Determination of Optimal Parameters of Heating Systems Based on Advanced Information Technologies

Valery A. Stennikov, Evgeny A. Barakhtenko*, Dmitry V. Sokolov

Melentiev Energy Systems Institute Siberian Branch of the Russian Academy of Sciences, Irkutsk, Russia

***Abstract* — This paper presents a new method for the development of software to determine optimal parameters of heating systems. The method is based on the Model-Driven Engineering paradigm. The essence of this paradigm is that the software is generated on the basis of formal descriptions represented by models. This method makes it possible to automate the process of software development. The ontologies of heating systems, problems, and software are a means of representing the models. The paper proposes metaprogramming to make the software architecture flexibly adjustable to the problem of parameter optimization of a concrete heating system in the course of the problem-solving process. Metaprogramming technologies enable the development of software to change or create software components when solving the problem. The proposed method includes four stages: 1) development of a computer model of the heating system; 2) formalization of the applied problem; 3) automatic construction of the software model; 4) automatic development of the software on the basis of the model. This method underlies the SOSNA software intended for solving parameter optimization problems of heating systems. The software makes it possible to calculate large-scale systems with a complex structure with any set of nodes, sections, and circuits. The use of the software to control the expansion of heating systems will enhance their energy efficiency and cost-effectiveness. The software was applied to solve the optimal reconstruction problems of urban heating systems.**

***Index Terms* — Model-Driven Engineering, metaprogramming, ontology, software engineering, heating system, optimization methods, nonlinear programming.**

* Corresponding author.
E-mail: barakhtenko@isem.irk.ru

http://dx.doi.org/10.25729/esr.2018.01.0010
Received: February 26, 2018. Revised: April 6, 2018. Accepted: April 15, 2018. Available online: April 25, 2018.

I. Introduction

In the harsh Russian climate, heating systems are essential for the society and economy. Today's heating systems have turned into complex spatially distributed pipeline systems of district heating systems. Their complexity is related to their closed two-line schemes, a multi-loop structure, a great number of heat sources and control components (pumping stations, pressure and flow rate regulators, heat points). Nowadays, the requirements of efficiency, quality of heat supply to consumers and reliability of heating systems are increasing. Such requirements generate the need to develop and apply effective methods and software to design these systems.

Therefore, the determination of optimal parameters is crucial to ensure the efficient operation of heating systems. This task can be considered either individually or as a subproblem of a general process aimed at design of the heating system [1,2]. Large sizes of heating systems and computational complexity of applied models, methods and algorithms do not allow the determination of the optimal parameters of heating system without specialized software.

The paper presents the experience of applying the concept of Model-Driven Engineering (MDE) to create the next generation software to determine optimal parameters of heating system. This concept represents a set of methodological approaches to the automated construction of complex software systems based on the models developed in advance [3,4]. The MDE concept was used to implement the SOSNA (Synthesis of Optimal Systems in terms of Reliability) software. The software is intended for the determination of optimal heating system parameters. The development of the SOSNA software is automated. It is based on a virtual model of the heating system at issue, a formal description of the problem to be solved and knowledge on the subject domain, which are stored as ontologies [5,6]. The method proposed in the paper is universal and can be applied in the development of software for solving a wide range of problems related to design of heating systems and other pipeline systems.

The software is developed based on the advanced metaprogramming technologies [7]. The metaprogramming technologies make it possible to adjust the software to calculation of systems with different types of equipment (pipes made of steel and metal plastic, of various diameters, pumps, etc.) and different ways of its installation.

II. Determination Of Optimal Heating Systems Parameters And Methods To Solve This Problem

The optimization problem of heating system parameters is stated as follows. The set parameters include: 1) a heat network scheme consisting of m nodes and n branches, represented by directed graph $G_{\text{hss}} = (J, I)$, where J – a set of vertices (nodes), I – a set of edges (branches); $J = J_{\text{c}} \cup J_{\text{s}} \cup J_{\text{c}}$, where J_{c}, J_{s} and J_{c} – sets of consumers, sources and branching points in the scheme, respectively; $I = I_{\text{p}} \cup I_{\text{s}} \cup I_{\text{c}}$, where $I_{\text{p}} = I_{\text{e}} \cup I_{\text{n}}$ – a set of branches in the linear part of the network consisting of sets of existing I_{e} and new I_{n} branches; I_{c} and I_{s} – sets of consumer-branches and source-branches, respectively; $I_{\text{ps}} \subset I_{\text{s}} \cup I_{\text{p}}$ – branches on which pumping stations (PS) are either installed or allowed; 2) pipeline lengths in the network branches L_i, $i \in I_{\text{p}}$; 3) a set of standard pipeline diameters D; 4) a set of numbers of all pipeline construction (reconstruction) ways possible for the network U; 5) sets of numbers of pipeline construction (reconstruction) permissible for each branch $U_i \subset U$, $i \in I_{\text{p}}$; 6) lower ($P_j^{\min}$, $j \in J$) and upper ($P_j^{\max}$, $j \in J$) pressure constraints; 7) lower ($v_i^{\min}$, $i \in I_{\text{p}}$) and upper ($v_i^{\max}$, $i \in I_{\text{p}}$) constraints on heat carrier flow speed in branches; 8) vector of nodal heat carrier sinks and sources $\mathbf{Q} = (Q_1, \ldots, Q_m)^{\text{T}}$; 9) a set of heads at pumping station T; 10) constraints on the minimum available head at consumers' $\Delta H_i^{\min}$, $i \in I_{\text{c}}$.

