Practical Application of the Heat Distribution and Consumption Model

Vasek L., Dolinay V.

***Abstract*—** This article focused on the practical experiments made by designed and implemented computer model of the distribution system of heat consumption in the urban agglomeration (SHDC - System of Heat Distribution and Consumption). This model is designed as a simulation model connected to prediction mechanism. The simulation is one of the methods, which can be effectively used for the analysis of large and complex dynamic systems properties, which the distribution system and heat consumption in the municipal heating networks is. The model was implemented in the form of computer applications to provides interfaces to adapt it into a real heating system. To provide necessary functionality, the model takes basic information from the system to be modeled, such as the lengths and diameters of the real pipe system along with operational data. The model and its subsequent links are designed for heat supply prediction, which can be used in system regulation. Depending on the structure of the particular real system, the temperature of heating water is usually required to predict and the other variables such as mass flow and the water temperature in the return line.

***Keywords*—** Distribution, heat, model, prediction, simulation, temperature.

I. Introduction

MOST countries deal with the energy resources problems. Good example is distribution and consumption of heat energy in the urban agglomeration which is very recent, especially in the context of finite worldwide energy resources and in the constant of increases in energy prices. There are also important ecological aspects, because obtaining and using of energy generally has mostly negative environment impact. Therefore, it is necessary to seek all paths leading to energy, including heat energy, savings [7].

Heat energy must be transported to the place of consumption in time when it is required and in the expected quantity and quality. Quality of supplied heat energy is expressed in the temperature of heat transferring media. The correct delivery time, quantity and quality of heat energy must go hand in hand with minimal distribution costs [1].

System of production, distribution and consumption of heat is very large and complex. Analysis of the features that need to be known for its efficient management is very difficult and there is not much practice, how such an analysis should be carried out with sufficient accuracy. Applicable procedures are almost exclusively based on modeling.

This paper describes the designed and implemented computer model of the distribution system of heat consumption in the urban agglomeration (SHDC - System of Heat Distribution and Consumption) and its subsequent use in experiments on the real heating system.

For our model, the chosen city system was simplified and model was trained on real measured data [13]. The main aim of this experiment was to verify model itself, its ability to adapt to real process and also to proof associated potential for prediction.

Beside the model description, this paper shows results of two days experiment on heating system of midsize city with more than hundred heat exchangers. Even the experiment confirmed many theoretical presumptions; it also showed several new tasks to deal with for the further improvements.

II. Model Description

The distribution network can be presented as a set of sources of heat energy (supply heating stations) and heat consumers, which are cross connected through piping. The pipes are divided for model in sections, which are linked in nodes. Section starts and ends in the node and can be divided in several pipe lines. Pipe line is a part of piping, which has constant characteristic from the point of view flow and heat transfer.

Simulation time is running in discrete time intervals constant length signed as Δt. Time interval Δt is identical to the sampling time interval and Δtj determine the simulation step j. As basic “moving” element (“transaction” in simulation terminology) consider "discrete flow quantum" DFQ of fluid (water). The DFQ flows in the network and gradually loses its energy, depending on the current position.

Pipeline losses can be determined by the relationship

$$^{j}Q_{i\,loss} = k_p * (^{j}T_i - {}^{j}T_{p\,ext}) * \Delta t \qquad (1)$$

where:

- k_p is the heat transfer coefficient in the current pipe line *p*,
- $^{j}T_i$ is water temperature for the DFQ_i
- $^{j}T_{p\,ext}$ is the outside temperature for the pipe line *p*, both in simulation step *j*.

Coefficient k_p is based on pipe structure - pipe material, style and material of insulation, pipe seating, etc.

For example, for the heat consumption at consumer *r* at time interval Δt_j the following equation can be used:

$$^{j}Q_{i\,cons} = s_r(^{j}T_i, {}^{j}T_{r\,ext}, \dots) * \Delta t \qquad (2)$$

where:
- $s_r(...)$ is the function describing heat consumption for the consumer *r*.

Determination of this function is obviously very difficult, but for the final solution of this task, especially in terms of its accuracy for those particular parts "consumers", it is very important. There may be applied many different important factors such as:
- type of the day: workday , weekend, holiday etc.,
- part of the day: morning, afternoon, evening, night,
- type of the consumers in the particular part of the network: flats, schools, industrial companies etc.,
- other weather conditions: sun intensity, wind, air humidity.

To determine the functional dependences of heat consumption on these factors it is also possible to successfully use the proposed simulation model.

Detailed information about particular parts, such as the flow modeling and heat transfer modeling can be found in [7].

