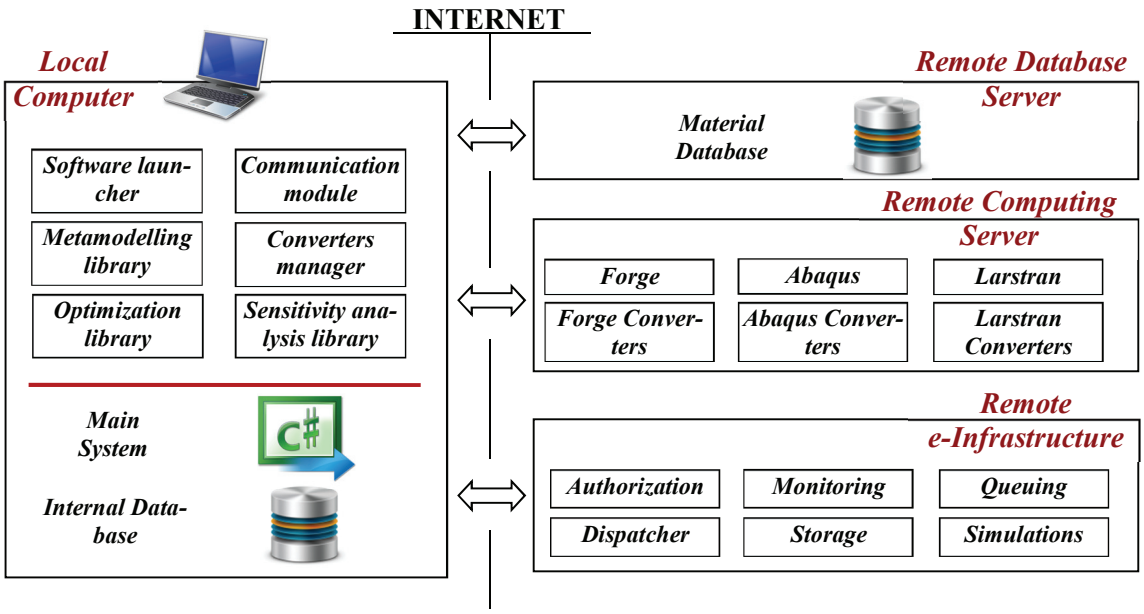


Fig. 2. *The main deployment diagram of the system.*

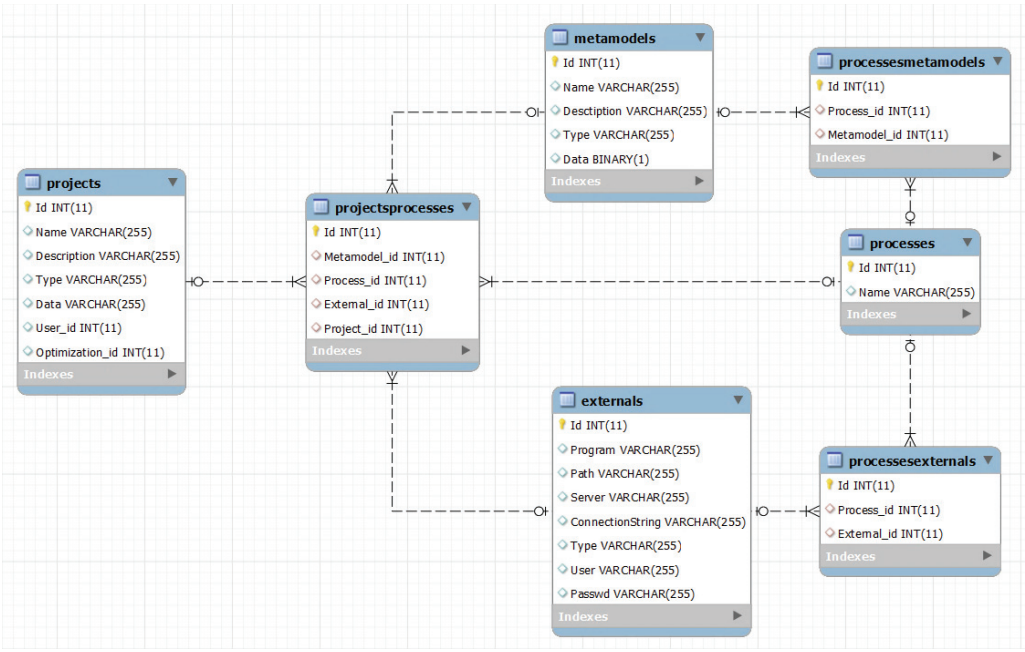


Fig. 3. *Group of tables joining projects, processes, metamodels and simulation tools (externals).*