By solving this problem we will determine optimal parameters of heating systems: 1) vector of pipeline diameters $\mathbf{d} = (d_1, \ldots, d_n)^{\text{T}}$; 2) vector $\mathbf{u} = (u_1, \ldots, u_n)^{\text{T}}$ with its components ($u_i \in U_i$, $i \in I_{Л}$) containing the numbers of optimal ways of pipeline construction (reconstruction); 3) vector of heads at pumping station $\mathbf{H} = (H_1, \ldots, H_n)^{\text{T}}$; 4) vector of heat carrier flow rates in branches $\mathbf{x} = (x_1, \ldots, x_n)^{\text{T}}$; 5) vector of pressures at nodes $\mathbf{P} = (P_1, \ldots, P_m)^{\text{T}}$.

It is necessary to minimize the total cost function that has the following form:

$$Z(\mathbf{d},\mathbf{u},\mathbf{H},\mathbf{x},\mathbf{P}) = \sum_{i \in I_{\text{p}}} Z_i^{\text{p}}(d_i, u_i) + \sum_{i \in I_{\text{ps}}} Z_i^{\text{ps}}(H_i, x_i) + \\ + \sum_{i \in I_{\text{p}}} Z_i^{\text{e}}(x_i, d_i) + \sum_{i \in I_{\text{p}}} Z_i^{\text{h}}(d_i) + \sum_{\substack{i \in I_{\text{c}} \\ \varphi(i)=(j,k)}} Z_i^{\text{c}}(x_i, P_j, P_k) \to \min, \quad (1)$$

where Z_i^{p} – costs of pipeline construction and operation in the network branch; Z_i^{ps} – construction and operation costs of pumping station; Z_i^{e} – cost of electricity used to pump a heat carrier; Z_i^{h} – cost of heat losses; Z_j^{c} – cost of electricity used to supply heat carrier to consumer; $\varphi(i)$ – function associating each branch $i \in I$ with a pair of incident nodes (j, k).

The model of flow distribution in heating system has the following form:

$$\mathbf{Ax} = \mathbf{Q}, \quad (2)$$

$$\mathbf{f}(\mathbf{x},\mathbf{s}) = \mathbf{A}^{\text{T}}\mathbf{P} + \mathbf{H}, \quad (3)$$

where $\mathbf{A}$ – the node-branch incidence $m \times n$-matrix of the calculated scheme; $\mathbf{s} = (s_1, \ldots, s_n)^{\text{T}}$ – vector of hydraulic resistances of branches with components $s_i = s_i(d_i)$, $i \in I$; $\mathbf{f}(\mathbf{x},\mathbf{s})$ – n-dimensional vector function with components $f_i(s_i, x_i) = s_i x_i |x_i|^{\beta - 1}$, $i = \overline{1,n}$, that reflect the laws of pressure drop in the network branches, β – exponent of power depending on pipeline type and heat carrier flow.

The system of conditions and constraints includes:

– a constraint on pressure at nodes

$$P_j^{\min} \le P_j \le P_j^{\max}, \ j \in J; \quad (4)$$

– a constraint on the heat carrier flow speed in the branches

$$v_i^{\min} \le v_i(x_i) \le v_i^{\max}, \ i \in I_{\text{p}}; \quad (5)$$

– a condition of discreteness of pipeline diameters

$$d_i \in D, \ i \in I_{\text{p}}; \quad (6)$$

– a condition of discreteness of pipeline reconstruction types

$$u_i \in U_i \subset U, ; \quad (7)$$

– a condition of discreteness of pumping station heads

$$H_i \in T, \ i \in I_{\text{ps}}; \quad (8)$$

– a constraint on available head at consumers

$$P_j - P_k \ge \Delta H_i^{\min}, \ \varphi(i) = (j,k), \ i \in I_{\text{c}}. \quad (9)$$

The minimization problem of function (1) subject to (2) - (9) is solved to determine the optimal parameters of the heating system.

Based on the theory of hydraulic circuits that was developed at Melentiev Energy Systems Institute SB RAS, we have devised effective methods for determining optimal parameters of heating systems. For branched networks, the method of step-by-step optimization was based on dynamic programming [1, 2]. For ring (multiloop $i \in I_p$) networks, we developed a multiloop optimization method based on the principle of successive improvement in solution [8, 1, 2]. An important characteristic of the enumerated methods is that they make it possible to flexibly adjust the computational procedure to the mathematical models of the used set of equipment.