III. Model realization

Introduced model was implemented in the form of a software application. The program modules are written in Java. All data - historical operational data, configuration and description data for distribution network, simulation results – are stored in database. Connection to database is realized through JDBC interface and all queries to database are defined by SQL statements. These chosen software tools and solution allow easy portability to different software environments.

The most important JAVA class is class SIMULATOR, which realize separately one complete simulation experiment, i.e. one simulation run for whole simulated time period. This gives the possibility to write application as multithreaded, so that there are running in parallel several instances of the class SIMULATOR, every one in their own thread. It gives higher performance for calculation, especially in case, when many simulation experiments must be provided. The most of the experiment mentioned in this article were performed on multi core processors, which enable to achieve a large number of calculation and obtaining results in relatively good.

Block diagram for the whole application can be seen in Fig. 1. Block diagram for class SIMULATOR is in Fig. 2.

Individual blocks have the following functions:

Block “INITIALIZATION”:
- Read necessary data from database.
- Create and initialize internal data structures for program modules.
- Create and reset data structures for results.

Block “PARAMETERS”:
- Get set of model parameters. The set of parameters is generated by PSO algorithm. It is realized in the procedure, which has as output set of model parameters and input is value of fitness for given set of parameters. This procedure is a part of the block “PARAMETERS”.

Block “START SIMULATOR”:
- Create a new instance of SIMULATOR class to which given parameters are assigned.
- Define and start a new thread in which the new instance of SIMULATOR is running.

Block “WAIT FOT RESULTS”:
- Wait the end of simulation for each SIMULATOR instance in individual thread.
- If the end is reached, the result (the value of fitness) is put in procedure with PSO algorithm.

Block “CALCULATION END”:
- This block is reached if the conditions for end of calculation are filled.
- The results are saved in database.
- Program ends.

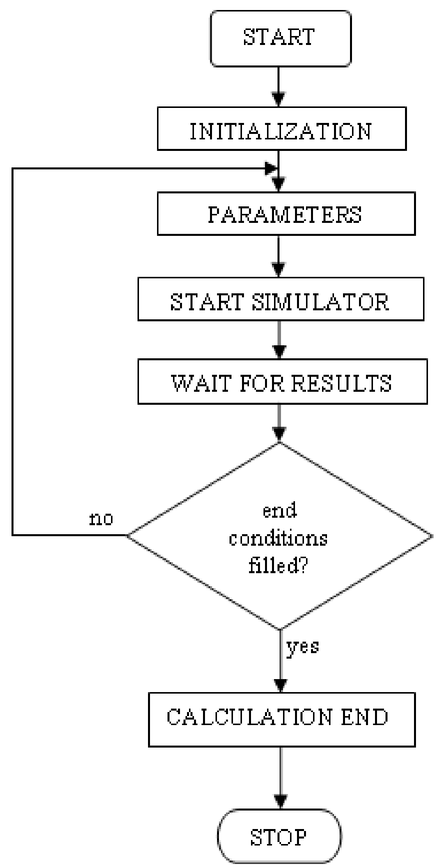

Fig. 1 Block diagram for application run

Notations for class SIMULATOR

NODE - a mark for node object in model;

first NODE - the node on the beginning of distribution network, usually the output point from heating station;

last NODE - the node on the end of distribution network, usually the input point on heating station;

input NODE - the node in which the section begins, is defined for each SECT;
output NODE - the node in which the section ends, is defined for each SECT;
SECT - a mark for section object in model;
first SECT - first section in distribution network, usually starts in first NODE
last SECT - last section in distribution network, usually ends in last NODE
next SECT - the next section in list of SECT, it not have to be necessarily a topologic consequential SECT
SIMULTIME - simulation time

