Publishing House
A K A P I T

COMPUTER METHODS IN MATERIALS SCIENCE
Informatyka w Technologii Materiałów

Vol. 12, 2012, No. 4

CASE-BASED REASONING APPROACH TO CONTROL OF INDUSTRIAL PROCESSES

GABRIEL ROJEK*, JAN KUSIAK

AGH University of Science and Technology, Faculty of Metals Engineering and Industrial Computer Science, Department of Applied Computer Science and Modelling
al. A. Mickiewicza 30, 30-059 Kraków
**Corresponding author: rojek@agh.edu.pl*

Abstract

The main topic of this article is the application of the case-based reasoning (CBR) methodology in the domain of control of industrial processes. Fundamental issues concerning CBR are presented as the background for the analysis of the application possibilities. The industrial control system based on the CBR approach process the knowledge concerning past production cases in order to find a case, that is relevant for a control of the present production. The found relevant case is being reused in the present problem as a pattern for control of industrial process. The CBR methodology enables also to learn from the solved current problem. Presented in the article analysis of design of the industrial control system with the CBR approach is a basis for implementation of an experimental application.

Key words: case-based reasoning, industrial process, process control, artificial neural network

1. INTRODUCTION

Case-based reasoning (CBR) is a research field in Artificial Intelligence. The main paradigm in this method is reasoning by reusing the knowledge gained in the past similar situations in solving the investigated current problem. A case-based reasoning solver uses *case base* - data set of past made and stored experience items (called cases). Each time new problem has to be solved, a case relevant to that problem is selected from the *case base* and adapted to the current situation. This approach is different from other major AI approaches. CBR is relying on situations experienced in the past instead of using general knowledge of a problem domain (Aamodt & Plaza, 1994). A distinctive feature of the CBR approach is incremental, sustained learning. Each time a problem is solved, a new experience is retained, that is available for future reasoning concerning new problem. CBR is, from the point of view of sustained learning, a cyclic process of solving new problems and gaining experience.

The main goal of this paper is analysis of possibilities of the CBR methodology application to control of industrial processes. Control of industrial processes is a new domain of application of the CBR methodology, with no known references in scientific literature regarding the control of metallurgical processes. As it is forseen, the CBR methodolgy should be especially efficient in control of industrial processes, which are nowadays controlled manually by operator using his/her experience. In such cases, the CBR control system can be an analogy to proceess of using and gaining of experience by a human operator. Results of an attempt of the CBR application to the zinc roasting process and a hypothetical simulated process are presented.

250 – 258 ISSN 1641-8581

2. CASE-BASED REASONING

2.1. Historical development of CBR

Solving a new problem by reusing information and knowledge of a previous similar situation is the basic idea of the CBR research field. This paradigm is supported by results from cognitive psychological research. Several studies indicate that the use of previously experienced situations (called as cases in the CBR research) has the dominant role at human decision making process, as presented by Ross (1989) or Anderson (1983). Problem solving by analogy is used as synonym for case-based reasoning. Solving by analogy in which an emphesize of the use of past experience as the means for solving of new and different problems was presented by Gentner (1983) and Carbonell (1986). The CBR methodology is also supported by many theories within phylosophical science, e.g. theory of 'natural concepts' presented by Wittgenstein (1953). This theory indicates that concepts, that are part of natural world, can be classified in a variety of ways. It is not possible to present a useful definition as a set of necessary and sufficient features for such concepts that can be defined only by set of instances, or cases in the CBR terminology. This theory coincides with the CBR methodology, in which a solution of a problem is generated on the basis of a set of previously gained eperience.

The first system, which implements the CBR methodology, is the CYRUS system, developed by Janet Kolodner at Yale University in 1983. The key element of this system is a case memory model. This model is a basis for several other case-based reasoning systems, e.g. MEDIATOR (Simpson, 1985), PERSUADER (Sycara, 1988) or CASEY (Koton, 1988). The PROTOS system focuses on integration of general domain knowledge and specific case knowledge, as it is presented in (Bareiss, 1988). One of earliest results in Europe was the MOLTKE system, which is a case-based resoner for complex technical diagnosis, as presented by Althoff et al. (1989). System presented by Plaza and López de Mántaras (1990) is a case-based learning system for medical diagnosis. There are many fields of applications of the CBR methodology (Aamodt & Plaza, 1994). One of known implementations is supporting user in loading of autoclave convection oven, where airplane parts are treated in order to get the right properties. This system uses relevant earlier situations in order to give advice for the current load. A system that aim is to select the most appropriate mechanical equipment at construction of ships is another known application of the CBR methodology. Other application areas of the CBR approach are help-desk and customer service, recommender systems in e-commerce, knowledge and experience management, medical applications, applications in image processing, applications in law, technical diagnosis, design, planning and human entertainment (computer games, music), as it is presented by Bergmann et al. (2009).