3.2. Internal Database

The database used by the system to store information, which is required during design of processes and cycles, is one of the most important functional modules. It is composed of more than thirty tables connected to each other with relations forming compact data storage. The most important tables, their fields and relations between them are described in this chapter. Other tables are dedicated mainly for administrative purposes and do not influence the functionality of production cycles design e.g. *Users*, *UsersPermissions*.

Projects table is the crucial point of the database. This table is used to gather information about users' projects, which are related to various approaches to industrial processes and cycles design. The table joins together almost all other data about applied simulation tools, metamodels, materials and industrial processes with user accounts containing authorization information for the system access. Figure 3 presents part of the database diagram with tables responsible for mentioned functionality. Table *Metamodels* contains only some fields describing name and type of metamodel, while the metamodels themselves are stored in *Data* field in the text format

readable by the internal system module for metamodels management and execution. Table *Externals* stores mainly communication data used to login to the remote server, where simulation tools are installed e.g. server address connection string, internal path and type of the external (*remote* or *local* in the case of programs located on the same computer as the system). Authorization data is usually not required for local external, thus, the proper field is left empty in such case.

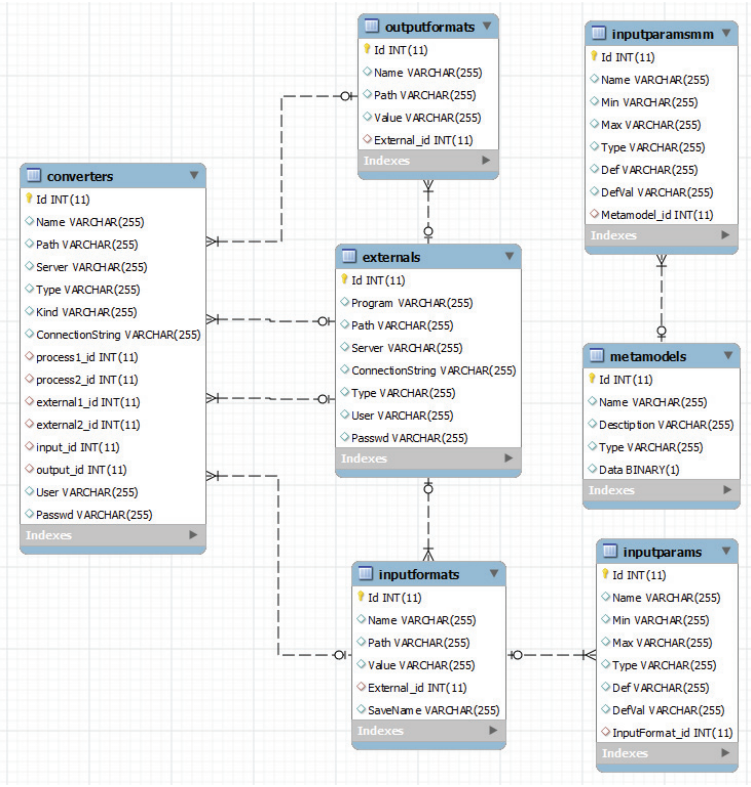


Fig. 4. *Group of tables joining simulation tools (externals) with their input/output formats and parameters, as well as metamodels and their input parameters.*

Each external is executed by input converter, which, as mentioned in chapter 2.1, is responsible also for input data preparation. During configuration of an external in the system, information about the input file is saved in two tables: *inputformats* and *inputparams* (figure 4). The former table keeps data describing name of the file and its path in the operating system, while the latter table is used to store optimization variables, which are retrieved from the input files by parsing. Besides their names, the optimization variables require also information about possible range with minimum and maximum values, default value and type i.e. integer, float, bool or enumerated. All these data is sufficient to apply automated procedure of simulation tools and metamodels execution with different input values, which is crucial for a reliable performance of optimization and sensitivity analysis. Output files are stored in database in *outputformats* table. The main difference in comparison to input files is that results of simulations are often saved in the form of binary files, which are difficult to analyse in unified way. Therefore, the database keeps only information about location and names of such files, while the output/optimization/chain converters are responsible for obtaining specific values, which are passed further as a list of parameters. The different situation occurs in the case of metamodels, where input as well as output parameters are analysed in the system by dedicated built-in module. Thus, converters are kept only in the table, which joins together externals with their input and output formats.