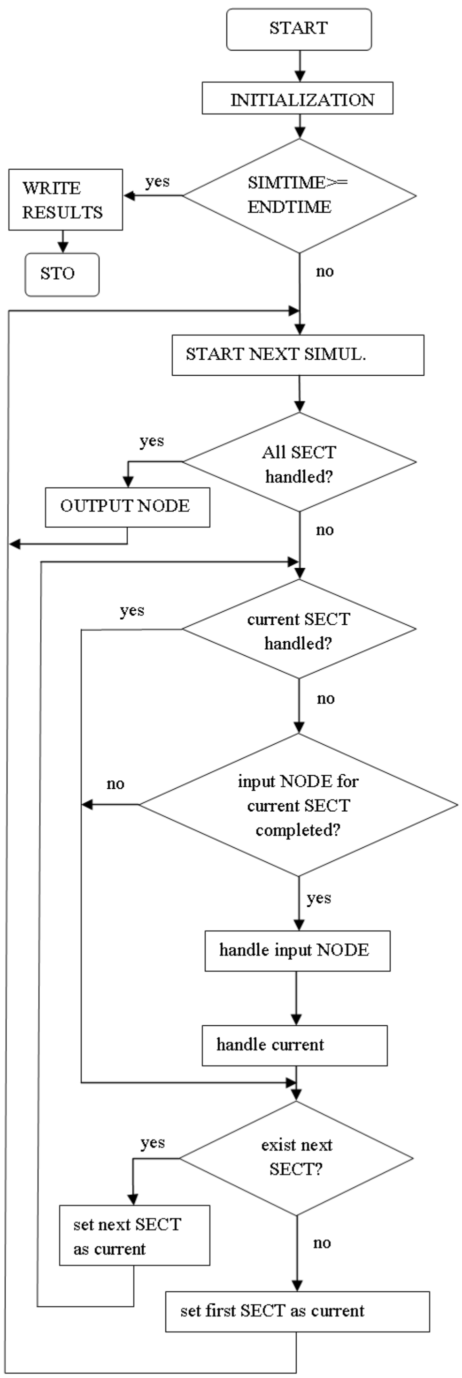

Fig. 2 Block diagram for class SIMULATOR

Short description of some selected block content:

Block “INITIALIZATION”:
- Set simulation time to start value.
- Create and initialize internal data structures for program modules.
- Create and reset data structures for results.

Block “START NEXT SIMUL STEP”:
- Add time interval Δt to the SIMULTIME
- Clear all marks “NODE handled” and “SECT handled”
- Create new DFQ. Its volume is calculated from current mass flow from heating station, its temperature is given by current temperature of water on output from heating station.
- Put this new DFQ into first NODE.

Block “NODE HANDLE”:
- Sum volume Vi of all DFQ’s (more exactly the parts of DFQ’s which reached the node) - volume V.
- Calculate temperature T of volume V
- Divide volume V to sections which are outputs from the node – volumes Vj
- Mark node as handled.
- Create new DFQ’s with volumes Vj, temperature T and put they into output sections for this node.

Block “LAST NODE”:
There are some similar functions as in other nodes.
Differences are:
- Write the output from this node into results
- Destroy input DFQ’s
- Don’t create new DFQ’s on the output from this node.

Block “input NODE complete ?”:
- The NODE is complete if all SECT, for which the NODE is output node, are handled

Block “SECTION HANDLE”:
- Calculate length of input DFQ from its volume and diameter of first pipe in section.
- Move all DFQ’s in section through all pipelines in section, i.e. calculate for each DFQ its new position. On calculation must be respect pass of DFQ’s over boundary lines between two consequential pipe lines and between last pipe line in section and output node – as described in chapter III.
- Calculate amount of heat and temperature change (decrease) for each DFQ in section based on heat transfer from DFQ to environment of pipe line corresponding to its current position (heating of space or energy losses).
- Mark section as handled.

Block “WRITE REUSULTS”:

- The results for each simulation step are written into internal data structures on the end of simulation step.

The results contents:

- Simulation time,
- input mass flow,
- temperature of input water,
- temperature of output water and
- other characteristics needed for the analysis.

A. *Particle swarm optimization*

After experiments with Differential Evolution, Self-Organizing Migrating Algorithm, Neural Network [12] and Levenberg–Marquardt algorithm, the Particle swarm algorithm was chosen as the numeric optimization algorithm suitable for problem without explicit knowledge of the gradient of function to be optimized.

PSO was first introduced in Kennedy & Eberhart, 1995 and was successfully applied on many optimization problems [7].

We use these PSO variant:

$$v'_{i,j} = \omega v_{i,j} + c_1 r_1 (global\ best_j - x_{i,j}) + c_2 r_2 (local\ best_{i,j} - x_{i,j}) \quad (3)$$

$$x'_{i,j} = x_{i,j} + v'_{i,j} \quad (4)$$

where:

n	is the number of particles, i = 1,…,n
m	is the dimension, j = 1, …, m
$x_{i,j}$	is the particle position
$x'_{i,j}$	is the updated particle position
$v_{i,j}$	is the particle velocity
ω	is the inertia component
c_1	is the social component
c_2	is the cognitive component
r_1, r_2, r_3	are uniform random numbers $\langle 0,1 \rangle$
$global\ best_j$	is the best global position
$local\ best_{i,j}$	is the best local particle position

The number of particles *n* we usually set two times more than dimension *m*. Inertia component ω is set about 0.8, social component *c1* is set about 1.4 and cognitive component *c2* is set about 0.6.