2.2. The CBR methodology

Despite many differences in construction CBR systems, the main algorithm of every CBR application stays common and consists of the following four sequential phases, called as a *CBR cycle* (Aamodt & Plaza, 1994) (figure 1):

1. *Retrieve* the most similar case (or cases) from the *case base.*
2. *Reuse* the information and knowledge in the retrieved case (or cases) in order to find a solution of a current problem.
3. *Revise* the proposed solution.
4. *Retain* the parts of this experience in the *case base* in order to use it for future problem solving.

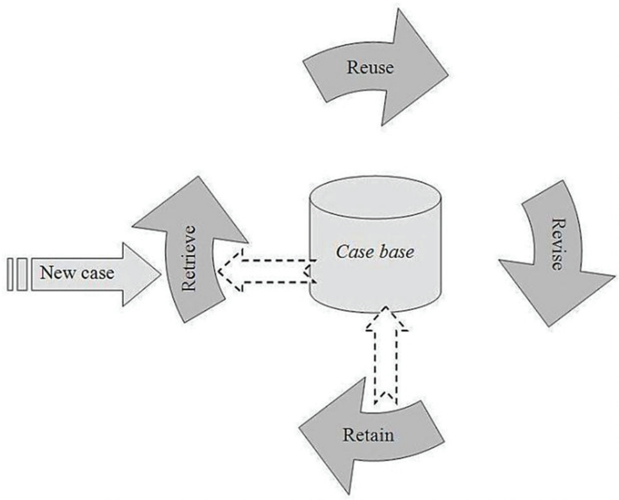

Fig. 1. *The CBR cycle.*

The CBR cycle interacts with the case base that contains knowledge regarding the domain of solved problem. That knowledge is stored in the form of cases, corresponding to the previously occurred and solved production cases.

Preliminary task at the design of a CBR system is construction of that *case base*, which means defining of a *case* and the information that has to be handled by this *case*. In some domains of applications, before first run of the *CBR cycle*, the *case base* is filled with *cases* collected during the past experience (activity/production/etc.).

The *CBR cycle* begins with a presentation of a new problem that has to be solved (called also as: new case, query or current problem). In the *retrieve phase*, one or more similar cases relevant for the current problem are selected from the *case base*. In the *reuse phase*, the solution corresponding to the selected case (or cases) is adopted to the current case. In the *revise phase*, the solution determined in the previous phase is verified in the current real conditions and, if possible, improved (e.g. by an expert). In the *retain phase*, the *case base* is updated by taking feedback from the current *revise phase*. In that phase, the CBR system learns from the new case stored in the case base, which can be a support in solving new problems in future. Details of subsequent phases of the *CBR cycle* are presented in Sections 3.2-3.5.

3. APPLICATION OF THE CBR METHODOLOGY TO CONTROL OF INDUSTRIAL PROCESS

The main objective of any industrial control system is achievement of required values of chosen criteria characterizing the whole production chain (e.g. economical or energetic criteria) or the final product (e.g. quality or dimensional criteria). The chosen criterion can be reached by control of the process signals and parameters on the basis of observation of measured process signals. Despite the increase of computer methods in industrial applications, many industrial processes are still controlled only manually by operators. There are many reasons for the use of manual control, e.g. the lack of knowledge concerning the chemical or physical nature of a controlled process or just problems with irregular measurements of signals that influence a process. The oxidizing roasting process of zinc sulphide concentrates, presented by Sztangret et al. (2011), is an example of such manually controlled process and has been chosen as the illustration for presented analysis concerning industrial implementation of the CBR methodology. Signals that influence considered process can be classified into three groups (Sztangret et al., 2011):

– Independent signals – cannot be modified or changed during a production cycle (e.g. chemical composition of raw materials, air pressure). It is only possible to measure an independent signal but not to change its value in direct or indirect way during the considered period of time. Independent signals usually have an influence on a quality criterion of the product like other input signals.
– Controllable signals – it is possible to set or to change them during a production process. An example of a controllable signal is a rotary speed of a fan cooling the controlled process. A voltage setting can directly modify this speed.
– Dependent signals – cannot be directly modified. Values of dependent signals can be interfered by other signals: independent, controllable or noises.