Additive objective function (1) allows the application of dynamic programming for the determination of optimal parameters of branched heating systems: for the networks of such systems, the flow rates in branches $\mathbf{x}$ are uniquely determined by the tree-like structure and nodal flow rates $\mathbf{Q}$. The computational procedure based on dynamic programming suggests determining the parameters of network components (branches and nodes) by their successive fitting in the direction from consumers to sources [1,2].

The idea of the multiloop optimization method lies in decomposition of a complex optimization problem of ring heating system into two less complex subproblems [8,2]:

1) Determination of optimal parameters (diameters $\mathbf{d}$ and heads $\mathbf{H}$) of a network tree by dynamic programming method at fixed flow rates $\mathbf{x}$;

2) Calculation of flow distribution in a ring network (flow rates $\mathbf{x}$ and pressure $\mathbf{P}$) at fixed diameters $\mathbf{d}$ and heads $\mathbf{H}$.

The method of multiloop optimization suggests an iterative computational process aimed at successive improvement in the heating system parameters, during which the enumerated subproblems are alternately solved (Fig. 1). A stopping criterion for the computation is the termination of a decrease in the objective function or coincidence of pipeline diameters in neighboring iterations. Thus, by successively solving the subproblems at issue we will arrive at a solution that cannot be improved by the method of multiloop optimization.

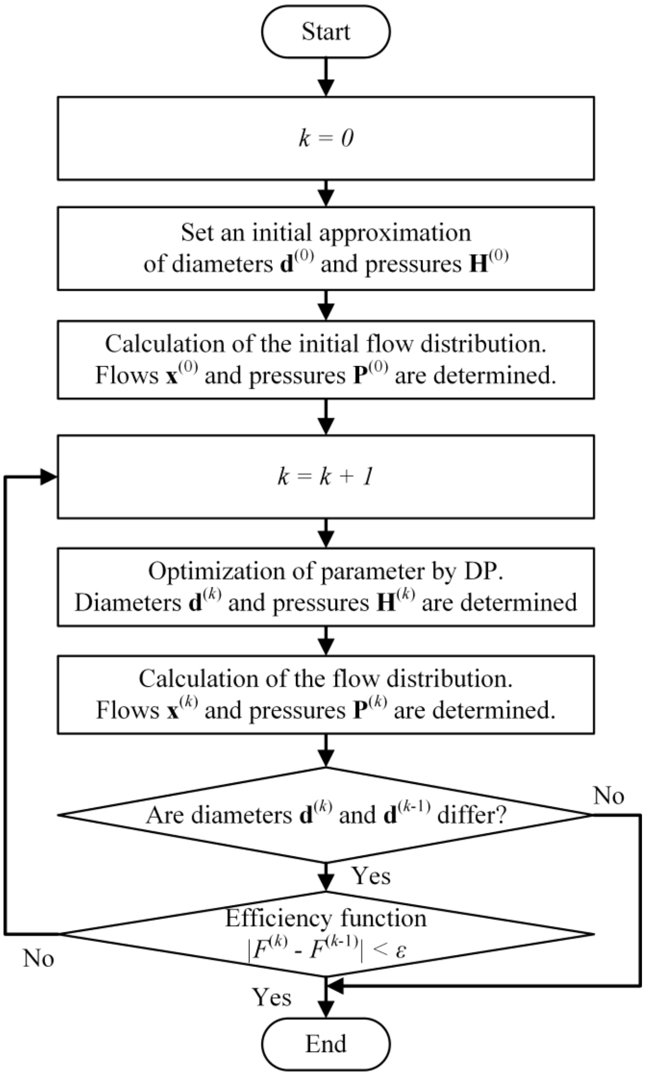

Figure1. A block-diagram of an algorithm for the determination of optimal parameters of a ring heat network by the multiloop optimization method, where N – iteration number; ε – set accuracy of the computational process

The calculation of flow distribution in the heating system is reduced to solving a system of linear and nonlinear algebraic equations. There are two main forms of such systems: with nodal flow distribution and loop flow distribution [1,2]. The problem statement in the case of nodal flow distribution implies the following. The specified parameters are: incidence matrix $\mathbf{A}$, vectors of nodal flow rates $\mathbf{Q}$ and heads $\mathbf{H}$, vector function of hydraulic characteristics $\mathbf{f}(\mathbf{x},\mathbf{s})$ and pressure P_m' at node m. It is necessary to determine the vector of flow rates in branches $\mathbf{x}$ and pressure at nodes $\mathbf{P}$, that satisfy the system of equations (2)-(3). This problem is solved by the pressure-based method [1, 2]. In the case of the loop flow distribution, the problem is solved by the method of loop flow rates [1, 2].

III. Literature Review

The development of method and software is based on the advanced metaprogramming technologies. Metaprogramming represents a programming technique capable to create programs that generate other programs when running, or programs that change themselves during execution. In Russia, the fundamental research in this area was conducted by A.P. Ershov. In 1958, he published his monograph "Programming Programme for the BESM Computer" [9]. The researchers from other countries are [7, 10, 11], to name but a few.