The fitness function is the minimum of the sum of squared residuals of measured and simulated return temperatures:

$$\sum_{i=0}^{n} (T_R measured(i) - T_R simulated(i))^2 \quad (5)$$

where: *n* is the number of samples.

1) *Stopping criterions*

We use MaxDistQuick as a stopping criterion. The optimization is stopped if the maximum distance of the major part of particles is below a threshold *eps* or the maximum number of iteration is reached:

1) Particles are sorted by the means of their quality (value of fitness function).
2) The subset of best *n* particles is chosen.
3) The Euclidean distance from best particle is estimated for each particle in subset.
4) Algorithm stops if the maximum distance in subset is below the threshold *esp*.

2) *Implementation*

PSO is implemented in JAVA language in this main function structure:

- Initialization - this function initializes all parameters and runs only once at the start of the algorithm.
- Update particles positions - this function calculates new positions of particles and returns true if the algorithm should stop.
- Get updated position - this function returns positions of one particle that should be evaluated together with particle number.
- Set fitness function value for particle – this function receives value of the fitness function and Paris this value by the means of particle number with particle.

This solution enables parallel implementation of PSO algorithm (Figure 2). There is a peer application that runs simulations in threads and runs PSO functions [7].

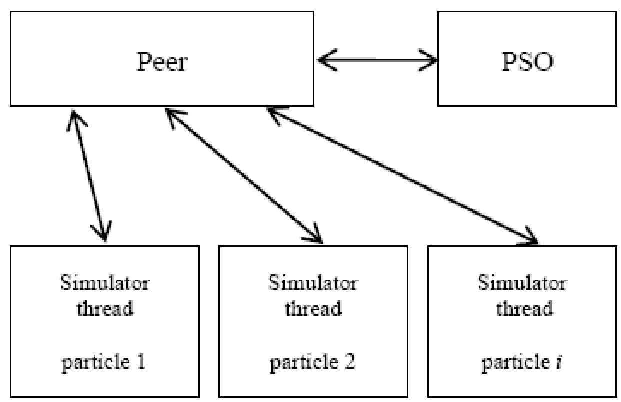

Fig. 3 Parallel implementation

Object relation data model

B. *Object relation data model*

For our model the object-relation features of Oracle 11g database were used. Our Oracle database contains structure of heat distribution network, real measured data and also simulated data computed by java application.

In our case we model distribution network objects such as nodes, pipelines, pipeline sections, energy consumers, energy sources and others. Oracle introduces abstract layer built on Oracle database’s relational technology. New object is based on any standard data type embedded in Oracle (FLOAT, NUMBER, DATE, etc.) or on another object or reference on object or reference on object collection. User defined metadata types are stored in schemes and are accessible through SQL, PL/SQL, Java or other supported interface. The data under the object layer is stored in columns and tables, but because of object approach we can use them as real-world entities. Using objects has following benefits [11]:

- developers access data structures used in application directly without object relational mapping (ORM)
- objects contain data and can have defined user operations (methods)
- using objects is more effective:
  - object types, methods and data are stored in database therefore developers don't have to create them in each application
  - related object are processed in one step. Single query on database returns also objects linked with searched object

Object types are composed from parts, called attributes and methods (illustrated in fig. 4):

- **attribute** stores information about main features of object. The attribute is declared by data type (this can be another object)
- **method** is a procedure or function providing useful operation over attributes [11].

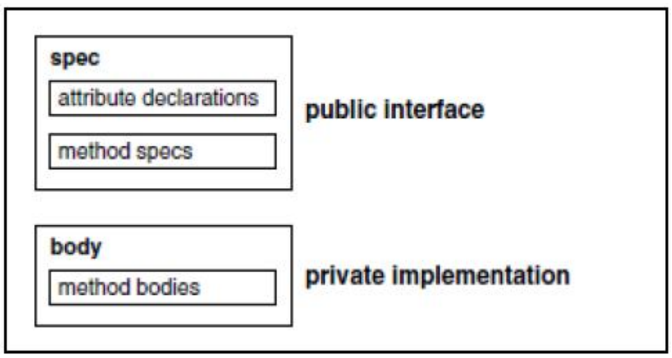

Fig. 4 Parallel implementation [11]