Presented above classification is schematically illustrated in figure 2.

Large discrepancies in sampling of all input signals and quality criterion are characteristic for majority of industrial processes. Some of independent

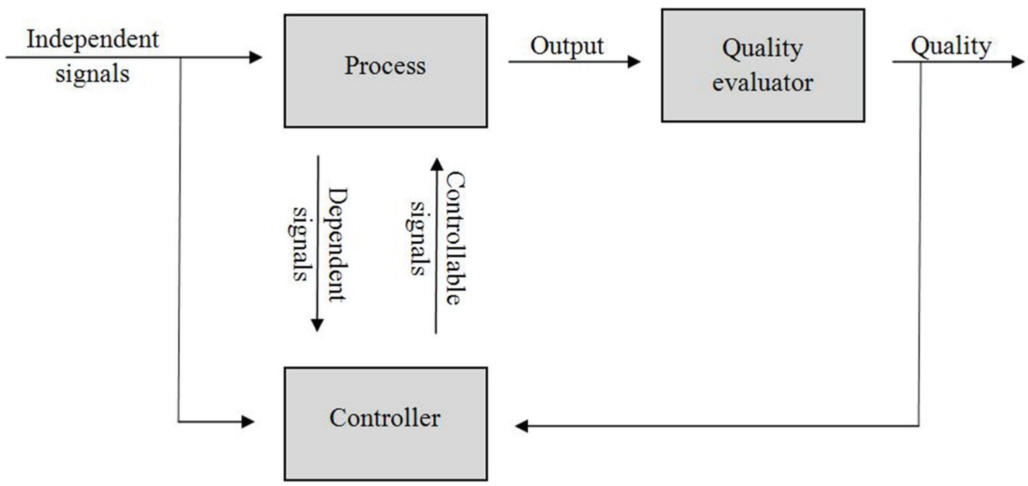

Fig. 2. Classification of the process signals.

signals are measured manually only once per production day (e.g. chemical composition of raw materials). On the other hand, dependent signals and controllable signals are measured on-line with much higher frequency (e.g. one minute or one second).

The goal of control of the considered process is to achieve the minimal concentration of sulphide sulphur in the roasted products, which can be considered as a process quality criterion. That criterion is measured manually several times a day (with random frequency). Such large discrepancies in sampling of signals and measurements of the quality criteria are obstacles in implementation of the automatic control systems, so the manual control based on the experience of the operator is still in use. Since CBR also relays on experience (that is retained every time the new problem is solved), the possibility of application of the CBR methodology to process control seems worth to study. Application of the CBR methodology to industrial process control requires definition of all elements of the *CBR cycle* described in section 2, which is presented in the following subsections.

3.1. A case and a case base

One of the most important tasks of the application of CBR methodology is a definition of a *case* and design of the *case base.* A *case* is usually defined as an experience item performed in the past. A *case* can be represented in the computer system in different ways. It can be a data structure, an instance of a class, an item in a table of database, or even an agent, when the agent technology is used.

In a domain of industrial processes an experience item relates to a time period of production. This period of time should be long enough to allow evaluation of final production results. Taking into consideration observations concerning frequency of different types of signal sampling and frequency of evaluation of the quality criterion in the considered zinc roasting process, one working day is proposed as the time period of a case. Such case contains all information concerning problem of the control of the considered industrial process that consists of:

– Description of the problem relating to the experience item.
– Description, how the problem was solved.
– Description, how the solution of the problem was assessed.

Referring to industrial control domain, values of independent signals (e.g. chemical composition of raw materials) describe a problem, that has to be solved, or was solved in the past. Independent signals influence the production, however, dependent and controllable signals are these, which assure the achievement of required product quality, and which can be seen as the solution (control signals) of a new problem. The solution for the specified problem has to be assessed. For the considered zinc roasting process, the assessment of a solution is described by the average quality measure of production made during the time period suitable for the case. A case, as a data structure, is presented schematically in figure 3.