The automatic construction of a software system involves ontologies that make it possible to formalize the description of objects of a subject domain, their properties, and interactions among these objects. Currently, the ontologies are finding increasingly wider application in engineering [12]. Some researchers describe the experience of applying ontologies to solve power engineering problems [13, 14]. General issues related to the development and use of ontologies are discussed in [15, 16].

We have made an overview of the existing software designed to model pipeline systems and solve other energy problems. Further, a description of some software systems based on advanced information technologies is presented.

The EPANET software designed by US EPA (United States Environmental Protection Agency) offers good possibilities for modeling water pipeline systems [17]. It includes a graphical subsystem enabling work with the model of pipeline system and a set of software modules intended for solving applied engineering problems. Graphical interface is implemented in Delphi, the software modules are implemented in C++ and are connected to the main program as Dynamic Link Libraries. The software allows modeling of hydraulic conditions in the network, given all their characteristics. The program (software) includes the hydraulic analysis algorithms capable to analyze networks of any size without limitations, calculate head losses due to friction, by using the Hazen-Williams, Darcy-Weisbach and Chezy-Manning equations, and calculate head losses in branches and junctions. EPANET can also be used to simulate the application of pumps, calculate power consumption by pumps and costs, simulate different types of valves and throttles.

The Thermoflow software was designed by a group of authors headed by Maher Elmasri. The software consists of several modules that make it possible to solve a wide range of problems arising in heat power industry. The software ESteam developed by Veritech has a block structure and is capable to develop flow charts of steam turbine, gas turbine and combined cycle plants that can be used to conduct a variety of studies, introduce changes in the flow chart, represent grouped results in different ways, construct diagrams, etc. [18].

The heating systems can be designed with the software provided by different developers, for example, software from Polyterm Ltd, “Golfstream” from TIC “Sibnefteprodukt”, WADSOP, ArcView GIS, etc.

The Melentiev Energy Systems Institute SB RAS has developed original methods and software implementing these methods. This software is called a system for machine program building. This system, using the data on the flow chart components and links among them, can generate a program for calculation of a complex thermal facility [19].

The ANGARA information-computation environment developed at Energy Systems Institute SB RAS by A.V. Alexeyev, N.N. Novitsky, and V.V. Tokarev to solve the problems within the theory of hydraulic circuits involves a metadatabase for flexible access to computation modules and databases [20]. The ANGARA environment can be used to implement a technology for the end-to-end modeling of pipeline systems of different types and purposes (heat supply, water supply, etc.) with a single graphical user interface. This advanced and sought-after technology makes it possible with one software tool to obtain results relating solutions to the following tasks: 1) design; 2) calculation of states; 3) dispatching control. The process of solving the enumerated problems requires effective methods and algorithms implemented as software modules. The Angara environment ensures fast connection of the required modules, which allows flexible organization of computational processes and solving problems of any complexity.

Currently, a great number of IT specialists are investigating the methods and technologies for construction of complex software systems. These include but are not limited to: [21, 22, 23] and [9, 24].

It is also worthwhile to draw attention to the research

[25], conducted at Melentiev Energy Systems Institute. It is focused on the information technologies applied in systems studies in the field of energy and proposes an original approach to the integration of these technologies in the framework of a single software package.

The general issues of development and analysis of the algorithms are considered in the studies presented in [26-31].

Based on the literature review and an analysis of the existing software and approaches to its implementation, we can conclude that the issues addressed in the paper are of great importance.

IV. Description Of The Problem And Setting The Research Objectives

Evolution of the market for heat pipelines, equipment and technologies applied for heating system construction considerably expands the possibility for the implementation of various technical solutions. The expansion of the possibilities, in turn, requires their consideration when determining optimal parameters of heating systems, since each type of the applied equipment has its distinctive characteristics and is represented by its set of mathematical models defining its parameters and techno-economic relationships.

Concrete mathematical models for the calculation of the components of cost function (1) (Z_i^{p}, Z_i^{ps}, Z_i^{e}, Z_i^{h}, Z_i^{c}) and hydraulic characteristics of branches (s_i, $f_i(s_i, x_i)$, $v_i(x_i)$) are chosen depending on the properties of the used equipment (material, composition, way of construction, operating time, etc.).

We will bring several examples of mathematical models applied to determine optimal heating system parameters.

Function $f_i(s_i, x_i)$, showing the law of pressure drop in the network branch for a steel pipeline has the following form ($\beta = 2$):

$$f_i(s_i, x_i) = s_i x_i |x_i|.$$

Hydraulic resistance of branch s_i, where the steel pipeline is laid can be calculated by the formula

$$s_i = \lambda_i \frac{(1+\alpha_i)L_i}{d_i^{5.25}},$$

where λ_i – coefficient, depending on equivalent roughness of the branch pipeline; α_i – coefficient of local losses in the branch.

Costs of construction and operation of new pipeline Z_i^{p} are calculated by the equation

$$Z_i^{\mathrm{p}}(d_i) = (a + f_{\mathrm{p}})K(d_i)L_i,$$

where a – discounting coefficient; f_{p} – a share of depreciation charges for the pipeline; $K(d_i)$ – specific capital investment in the pipeline.