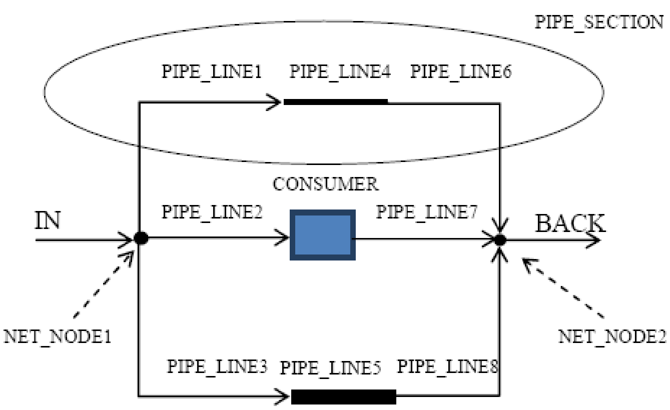

Fig. 5 Relation among objects [11]

Figure 2 shows relations among objects in distribution network. IN and BACK mean input and output places where heat transfer medium (liquid which supply or carry energy to consumer) enters and leaves particular part of network [11].

IV. Model Use and Applicability

It is expected that the proposed simulation model will be used in the control system SHDC for the following purposes:

- Identification of model parameters for the selected time period
- Prediction of appropriate timing of the supplied amount of heat energy for the next period.

A. Identification of model parameters for the selected time period

As mentioned, essential for the modeling approach to SHDC is to determine the function *sr(...)* used in equations (6). This means that it is necessary to choose the appropriate form of parametric functions and find values of the parameters for the given conditions.

The procedure will be described in detail on simple example where the function *sr(...)* shall only affect consumption fluctuations during the day. We will therefore assume that the function *sr(...)* will have the form.

$$s_r(\ldots) = \lambda_r * ({}^{i}T_j - T_{ext}) * k_h \tag{6}$$

where:

- λ_r is the coefficient of heat transfer in segment *r* (here we suppose that the segments are pipe sections as well as consumer units, depending on the value of the coefficient λ),
- ${}^{i}T_j$ is the current temperature DFQ_i for the particular simulation step *j*,
- T_{ext} is the current outside temperature and
- k_h is coefficient which corrects heat consumption oscillations during a day.

To determine searched values k_h (points of timeline) is possible to use several methods based on principles allowing us to find a function(s) which should have the best course approximating analyzed variables. One option is for example to use genetic algorithms. In the presented solution was the method PSO (Particle Swarm Optimization) used – described in chapter II.A. This method has been lately compared with other methods, such as SOMA, neural networks [12] and Levenberg-Marquard algorithms for nonlinear methods of least squares. It was found that the results achieved in terms of accuracy and speed of convergence is similar. PSO is therefore comparable for the determination of the correction factors and we use it.

B. Prediction of appropriate timing of the supplied amount of heat energy for the next period

The individual steps of the proposed procedure are as follows:

- Prediction of conditions for a selected future time interval of the SHDC control.
- Seeking the time interval with similar conditions.
- Identification of model parameters for time period with similar conditions.

- Identification of model parameters for previous time period.
- Calculation of control actions for the selected time interval

Implementation of these particular steps is described in [7] as well as test results. Remaining part of this paper introduce above mentioned steps in practical experiment as already advised.

V. THE EXPERIMENT

This chapter shows two days experiment performed on heating system of midsize city with more than hundred heat exchangers. The experiment was conducted from March 8th to March 10th. Its main purpose was to predict sequence of heating water temperatures (*Tv*) to control quality of heat supply [1].

The first step of experiment was to prepare simplified pipe model. Because there were not enough information about heat consumption spreading, the city for model purpose were divided into twelve heat consuming spots which represents group of heat exchangers with similar distance from the heating plant. Also all spots have the same power requirements.

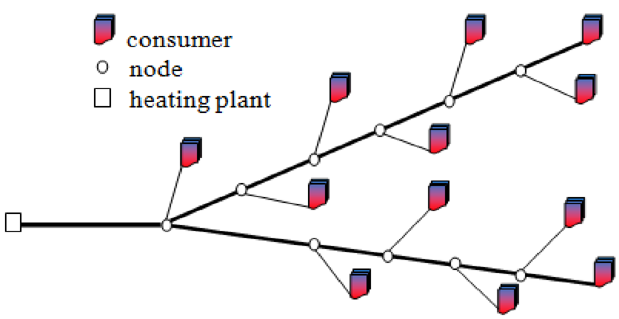

Fig. 6 Simplified diagram of the network model

A. Identification

The next step was to set up model for identification. According to unknown condition of day in future a day with similar outside temperature from the past were chosen to train the model [3]. These steps provide resources for identification and expected heat consumption were calculated.