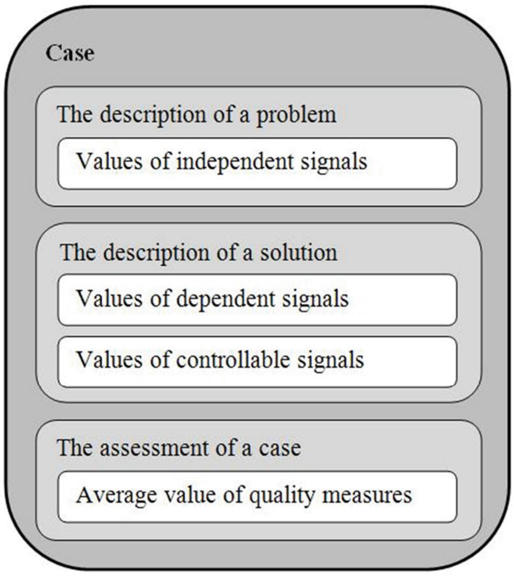

Fig. 3. *Definition of a case in the industrial control domain.*

A *case base* is just a collection of all cases, which refer to experiences gained in the past. It is possible that two or more cases refer to the same or very similar problem (described by similar values of independent signals) and the solutions are different.

Case base is continuously growing, because every solved problem is being stored in it. As a consequence of increasing number of stored cases, the efficiency of retrieval task decreases, because more and more previous cases must be searched. The problem of the growth of the *case base* can be solved with specific index structures in order to improve retrieval efficiency, as presented by Bergmann (2002).

3.2. Retrieve Phase

The *retrieve phase* is defined as a selection of one or a group of cases from the *case base*, with the highest similarity to the new case that has to be solved. Main task of this phase is to find a nearest case (or cases) (e.g. *k-nearest-neighbours)* using a specific similarity measure. The similarity measure can be inverse Euclidean, Hamming distance or any other, depending of the application domain. The similarity measure induces the preference order of similar cases to the new problem. The preference order should enable to select one (or more cases), which are relevant to the new case (that represents current problem).

In the domain of industrial control, relevant previous solved case is a case, of which: (1) description of the problem is similar to the present one and (2) solution of the problem is the best one of all similar cases. It is proposed first to choose cases representing similar problem and next to select among them only one that has the best assessment value. The *retrieve phase* consists of two steps:

1. Search of such cases from the *case base,* which are characterized by the highest similarity of independent signals to these of the current problem.
2. Selection of a single case among the cases found in step (1), which gives the best value of the quality criterion[1]. That selected case becomes the basis for next phases of the *CBR cycle*.

3.3. Reuse Phase

After selection of a case in the *retrieve phase*, the solution contained in that case can be reused to solve the current problem. The *reuse phase* can be very simple, when the solution can be used directly (without any change) as the proposed solution for the current case (this happens often for most of classification tasks). In such case the solution can be just copied. On the other hand there are domains, which require adaptation of the solution. There are two main ways to adapt retrieved past case to the current problem: (1) transformation of the past case, (2) reuse of the past method that generated the solution, as it is presented by Aamodt and Plaza (1994). The first way requires some knowledge by transformation using operators, which enable to transform the past case into a solution for the new case. The second way is called derivational reuse and involves inquiring of the way, how problem was solved in the retrieved case. The retrieved case contains information about the method used for solving the retrieved problem. This information should be used to restore the instance of the retrieved method and next this method should be used at the new context of the current problem.

In case of the CBR application to industrial control, every past case contains description of the solution in the form of the values of dependent and controllable signals (as it is presented in figure 3). Having a goal to reuse the solution included in the selected past case, that solution has to be modeled and next used in the control of present case of production (what refers to the derivational reuse). The *reuse phase* consists of two steps:

1. A model of the solution relevant for the selected past case should be prepared by using values of dependent and controllable signals.
2. The prepared model can be used to solve the actual problem.