The database is implemented in MySQL technology, which assures flexible usage of this server by different implementation technologies, while the majority of them are equipped with special communication plugins supporting connection and data management.

3.3. Graphical User Interface

GUI is implemented by using .NET framework 4.5 on the basis of Windows Presentation Foundation (WPF) and C# language, while the particular views (graphical windows forms) are created in Extensible Application Markup Language (XAML). The main project contains more than thirty views, which guarantee rich application interface and twenty models being used to access all tables in the databases. Additionally, four more projects are developed within the same software solution, dedicated to optimization, sensitivity analysis, remote access and material database management. The main window of the system is presented in figure 5. It is divided into ribbon menu and the following panels:

– General information – supports creation of production cycles composed of separate production processes. Checkboxes allow to mark optimization checkpoints, while green arrow is used to start validation of the cycle with assumed values of processes and optimization parameters.
– Parameters – this panel lists all the parameters parsed from an input file with their default values. These parameters are used by simulation

tools and metamodels for numerical calculations. Moreover, they are modified during optimization process, thus, a user can control optimal solution during each iteration of optimization loop.

- Goal function – presents predefined objective functions for the selected process and external. The parameters of this function are kept in the database together with the whole parameterized equation. These equations are parsed by NCALC library (http://ncalc.codeplex.com/) before evaluation of their values.
- Material – contains list of materials stored in remote database and allows to edit selected material or to add a new one. All modifications are also passed to database management engine, which is installed on external server.
- Simulation – panel based on a card layout includes two elements, i.e. numerical simulation or metamodel. Both cards allow user to select particular simulation tool and its input file, if available, according to which the parameters are changed.
- Optimization – allows start of optimization procedure and presents current progress on the bar. Start of optimization runs configuration wizard at first, by which user selects the optimization method, e.g. EA or PSO, and setup its parameters. Then the multi-iterative calculations are executed.

4. USE CASES

The system was tested by using different configurations of industrial processes and cycles. Two most interesting cases of rods and rails production are presented in this chapter.

4.1. Production cycle of rods

The whole production cycle for fasteners is composed of a number of operations, see publications (Pietrzyk et al., 2008; Kuziak et al., 2011; Skóra et al., 2012). The main operations are: hot rolling, accelerated cooling, pickling, phosphatizing, drawing, annealing, multistage forging and heat treatment (figure 6). Simulation results of the whole production chain were presented in those publications, therefore, an optimization of this chain using the developed system is theoretically possible. On the other hand, such optimization would be extremely time consuming. Since the direct influence of the hot forming part of the cycle on the properties of the cold-formed product is negligible, the optimization of the whole chain is not justified from the practical point of view. Therefore, an approach known as variant optimization was applied by Skóra and Pietrzyk (2014) for the sequence of cold forming (drawing, multi stage forging) operations. In that approach trial and error simulations were combined with the knowledge of an expert and the best solution was found.