Melentiev Energy Systems Institute SB RAS has been developing software for the determination of optimal parameters of heating systems for 40 years [1,2]. The software modules of the previous generation that implement the methods and algorithms to cope with this task are intended for a certain set of equipment which considerably complicates their adjustment to the calculation of networks with some other set of equipment. Therefore, the need has arisen to develop and apply new adaptive approaches to adjust the software to modeling of real heating systems with a large set of equipment involved. The implementation of the next generation software for the determination of optimal heating system parameters calls for new methods capable to flexibly adjust software to specific features of expansion and characteristics of equipment of the studied heating system.

V. Methods For Model-Driven Engineering Of Software

Based on the conducted research, we propose a new method for model-driven software development based on the MDE conception [32]. This method allows automated software construction both to determine optimal parameters of heating systems and to cope with the other related tasks of heating system design. Let us present the main principles of this method.

1. Integration of a virtual model of a specific heating system, models of heating system components (i.e. individual subsystems) and methods (algorithms) is performed only in the context of an applied problem.
2. Software components depending on the properties of the modeled heating system (a set of models of heating system components) are developed automatically by using metaprogramming based on the virtual model of this heating system and ontologies that contain the description of equipment and applied problems to be solved.
3. The software intended for modeling a certain heating system is designed in the context of an applied problem with the help of metaprogramming, based on automatically constructed software models of heating system components, and components implementing the methods and algorithms for solving the applied problems. The process of software construction is controlled by the knowledge stored in ontologies: a heating system ontology, a problem ontology and a software ontology.

XML language [5] is used as a tool for formal representation of ontologies. This language was used to develop the domain-oriented language. MathML (Mathematical Markup Language. Site of the project: http://www.w3.org/Math/) is applied for storage of mathematical models of the heating system components.

The heating system ontology consists of: a description of a hierarchical structure of heating system; a classification of the equipment used in the heating networks and its technical characteristics; a description of heating system components and their parameters (technical characteristics, hydraulic parameters and boundary conditions); and a classification and description of the applied mathematical models (for example, equations defining the laws of head loss in the network branches, formulas for calculation of resistance, etc.). Figure 2 presents a fragment of the heating system ontology.

The problem ontology contains a description of the applied problems (for example, optimization of multiloop network parameters) and methods for solving them (for example, the multiloop optimization method), a description of algorithms, enumeration of parameters that represent initial data and parameters obtained by solving the problem.

The software ontology is intended for the storage of knowledge necessary to automate the construction and use the software. This ontology contains a description of software components, and their properties, metadata (input and output parameters, description of data formats), a description of technologies and interfaces for access to the software components.

The developed approach is intended for the use of Java [33] as the main language and programming platform.

The technique of model-driven software development includes four stages. A scheme demonstrating the interaction among the stages of this technique is presented in Fig. 3.

A. Stage 1. Construction of a virtual heating system model

In this stage, an engineer constructs a virtual hierarchical model of a specific heating system. This model represents a directed graph with its vertices corresponding to nodes (sources, consumers, connection nodes) and edges corresponding to branches (passive branches, active branches with pumping stations). For each component of the model, there is information on the installed equipment,

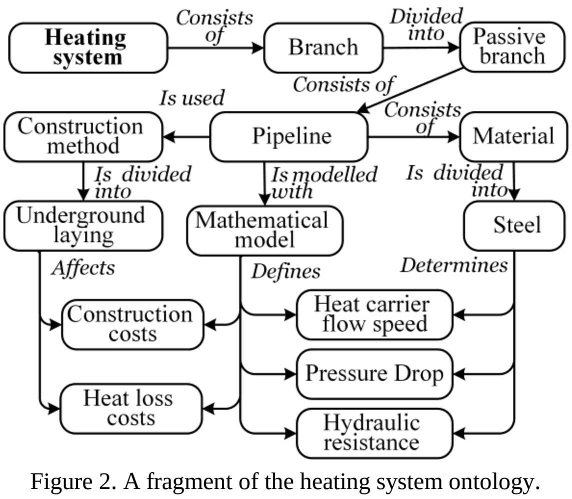

Figure 2. A fragment of the heating system ontology.

its characteristics and methods of construction. The obtained model is stored in a database for a repeated application. Figure 4 presents a model of heating system in two-line representation. It contains one heat source, transmission and distribution networks, and two heat consumers. A graphical schematic editor is used to construct the models of real heating systems consisting of thousands of components. This editor allows construction and use of calculated schemes of heating systems on a site plan.