The *reuse phase* can be realized by application of artificial neural networks or other modeling methods. In the case of using the artificial neural network as the modeling method, such network takes dependent signals as the input signals and returns controllable signals as the output. That artificial neural network should be trained with values of dependent and controllable signals corresponding to the case found in the *retrieve phase*. In the second step the trained network can be used in order to predict values of controllable signals on the basis of present measured dependent signals. During the second step, all values of dependent and controllable signals should be saved in order to be used during the *retain phase*.

3.4. Revise Phase

The *revise phase* concerns the feedback of the applied in the *reuse phase* solution into the current problem that is being solved. That solution is evaluated and in a case of negative evaluation results, there is a possibility to repair that solution using domain-specific knowledge. This phase consists of two tasks: evaluation of the solution and fault repair (Aamodt & Plaza, 1994).

The evaluation task uses the result of application of the suggested (in the *reuse phase*) solution to the real environment. This task is usually performed outside CBR system and makes necessary to link CBR system with real conditions. Depending on the type of the application domain, the evaluation can take some time to be performed (e.g. time needed for

[1] If more cases represent the best value of the quality criterion, additional criterion of selection has to be implemented in order to select a single case.

performing some processes in the application domain).

Fault repair involves detecting the errors of the current solution and searching the explanations for them. The failure explanation can be used next to modify the solution and to eliminate the errors occurrence.

In the domain of industrial control, the feedback of the *revise phase* is in the form of quality measures of real products made during current period of production. In the case of considered zinc roasting process the measurement of the quality value is distant in the time (measured at the end of the current production period). It causes, that fault repair process is not possible, however the quality measure is important due to its use in the next phase of the *CBR cycle*.

3.5. Retain Phase

The *retain phase* starts, when the current problem was solved and its solution was found. That phase is considered as the learning process in CBR system (Bergmann et al., 2009). This learning process usually occurs by simply adding the revised case to the *case base*. Thanks this, the whole system gains new experience according to the present case and the revised solution becomes available for reuse in solving future problems. The *retain phase* enables to learn from a success or a failure of the proposed solution. However, as it was mentioned, continuous increase of the *case base* causes decrease of efficiency of the *retrieve phase*.

Considering proposed in subsection 3.1 definition of a case of analyzed industrial process, the current case already has known description of the problem, description of the applied solution (in the form of values of dependent and controllable signals already saved during the *reuse phase*) and the assessment (in the form of average value of quality measures during the *revise phase*). The current case is just added to the *case base* and becomes one of past cases representing experience items stored in CBR system.

4. EXPERIMENTS WITH APPLICATION OF THE CBR METHODOLOGY

The first Authors' experiment with application of the CBR methodology to the industrial domain was a design and development of a system, which realizes the first phase of the CBR cycle – the *retrieve phase* – was presented by Rojek et al. (2011). Presented here further attempt to implementation of the CBR cycle was limited by the *revise phase*, which consists of the evaluation tasks. The evaluation task, as it is assumed e.g. by Aamodt and Plaza (1994), should be performed outside the CBR system by application of solution suggested by the system to the real process. Since it is an initial stage of the research and costs of such experiments are very high, the *revise phase* was not possible. However, the problem with the costly evaluation of solution suggested with CBR reasoning system can be solved by the use of a simulation of an industrial process instead of a real process application. Application of the CBR approach presented by Rojek and Kusiak (2012a, 2012b) concerns a simulation of an industrial process. This simulation enables to obtain the average value of quality measures for one production day knowing all input signals. In order to control the simulated process a control system with the implemented CBR methodology was designed and implemented. This system performs only first three (of all four) phases of the CBR cycle. 30 simulated past days of production are a basis for the *case base*. Results presented by Rojek and Kusiak (2012a) concern 10 days of simulated production, which are controlled with the implemented system. Average quality measure computed for 30 past production days was 49.45. The average quality measure for 10 days controlled by the CBR system was 34.63 and was better than for past 30 days, due to the fact that smaller values of quality measure were better. Tests presented by Rojek et al. (2012b) were performed with the use of the same control system. Those tests were expanded to 30 days of production controlled by the CBR system. The average quality measure for 30 days under control of implemented system was 27.78, what is even better average result than for 10 days.

Presented in the following subsections simulation of a hypothetical industrial process and a control system with the implemented CBR methodology are modifications of works presented by Rojek and Kusiak (2012a, 2012b). However the simulation is only simplification of a real industrial process, this simulation enables to evaluate results of the control system, what is main advantage of such research.