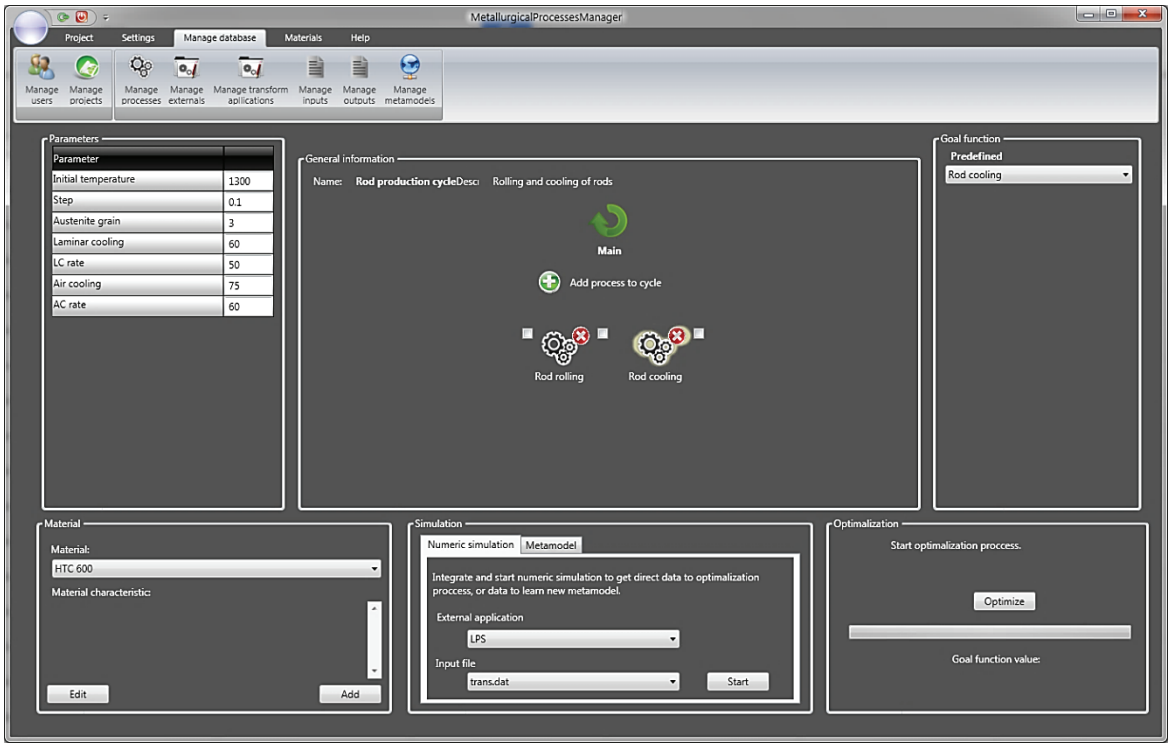


Fig. 5. The main window of the system.

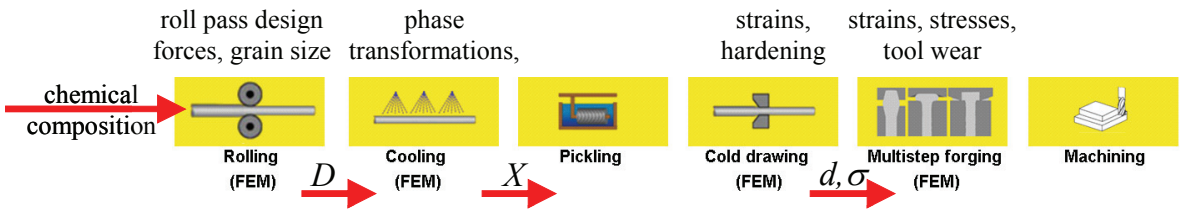


Fig. 6. *Schematic illustration of the manufacturing chain for fasteners; parameters transferred between the processes are: D – austenite grain size, X – volume fractions of phases, d – rod diameter, σ - flow stress distribution.*

In the present work optimization of the hot part of the manufacturing chain was performed using the developed system. The optimization of hot rolling included numerical simulations in Abaqus with dedicated procedures implemented in Python programming language. Detailed description of applied optimization procedure and obtained results are presented by Legwand et al. (2014). The process of rod cooling after rolling is presented in this work. The Evolutionary Algorithm approach was applied as optimization method and executed in 100 iterations with 20 individuals in the population. Time of cooling and heat transfer coefficient were selected as design variables in optimization. The heat transfer was defined as the ratio between the water pressure and the nominal water pressure in the accelerated cooling system. The objective function was defined as follows:

$$\Phi = \sqrt{w_b \frac{1}{|\Omega|}\oint_x \left(Fb(x) - Fb_{dest}\right) d\Omega + w_m \frac{1}{|\Omega|}\oint_x Fm^2(x) d\Omega + w_{r02} \frac{1}{|\Omega|}\oint_x \frac{1}{R_{02}^2(x)} d\Omega} \quad (1)$$

where: Ω – surface of rod slice, Fm – volume fraction of martensite, Fb_{dest}= 0.2 – assumed value of bainite fraction, F_b – volume fraction of bainite, R_{02} – yield stress, w_m = 1.0, w_b = 9.0, w_{r02} = 100.0 – optimization weights for martensite, bainite and yield stress. The values of weights were arbitrary applied. After 13th iteration the minimum value of the objective function has been found for time of cooling 1.6 s and heat transfer coefficient 0.54.