Individual parts of identification process are shown further.

The experiment aimed to predict values of heating water temperature necessary to satisfy city consumption. The experiment was to begin on 8th of March 2 PM. To train a model at least two extra hours are necessary, so at this time all data were updated. Interval to predict was set up to 30 hours and the time to initialize process to 36 hours. Base on the similar day principles [3], similar day was selected; see fig. 11 for its conditions.

The identification process focuses on heat consumption, where two sets of data are sought:

Initialization set – this described immediately preceding behavior of the system consumption. As described earlier, the model adapts its parameters and focused on the minimum deviation between the measured and calculated water temperature in return line. Deviations achieved for identified system are shown fig 7.

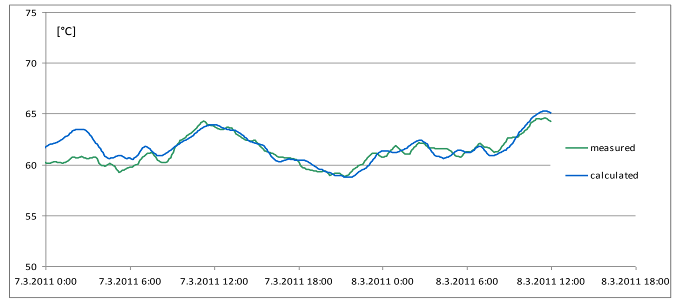

Fig. 7 Deviation between the measured and calculated temperatures in return line (initialization set)

Identified model is able to shows heat consumption distributed in time. See fig 8.

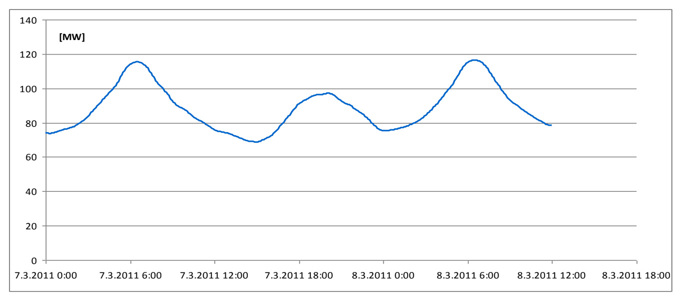

Fig. 8 Heat consumption distributed in time for initialization set

Expected set – this data describes expected behavior of the system. The prediction mechanism meet the needs of heat consumption in time, so as well as initial set, the expected behavior needs to be identified. Similar day mechanism [3] selected 26th of February as the best match (based on weather forecast). The adaptation mechanism of the model takes in account the initialization set parameters and extends its parameters - again focusing on the minimum deviation between the measured and calculated water temperature in return line. Deviations achieved for identified system in expected set are shown in fig 9.

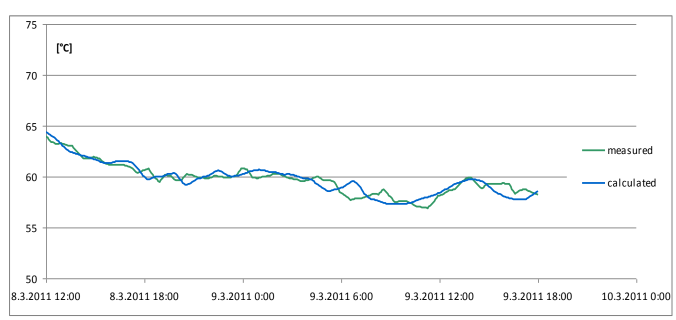

Fig. 9 Deviation between the measured and calculated temperatures in return line (expected set)

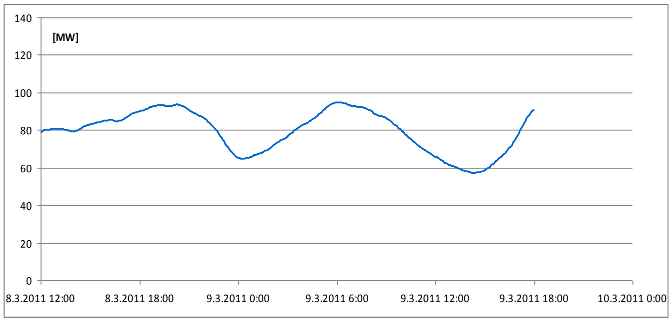

Fig. 10 Heat consumption distributed in time for expected set

B. *Prediction*

Obtained heat requirements (initialization and expected consumptions set) are used in prediction process. Predicted values of heating water, together with identified consumption form the basis for the computation of the remaining values. This process and obtained results will be explained.