4.1. Simulation of a hypothetical industrial process

Presented simulation is inspired by the oxidizing roasting process of zinc sulphide concentrates, presented by Sztangret et al. (2011). Every single day of production is characterized by:

- Independent signal z: one value, that is a floating point number from a range (0,10).
- Dependent signals x_1 and x_2: a vector of 100 values x_1 and x_2. Every x_1 and x_2 value is a floating point number from a range (0,10).
- Controllable signals u_1 and u_2: a vector of 100 values u_1 and u_2. Every u_1 and u_2 value is a floating point number from a range (0,20).
- Quality criterion q: one value of a non-negative floating point number.

Basis for next presented control system are 50 days of past production. Values of above presented signals for 50 days are obtained by using a simulation. Values of independent signal (z) and dependent signals (x_1 and x_2) are set by random numbers. Values of controllable signals are computed as multiplication: $u_1=w_1 \cdot x_1$ and $u_2=w_2 \cdot x_2$, where w_1 and w_2 are floating point numbers from a range (0,2). Factors w_1 and w_2 are constant for every single production day. Those factors relate to the way of control, which is constant for the whole production day. For every one of computed 50 days factors w_1 and w_2 are set by random values.

The quality criterion (q) is equal to average of temporary quality, which is computed as $q_t = ((10 - z) - u_1)^2 + ((10 - z) - u_2)^2$. The goal is to obtain the smallest value of quality value. The quality of hypothetical products depends on values of independent signal and controllable signals, which are set, as it is assumed, on the basis of dependent signals. The same quality criterion is used to assess 50 past day of production, which are basis for *case base* of presented next control system, and is used to assess results of control, which is solution obtained by using of control system. The quality criterion is unknown for the control system and is only used at simulation of production process.

4.2. Control system

A control system was implemented for simulation presented in the previous subsection. At implementation of this control system agent technology was used. The main idea is to divide functioning of the whole system between agents, which realize parts of the whole system.

The fundamental task at implementation of CBR system is construction of a *case*. As it was presented in subsection 3.1, a *case* represents one day of production. Use of the agent technology enables to implement a *case* as an agent, which is called Past Episode Agent. Every single past case is represented by one Past Episode Agent, that stores all data, as it is presented in figure 3. The Past Episode Agent can also communicate with other agents in the system and give information according to stored data. A *case base* is represented by agents. Because 50 past days of production are obtained from simulation, the *case base* contains 50 past *cases* of production, each of them is representing one past day of production. So, at the start of presented control system 50 Past Episode Agents exist in the system.

It is assumed, that the control of production starts when value of independent signal is known. This time new agent, called Control Agent, is created. The main goal of this agent is to control the simulated hypothetical process. The first activity of this agent is taking current independent signals and searching for one Past Episode Agent, which represents relevant past case of production. This activity is realization of the *retrieve phase* and, as was presented in subsection 3.2, consists of two steps:

1. Search of Past Episode Agents, which represent cases with highest similarity measure. The similarity measure is Euclidean distance between values of dependent signal of the current case and the past case.
2. Selection of a single Past Episode Agent among the agents found in step (1), which represents a case with the best value of quality criterion.

After selection of one Past Episode Agent, that represents relevant case, the Control Agent moves to the *reuse phase*, which was presented in subsection 3.3. This phase consists of two steps:

1. Modeling of the way, how solution in the relevant case was obtained.
2. Using prepared model in order to solve the current problem.

These two steps are realized with the help of artificial neuron net. The used net is multi-layered perceptron, that consists of 5 layers of 2, 4, 7, 4, 2 neurons. Input signals are dependent signals x_1 and x_2, output signals are controllable signals u_1 and u_2. The first above presented step is realized by supervised learning. The learning set consists of dependent and controllable signals, which are stored by chosen Past Episode Agent. After the end of learning step, the Control Agent is using the net in order to predict values of control signals for current values of dependent signals.

The Control Agent stops its functioning, when the simulated production days ends. That time all values of independent, controllable and dependent signals are saved in a file. This enables to compute quality value, as presented in the previous subsection.