The convergence of the optimization method is presented in figure 7. Finally determined values of design variables of optimization allowed to obtain the following average values of parameters used in objective function: martensite fraction Fm = 0.196, bainite fraction F_b = 0.09, yield stress R_{02} = 372.8 MPa.

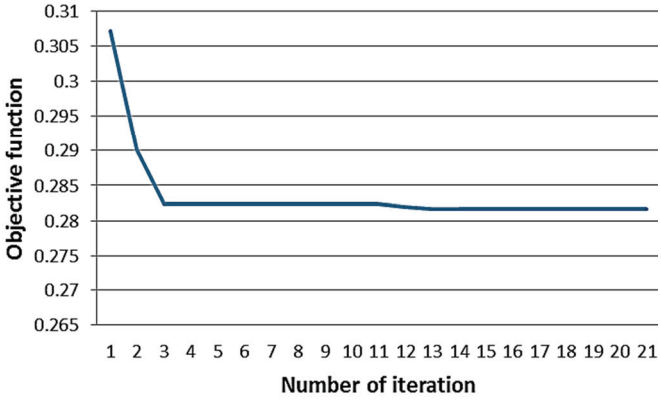


Fig. 7. *The convergence of the Evolutionary Algorithm for optimization of rods cooling.*

4.2. Production cycle of rails

Over last decades a constant progress has been observed in the rail transportation sector, connected with an increase in train speed and application of greater axle loads due to an increase in the weight of materials carried by rail transport. The rail transport capacities depend strongly on the track parameters. This requires rails with increased abrasive wear strength, fatigue strength and resistance to contact-fatigue defects occurrence. These features can be obtained in rails by specific microstructure of used steel, especially for pearlite structure after accelerated cooling with small distance between the cementite lamellae (S_0 around 0.10-0.12 μm), as compared to the structure after the natural cooling in the air (S_0 around 0.2-0.3 μm). Kuziak and Zygmunt (2013) has proposed the heat treatment process for the rail head, which allows to obtain pearlite with small interlamellar spacing and optimization of this process was considered in this paper.

The new method of pearlitic rails head heat treatment has two objectives. Maintaining the pearlitic transformation temperature as low as possible to obtain small interlamellar spacing is the first objective. Avoiding bainite in the rail head and maintaining purely pearlitic microstructure at the whole cross section of the rail is the second objective. Beyond

this, uniform distribution of hardness at the cross section of the rail head is required. These objectives were used to formulate the optimization task in the present work.

On the basis of the Kuziak and Zygmunt (2013) the system for controlled cooling of rails was build and installed in IMŻ Gliwice (Kuziak et al., 2011) and this system was considered in the present work. The rail is inserted in the tank with the coolant directly after rolling, and initially the level of the solution is kept below the running surface level (solid line in figure 8). The process of accelerated cooling starts when the rail head temperature reaches about 820°C. Then the pumps supply the solution to the tank, which leads to gradual immersion of the rail head (figure 8 – dotted line). Keeping the head immersed for a longer period of time would lead to occurrence of the bainite in the microstructure. Therefore, the head is immersed cyclically into the polymer solution for short periods of time, after which part of the solution is removed from the tank, which brings its level below the running surface of the head. Thus, the rate at which heat is transferred to the environment from the head is substantially reduced. At this stage transfer of heat from the hotter rail head centre increases the temperature of the running surface. For the proper performance of the head hardening process, it is critical for the running surface temperature not to increase above 570°C. To prevent this, the coolant level is raised in the tank again and accelerated cooling stage is repeated. Sequence "accelerated cooling – cooling in still air" can be repeated many times until the pearlite transformation is completed in the entire head. Proper selection of times is crucial to maintain low temperature of pearlitic transformation and to avoid danger that the running surface is cooled below the start temperature of the bainitic transformation prior to the pearlitic transformation completion.