B. Stage 2. A formal description of an applied problem

In this stage, an engineer formally describes an applied

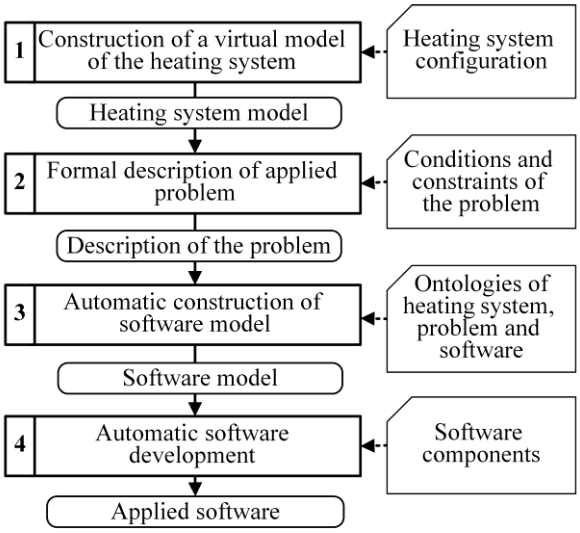

Figure 3. A technique for the model-driven software development.

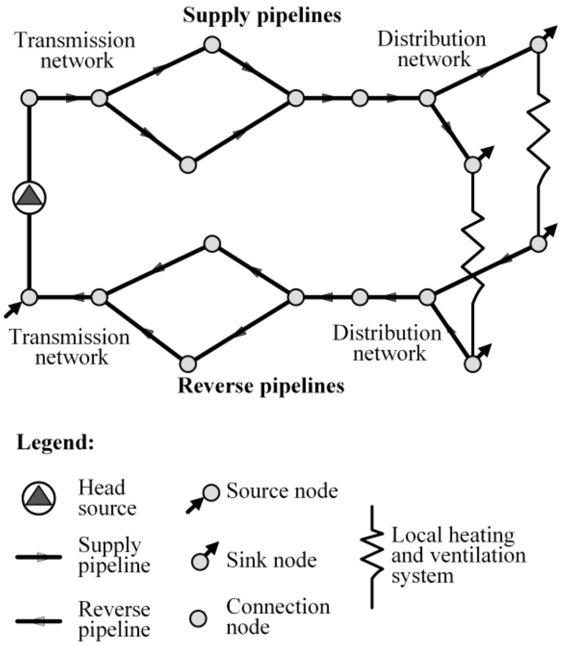

Figure 4. A model of heating system in a two-line.

problem. This implies setting conditions and constraints to determine optimal heating system parameters that include: 1) a set of standard pipelines and their characteristics; 2) constraints on pipeline construction (reconstruction) ways; 3) constraints on pressure at nodes; 4) constraints on speed of heat carrier flow; 5) design values of heat carrier sinks for consumers and heat carrier sources for sources; 6) a set of standard pumping stations and their characteristics.

C. Stage 3. Automatic construction of a software model

This stage suggests the creation of a software model intended for solving the applied problem. This model represents an aggregate of data structures that reflect the following software properties: a structure and composition of software components, description of a computational process in the form of an oriented graph, description of mathematical models of equipment, a list of applied methods and algorithms. The software model is built by an inference engine that for its operation uses a virtual model of heating system, a formal description of the problem and ontologies. The inference rules implemented in XSLT and XPath languages are applied for logical inference from the ontologies written in XML [34, 5].

An algorithm for automated construction of a software model includes the following steps.

Step 1. A list of equipment used to construct a heating system, and ways of its construction, is formed based on the virtual model of the heating system.

Step 2. A list of new equipment allowed to be installed for network reconstruction and ways of its installation is made based on formal description of the problem of determining the optimal heating system parameters.

Step 3. Based on the heating system ontology, the inference engine constructs the data structures that contain description of heating system component models necessary to solve the problem, based on the lists of equipment and ways of its installation that were formed in the previous steps.

Step 4. Based on the virtual model of the heating system, formal description of the applied problem and ontology of problems, the specific methods are chosen to solve the problems, a list of subproblems is made up, and the methods and algorithms are chosen.

Step 5. An oriented graph, defining the computational process is constructed to solve the problem of the determination of optimal heating systems parameters, based on the problem ontology. The nodes of this graph correspond to the steps of problem solving, and edges - to the links between them.

Step 6. Based on the list of methods and algorithms, the inference engine using the software ontology, makes a list of software components necessary to solve the problem of determining optimal parameters of the heating system.

Step 7. The data structures defining the relations "problem – method – component" are formed.

Step 8. Based on the description of components (implementing the methods and algorithms), and the software ontologies that were made in the previous steps, the component interfaces that implement mathematical models of heating system components are determined.

Step 9. Based on the problem ontology and software ontology, the data structures describing input and output parameters for each program component, types of data and ways of their transfer are formed.

D. Stage 4. Automatic construction of software based on its model

In this stage, the software is automatically developed based on its model, with the aid of metaprogramming techniques. The following subproblems are solved in the course of its development.

Subproblem 1. Formation of a set of software components implementing mathematical models of heating system components. Below is an algorithm of constructing a set of software models of heating system components. The algorithm consists of three steps.

Step 1. The development control component calls XSLT-processor to develop models of heating system components, necessary to solve the applied problem. XSLT processor, based on a list of equipment from the software model, mathematical models in MathML format and transformation rules in XSLT format, creates a set of data structures that contain a description of models of heating system components.