4.3. Obtained results

Table 1. presents characteristic of 50 days of production, which are used at construction of *case base* for presented system. The average quality value for 50 past days is equal to 52,51.

Presented in the previous section control system was run 25 times. Every time one new production day was simulated, by giving random independent signal (constant for the whole day) and random dependent signals (100 values of x_1 and x_2 per a day). The control system computed values of controllable signals (100 values of u_1 and u_2) for every day of production. The simulation enabled to assess the quality of hypothetical production, what is presented in table 2.

Table 1. *The quality values for 50 past production days.*

Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10
95,53	24,74	34,17	90,95	155,61	42,55	20,67	89,67	127,45	74,71
Day 11	Day 12	Day 13	Day 14	Day 15	Day 16	Day 17	Day 18	Day 19	Day 20
57,76	66,57	36,90	58,28	16,33	7,44	60,31	43,99	19,91	61,46
Day 21	Day 22	Day 23	Day 24	Day 25	Day 26	Day 27	Day 28	Day 29	Day 30
64,52	11,92	64,55	37,31	69,30	37,17	100,93	84,90	33,83	59,72
Day 31	Day 32	Day 33	Day 34	Day 35	Day 36	Day 37	Day 38	Day 39	Day 40
54,75	12,60	33,68	75,64	53,09	67,65	45,64	77,39	19,10	38,11
Day 41	Day 42	Day 43	Day 44	Day 45	Day 46	Day 47	Day 48	Day 49	Day 50
2,98	73,14	1,60	47,71	41,98	30,29	75,51	18,96	36,37	70,19

Table 2. *The quality values for 25 production days under control of implemented system.*

Day 1	Day 2	Day 3	Day 4	Day 5	Day 6	Day 7	Day 8	Day 9	Day 10
9,29	10,27	46,51	0,58	1,02	16,47	2,72	9,20	8,80	42,85
Day 11	Day 12	Day 13	Day 14	Day 15	Day 16	Day 17	Day 18	Day 19	Day 20
20,02	1,08	18,91	13,91	0,93	0,40	20,80	12,34	53,01	14,48
Day 21	Day 22	Day 23	Day 24	Day 25					
11,91	11,87	58,15	15,21	11,73					

The average quality value for 25 production days under control of the implemented system is equal to 16,50. This result is better than for 50 simulated days, because the aim of control is to obtain the smallest possible quality value.

5. SUMMARY AND CONCLUSIONS

Presented case-based reasoning (CBR) belongs to the field of Artificial Intelligence that integrates problem solving and learning as two inseparable processes. Problem solving uses the results of learned in the past cases, while the learned cases are effects of solutions of formerly analyzed problem. Although, there are many known application areas of CBR systems, especially applications supporting humans in decision making, the application of the CBR methodology to the control of industrial processes is rather unknown.

Done research concerning application of the CBR methodology to control of industrial process on the one hand relates to a real industrial process and on the other hand relates to simulated industrial processes. The experiment, that uses real industrial data, enables to foreseen, if it is even possible to implement the CBR methodology to the domain of industrial process. The experiments, that use simulation of an industrial process, enable to estimate results of such process control with the CBR methodology. In both cases obtained results indicate, that the CBR methodology can be successfully used as an approach to control of industrial process.

The future work should aim at practical tests of the whole CBR cycle in simulated industrial environment and should aim also at building the systems or its parts, which enable case-based reasoning in real production environment. It is worth also to analyze application of the CBR methodology (and its results) to control of experimental process in the sphere of metallurgy and materials engineering.

Acknowledgements. Financial assistance of the MNiSzW (Działalność statutowa AGH nr 11.11.110.085).

REFERENCES

Aamodt, A., Plaza, E., 1994, Case-Based Reasoning: Foundational Issues, Methodological Variations and System Approaches, *AI Communications*, 7, 1, 39-59.