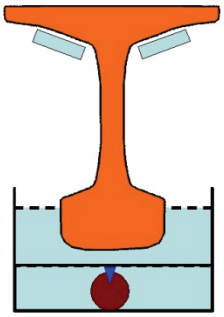


Fig. 8. *Schematic illustration of one stage of controlled cooling of rail head; head above the level of the solution (solid line) and head immersed in the solution (dotted line).*

The times of the immersion and the times of the air-cooling in subsequent stages, as well as heat transfer coefficient, form the set of four design variables in the optimization. The following objective function was proposed:

$$\Phi = \sqrt{w_{hb}\frac{1}{|\Omega|}\oint_x\left(\frac{HB(x)-HB_{avg}}{HB(x)}\right)^2 d\Omega + w_{S0}\frac{1}{|\Omega|}\oint_x S_0^2(x)d\Omega + w_b\frac{1}{|\Omega|}\oint_x Fb^2(x)d\Omega} \quad (2)$$

where: Ω – surface of the railhead, HB_{avg} – the average hardness of the rail head, S_{0i} – interlamellar spacing, Fb – volume fraction of bainite, w_{hb} = 1.0, w_{s0} = 100.0, w_b = 1.0 – optimization weights for hardness, interlamellar spacing and bainite. The values of weights were arbitrary applied.

Objective function (2) should allow to find the cooling sequence, which gives the minimum volume fraction of bainite and minimum interlamellar spacing, as well as uniform distribution of the hardness in the rail head.

The whole rolling-cooling sequence was considered. Rolling process was simulated using the finite element code Larstran with the microstructure evolution model for the rail steel implemented in this code (Smyk et al., 2013). Austenite grain size distribution is the main output parameter of rail rolling model, which is used in further simulations of controlled cooling. The austenite grain size influences the kinetic of pearlitic transformation. Temperature distribution at the cross section of the rail was calculated using finite element code described by Lenard et al. (1999). At each Gauss integration point of this code, equations describing phase transformations and properties of the material, are solved. The equations, which allow to calculate volume fractions of phases, as well as interlamellar spacing and hardness, were proposed by Pietrzyk and Kuziak (2000) and were further developed by Pietrzyk and Kuziak (2012).

The process of rails cooling was optimized by using the Evolutionary Algorithm approach. Similarly to the case from chapter 4.1 the method was executed in 100 iterations with 20 individuals in the population. After 18th iteration the minimum value of the objective function has been found for the following design variables: heat transfer coefficient - 500 W/m^{2}K (in the optimization it was understood as a decrease of the heat transfer coefficient with

respect to its maximum value for the polymer solution), time of the first immersion 51.6 s, time of the air cooling 42.9 s and time of the second immersion 54.7 s. These parameters of the process result in the following values of objective function parameters: deviation of hardness HB_{dev} = 0.11, interlamellar spacing S_0 = 0.134 μm, bainite fraction Fb = 0.01. The convergence of the optimization method is presented in figure 9.

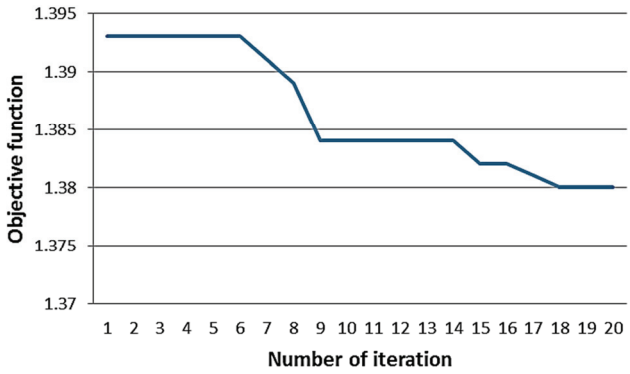


Fig. 9. *The convergence of the Evolutionary Algorithm for optimization of rails cooling.*

5. CONCLUSIONS

Computer system supporting design of production processes and cycles in metal forming industry was presented in the paper. The main goal of this system, i.e. integration of different simulation tools, optimization library, sensitivity analysis library, metamodelling and material database, has been achieved. The system is user friendly and adapts easily to new use cases. The system was tested for various local and remote simulation tools as well as for different scenarios of production cycles optimization. Obtained results proved high reliability of applied numerical procedures.

The future development of the system will be directed to implementation of new communication layer for higher spectrum of e-infrastructure, which is still dedicated only for some selected computer clusters environments. Additionally, the metamodelling built-in module will be equipped with new functionalities allowing not only execution of metamodels, but also their training and validation by using testing sets of data.