1) *Conditions for day to predict*

The only weather forecasts are known, for the days to predict. The figure 11 shows the forecast for days to predict (blue curve). To assess the result the figure 11 also shows the measured temperature (red line). Of course this temperature was obtained when the day passed. As explain earlier, the identification process is based on condition of similar day [2] so the outside temperature of these days is presented on figure 11 as well (green line). The days selected as similar ware 26th February and 5th of March.

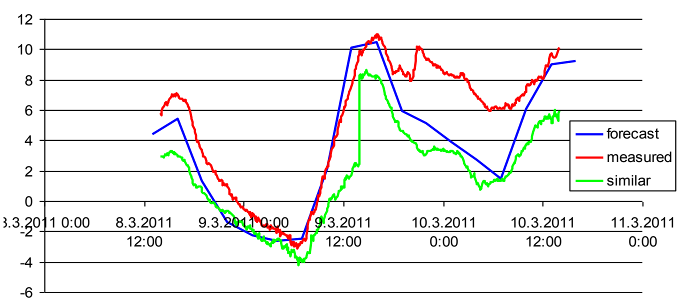

Fig. 11 Outside temperatures significant for the experiment

Based on the heat requirements, the prediction mechanisms took the place. The sequence of *Tv* were predicted, see fig. 12. These *Tv* were than imposed into the heating plant system and real control took the place.

The next day was model updated and all steps were repeated for subsequent time period. The new prediction was prepared from 10 AM, to update last hours of previous one. See fig. 13.

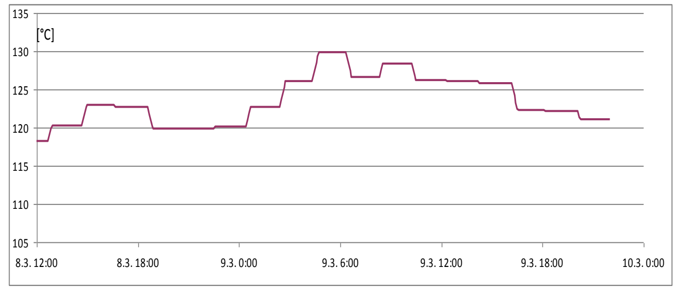

Fig. 12 Proposed values of Tv for first day (water in the supply line)

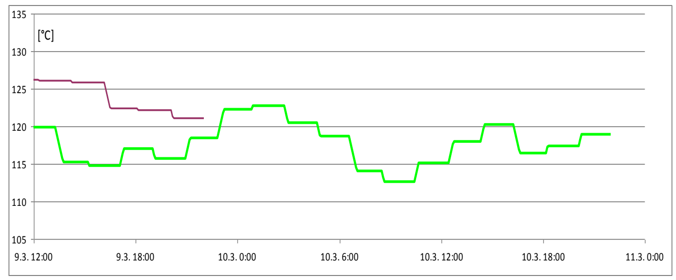

Fig. 13 Proposed values of Tv for second day

C. *Evaluation of experiment*

The proposed values were found acceptable by the heating plant authorities and followed for both days. Later, prediction from model was compared with the measured results. Results are shown on the following pictures.

Graph legend:
____ first day prediction
____ second day prediction
____ measured course

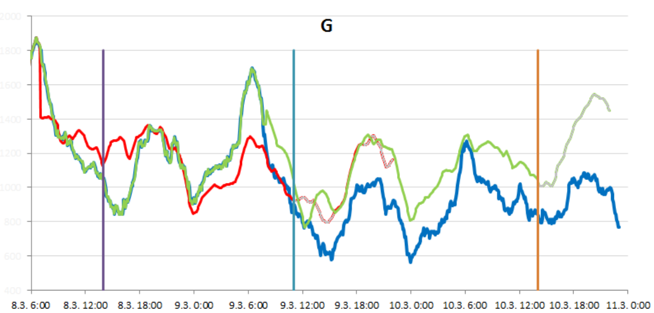

Fig. 14 Flow (transferred fluid)

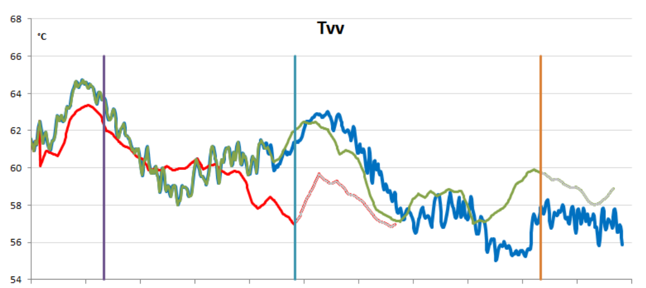

Fig. 15 Returned water temperature (water in the return line)

VI. Conclusion

The results obtained during the model verification, tested on the real measured data shown that the proposed simulation model is a well suited tool for analyzing the properties and behavior of SHDC.