Althoff, K.-D., Kockskämper, S., Traphöner, R., Wernicke, W., Faupel B., 1989, Knowledge acquisition in the domain of CNC machine centers; the MOLTKE approach,

EKAW-89; Third European Workshop on Knowledge-Based Systems, Paris, 180-195.
Anderson, J.R., 1983, *The architecture of cognition*. Cambridge: Harvard University Press.
Bareiss, R., 1988, *PROTOS; a unified approach to concept representation, classification and learning*, University of Texas at Austin.
Bergmann, R., 2002, *Experience Management - Foundations, Development Methodology, and Internet-Based Applications,* Springer-Verlag Berlin, Heidelberg.
Bergmann, R., Althoff, K. D., Minor, M., Reichle, M., Bach, K., 2009, *Case-Based Reasoning, Introduction and Recent Developments*, Kunstliche Intelligenz: Special Issue on Case-Based Reasoning, 23, 1, 5-11.
Carbonell, J., 1986, Derivational analogy; A theory of reconstructive problem solving and expertise acquisition. *Machine Learning: An Artificial Intelligence Approach*, 2, chapter 14, Morgan Kaufman Publishers, 371-392.
Gentner, D., 1983, Structure mapping - a theoretical framework for analogy, *Cognitive Science,* 7, 2, 155-170.
Kolodner, J., 1983, Maintaining organization in a dynamic long-term memory, *Cognitive Science,* 7, 4, 243-280.
Koton, P., 1988, *Using experience in learning and problem solving.* Massachusetts Institute of Technology.
Plaza, E., López de Mántaras, R., 1990, A case-based apprentice that learns from fuzzy examples, *Methodologies for Intelligent System,* 5, 420-427.
Rojek, G., Kusiak, J., 2012a, Industrial control system based on data processing, *Proceedings of the 11th international conference on Artificial Intelligence and Soft Computing - Volume Part II*, *Lecture Notes in Computer Science, Lecture Notes in Artificial Intelligence, 7268*, Springer-Verlag, 502-510.
Rojek, G., Kusiak, J., 2012b, System optymalizacji procesu produkcyjnego z zastosowaniem agentowego systemu przetwarzania informacji, *Hutnik-Wiadomości Hutnicze,* 79, 1, 71-74.
Rojek, G., Sztangret, Ł., Kusiak, J., 2011, Agent-based information processing in a domain of the industrial process optimization, *Computer Methods in Materials Science,* 11, 297-302.
Ross, B.H., 1989, Some psychological results on case-based reasoning, *Proceedings of the DARPA Case-Based Reasoning Workshop*, 2, Morgan Kaufmann, San Mateo, Calif., 144-147.
Simpson, R.L., 1985, *A computer model of case-based reasoning in problem solving: An investigation in the domain of dispute mediation (analogy, machine learning, conceptual memory),* Georgia Institute of Technology.
Sycara, K., 1988, Using case-based reasoning for plan adaptation and repair, *Proceedings of the 1988 Case-Based Reasoning Workshop*, Morgan Kaufmann, 425-434.
Sztangret, Ł., Rauch, Ł., Kusiak, J., Jarosz, P., Małecki, S., 2011, Modelling of the oxidizing roasting process of sulphide zinc concentrates using the artificial neural networks, *Computer Methods in Materials Science,* 11, 122-127.
Wittgenstein, L. 1953, *Philosophical investigations*, Basil Blackwell, 31-34.

WNIOSKOWANIE OPARTE NA EPIZODACH JAKO PODEJŚCIE DO STEROWANIA PRZEMYSŁOWEGO

Streszczenie

Głównym tematem niniejszego artykułu jest próba aplikacji wnioskowania opartego na epizodach (ang. Case-based reasoning, CBR) w zakresie produkcji przemysłowej. Prezentowane podstawowe zagadnienia dotyczące CBR stanowią tło dla rozważań dotyczących analizy możliwości aplikacji metodologii. System przemysłowego sterowania bazujący na podejściu CBR powinien przetwarzać wiedzę dotyczącą uprzednio wykonanej produkcji wraz z jej rezultatami w celu wyszukania epizodu we wcześniejszej produkcji, który jest odpowiedni dla sterowania bieżącą produkcją. Wyszukany epizod powinien być powtórnie użyty w bieżącym problemie sterowania, jako szablon sterowania procesem przemysłowym. Metodologia CBR pozwala również na naukę na podstawie bieżących problemów, które są rozwiązywane przez system implementujący tą metodologię. Prezentowane w artykule analizy projektu oraz implementacji systemu sterowania z podejściem CBR stanowią podstawę do implementacji eksperymentalnej aplikacji.

Received: November 29, 2012
Received in a revised form: December 20, 2012
Accepted: December 21, 2012