Step 2. The development control component calls code builder that, based on the data structures defining the models of heating system components, and the description of interfaces of software components from the software model, builds a software code in the Java language.

Step 3. The development control component calls the Java compiler that compiles a software code built in the previous stage to make a set of necessary models.

Subproblem 2. Loading of software components, that implement the methods and mathematical models, into the memory by the tools of reflective programming which is one of the metaprogamming types [35]. Integration of these components into a single software package is performed through standardized interfaces ensured by design patterns [21].

The system loader that implements the Factory design pattern does the following: 1) it receives a description of software components from the software model; 2) finds the components that correspond to the problem to be solved; 3) prepares data structures necessary for their fetching; 4) loads the components into the memory by using the reflective tools of the Java language; 5) transfers references to the components to the software integration environment.

Subproblem 3. Filling of the hash tables with pairs "number – reference". These hash tables are used by software components represented by methods that receive references to the relevant software components represented by models, according to the numbers of mathematical models of components included in the heating system model.

VI. RESULTS

The software obtained as a result of automated development according to the proposed methods consists of three architectural layers (Fig. 5): 1) a subsystem of computation control (supervisor) that contains the components controlling the computational process; 2) a computing subsystem that solves an applied problem by using the software components that implement methods, algorithms and models; 3) a subsystem of data storage that provides data exchange between database and local memory.

The developed SOSNA software has been successfully applied in Melentiev Energy Systems Institute to determine optimal parameters of heating systems when solving the problems of heating system design, expansion and reconstruction. The proposed methods of model-driven software development enable its flexible application in modeling of the heating systems with a large set of equipment.

The SOSNA software was used to perform multivariate calculations that made it possible to determine the optimal parameters and make recommendations on rational

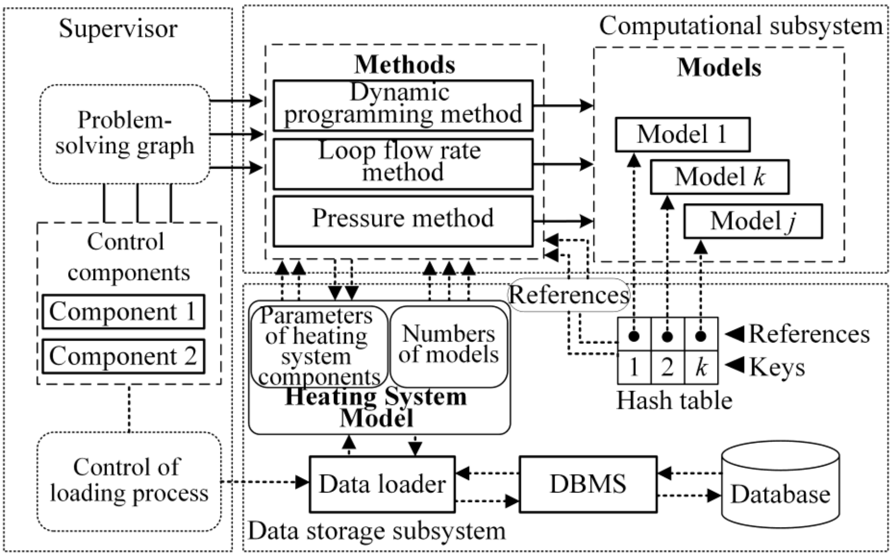

Figure 5. Architecture of software for optimization of heating system parameters.