ACKNOWLEDGEMENTS

Financial support of the NCBiR, project no. R07 0006 10, is acknowledged.

REFERENCES

Kuś, W., Mucha, W., 2014, Idea of the optimization strategy for industrial processes, *Computer Methods in Materials Science*, 14, (1), 13-19.

Kuziak, R., Skóra, M., Węglarczyk, S., Paćko M., Pietrzyk, M., 2011, Computer aided design of the manufacturing chain for fasteners, *Computer Methods in Materials Science*, 11, 243-250.

Kuziak, R., Zygmunt, T., 2013, A new method of rail head hardening of standard-gauge rails for improved wear and damage resistance, *Steel Research International*, 84, 13-19.

Legwand, A., Perzyński, K., Madej, Ł., Pietrzyk, M., 2014, Approach for an automatic optimisation of production chain as a tool for intelligent manufacturing in metal forming, *Computer Methods in Materials Science*, 14, (1), 64-74.

Lenard, J.G., Pietrzyk, M., Cser, L., 1999, *Mathematical and physical simulation of the properties of hot rolled products*, Elsevier, Amsterdam.

Miyata, S., Kudo, K., Kato, T., Hiroyasu, T., Miki, M., Kamiura, J., Hiroyasu, H., Uchiyama, M., Fukumoto, M., 2003, Mass simulations based design approach and its environment: Multi-objective optimization of diesel engine with Distributed Genetic Algorithm using iSIGHT, MOGADES and HIDECS, *Parallel Computational Fluid Dynamics 2002*, 499-506.

Pietrzyk, M., Kuziak, R., 2000, Modeling of controlled cooling of rails after hot rolling, *Proc. Conf. Rolling 2000*, Vasteros, CD ROM.

Pietrzyk, M., Madej, Ł., Węglarczyk, S., 2008, Tool for optimal design of manufacturing chain based on metal forming, *Annals of the CIRP*, 57, 309-312.

Pietrzyk, M., Kuziak, R., 2012, Numerical simulation of controlled cooling of rails as a tool for optimal design of this process, *Computer Methods in Materials Science*, 12, 233-243.

Skóra, M., Węglarczyk, S., Kusiak, J., Pietrzyk, M., 2012, Computer aided design of manufacturing of fasteners – selection of the best production chain, *Key Engineering Materials*, 504-506, 157-162.

Skóra, M., Pietrzyk, M., 2014, Proper assembly and geometrical parameters as a criterion for the computer aided design of manufacturing cycle for screws, *Computer Methods in Materials Science*, 14, (1), 75-85.

Smyk, G., Franzke, M., Kuziak, R., Pietrzyk, M., 2013, Numerical simulation of rolling-cooling sequence for rails, *Hutnik-Wiadomości Hutnicze*, 80, 593-597.

Wang, W., Mo, R., Zhang Y., 2011, Multi-objective aerodynamic optimization design method of compressor rotor based on isight, *Procedia Engineering*, 15, 3699-3703.

KOMPUTEROWY SYSTEM DO OPTYMALIZACJI PROCESÓW I CYKLI PRODUKCYJNYCH PRZETWÓRSTWA METALI

Streszczenie

Artykuł przedstawia projekt oraz implementację systemu komputerowego przeznaczonego do optymalizacji procesów oraz cykli produkcyjnych. Zaproponowany system został zaprojektowany jako oprogramowanie umożliwiające elastyczną integrację

różnych programów komputerowych dedykowanych do obliczeń numerycznych dla procesów przetwórstwa metali. Ponadto, system integruje również bibliotekę do optymalizacji, analizy wrażliwości, metamodelowania oraz zewnętrzną bazę danych udostępniającą informacje o materiałach. Dzięki takiemu rozwiązaniu zaproponowany system komputerowy posiada funkcjonalność wspierającą projektowanie procesów i cykli produkcyjnych z wykorzystaniem wieloiteracyjnych metod optymalizacji oraz symulacji numerycznych i metamodeli. Kolejne rozdziały opisują główną ideę systemu, wybrane szczegóły implementacji, a także analizę przypadków użycia dla różnych procesów i cykli produkcyjnych oraz uzyskane wyniki.

Received: December 14, 2013
Received in a revised form: January 1, 2014
Accepted: January 3, 2014