Introduced simulation model has been subjected to real heating plant system. The results obtained in this case show that model perform well in a real situation as well, however several insufficient remain. As can be seen on fig. 14 the flow predicted and measured course have considerable deviation. This is probably due to inappropriate binding between similar and examined days. To eliminate this insufficient, the current research continues.

Nevertheless, used methods and algorithms appear to be leading to improved performance of existing heating systems.

Acknowledgment

The work was performed with financial support of research project NPVII-2C06007, by the Ministry of Education of the Czech Republic and is also supported by the European Regional Development Fund under the Project CEBIA-Tech No. CZ.1.05/2.1.00/03.0089.

References

[1] Balate, J. Automatic Control, 2nd edition, BEN - technical literature, ISBN 80-7300-148-9, Prague, 2004.
[2] Dolinay V., Vasek L., Kral E. Heat Supplies Prediction Based on Similar Days. In Annals of DAAAM for 2010 & Proceedings of the 21st International DAAAM Symposium. Vienna : DAAAM International, pp. 227-228. ISBN 978-3-901509-73-5, ISSN 1726-9679, 2010.
[3] Dolinay, V., Vasek, L., Municipal Heating Network Simulation Experiments Based on Days with Similar Temperature, In procceedings of the 10th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications (EHAC'11), Cambridge 2011, pp.318-320, ISBN: 978-1-61804-004-6.
[4] Dolinay V., Palka J. & Vasek L., (2010), Sunny Day as Sharping Factor of Heat Consumption Survey, Proceedings of 14th WSEAS International Conference on Systems, Corfu Island, Greece, July 22-25, 2010, pp. 268-270.
[5] Incropera, F. P.; DeWitt, D. P. & Bergman, T. L. Introduction to Heat Transfer, Wiley, ISBN: 978-0471457275, London, 2006.
[6] Kalos, A., (2005). Discrete Particle Swarm Optimization for Automating the Design of Multivariate Autoregressive Neural Networks, WSEAS Transactions on Information Science and Applications, Vol. 2, No. 11, Nov. 2005, pp. 1807-1815.
[7] Kral E., Vasek L., Dolinay V. & Varacha P. Usage of PSO Algorithm for Parameter Identification of District Heating Network Simulation, Proceedings of 14th WSEAS International Conference on Systems, Corfu Island, Greece, pp. 657-659, 2010.
[8] Kral, E., Vasek, L., Dolinay, V., Capek, P., Usage of Peak Functions in Heat Load Modeling of District Heating System, In procceedings of the 10th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications (EHAC'11), Cambridge 2011, pp.404-406, ISBN: 978-1-61804-004-6.
[9] Law A.M., Kelton W.D. Simulation Modeling and Analysis, McGraw-Hill, ISBN: 978-0070592926, 1999.
[10] Li, J.; Poulton, G.; James, G.; Zeman, A.; Wang, P.; Chadwick, M. & Piraveenan M. R., (2007). Performance of Multi-agent Coordination in Distributed Energy Resources, WSEAS Transactions on Systems and Controls, Vol. 2, No. 1, Jan. 2007, pp. 52-58.
[11] Palka, J., Vasek, L., Dolinay, V., Object relation data model of heat distribution network, In procceedings of the 10th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications (EHAC'11), Cambridge 2011, pp.426-428, ISBN: 978-1-61804-004-6.
[12] Varacha, P. Impact of Weather Inputs on Heating Plant - Agglomeration Modeling. Proceedings of the 10th WSEAS International Conference on Neural Networks, Mastorakis, N., pp. 159-162, ISBN 978-960-474-065-9, Prague, 2009.
[13] Vasek L., Dolinay V. Simulation Model of Heat Distribution and Consumption in Municipal Heating Network. International journal of mathematical models in applied sciences, <http://www.naun.org/journals/m3as/>. ISSN 1998-0140, 2010.
[14] Vasek, L., Dolinay, V., Simulation Model of Heat Distribution and Consumption in Practical Use, In procceedings of the 10th WSEAS International Conference on Electronics, Hardware, Wireless and Optical Communications (EHAC'11), Cambridge 2011, pp.321-324, ISBN: 978-1-61804-004-6.