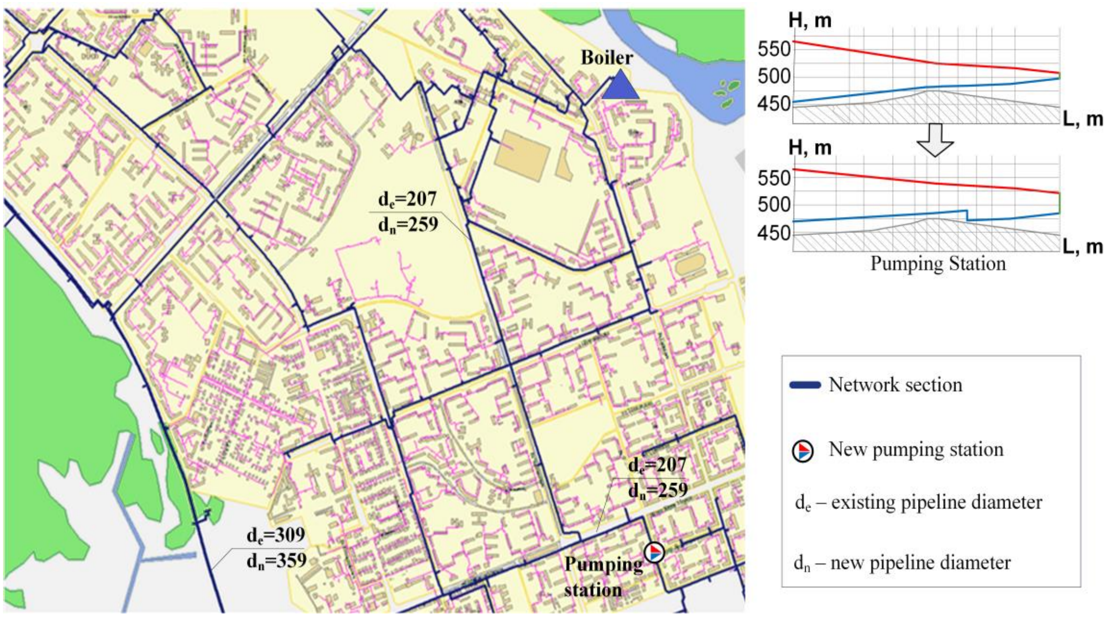

Figure 6. A fragment of Bratsk heating system with recommended reconstruction measures.

reconstruction and expansion of the heating systems in the Central and Admiralteisky districts of Saint Petersburg, the town of Bratsk (Fig. 6) and Magistralny urban-type settlement. The proposed recommendations on the reconstruction of the heating system ensure their adaptation to the growing heat loads and foster their optimal operation.

VII. Discussion And Conclusions

The paper proposes original methods for model-driven development of the SOSNA software aimed at determining optimal parameters of heating systems. These methods are based on the MDE conception that allows the development of sophisticated software packages in the context of an applied problem. The software takes into consideration the specific features of construction and makes recommendations on the equipment for the studied heating system. The software can be used by research, design and operation organizations dealing with heat supply. It makes it possible to generate reconstruction and expansion recommendations which can enhance heating system efficiency and provide the required quality of heat supply to consumers.

Acknowledgements

The work was carried out within the project No. 16-07-00948 of the Russian Foundation for Basic Research and the project III.17.4.3 (No. AAAA-A17-117030310437-4) of the fundamental research program of the Siberian Branch of the Russian Academy of Sciences.

References

[1] A. P. Merenkov and V. Ya Khasilev, *The theory of hydraulic circuits*, Moscow: Nauka, 1985, 280 p. (in Russian).

[2] A. P. Merenkov, E. V. Sennova, S. V. Sumarokov, V. G. Sidler and N. N. Novitskii, *Mathematical modeling and optimization of heat, water, oil and gas supply systems*, Novosibirsk: Nauka, 1992, 407 p. (in Russian).

[3] M. Brambilla, J. Cabot and M. Wimmer, Model *Driven Software Engineering in Practice. Synthesis Lectures on Software Engineering #1*. San Rafael: Morgan & Claypool Publishers, 2012, 182 p.

[4] M. Völter, T. Stahl, J. Bettin, A. Haase and S. Helsen, *Model-Driven Software Development: Technology, Engineering, Management*. New York: Wiley, 2006, 446 p.

[5] M. R. Kogalovsky, *Perspective technologies of information systems*. Moscow: DMK-Press, 2003, 288 p.

[6] V. A. Stennikov, E. A. Barakhtenko and D. V. Sokolov, “Ontologies application for implementation of the software for solving optimal design problems of heat supply systems”, *Informacionnye tehnologii*, no. 3, pp. 2-7, 2013. (in Russian).

[7] Bartlett J. “The art of metaprogramming, Part 1: Introduction to metaprogramming”, October 20, 2005. [Online]. http://www.ibm.com/ developer-works/library/l-metaprog1/

[8] S. V. Sumarokov, “The method of solving the multi-extremal network problem,” *Ekonomika i matematicheskie metody*, vol. 12, no. 5, pp. 1016-1018, 1976. (in Russian).

[9] A. P. Ershov, *Programming Programme for the BESM Computer*, London: Pergamon Press, 1959. 158 p.

[10] N. Wirth, *Algorithms and Data Structures, Upper Saddle River*, New Jersey: Prentice Hall, 1985, 288 p.

[11] B. C. Smith, Procedural Reflection in Programming Languages, Department of Electrical Engineering and Computer Science, PhD dissertation, Massachusetts: Massachusetts Institute of Technology, 1982, 100 p.

[12] G. B. Evgenev, *Intelligent Design Systems*, Moscow: Bauman MSTU, 2009. 334 p. (in Russian).

[13] Y. A. Zagorulko and G. B. Zagorulko, “Ontological approach to the development of the decision support system at the oil and gas producing enterprise,” *Vestnik NGU*, vol. 10, pp. 121-129, 2012. (in Russian).

[14] T. N. Vorozhtsova and S. K. Skripkin, “Using ontologies in modeling of software,” *Vychislitel'nye tekhnologii*, vol. 13, 2008, pp. 376-381. (in Russian).

[15] T. Gruber, *What is an Ontology?* Stanford University, Nov. 9, 2009. [Online], http://www-ksl.stanford.edu/kst/what-is-an-ontology.html

[16] N. Guarino, Understanding, Building, and Using Ontologies. Nov. 1, 2010. [Online]. http://ksi.cpsc.ucalgary.ca/KAW/KAW96/guarino/guarino.html

[17] EPANET. Model for Water Distribution Piping Systems. February 21, 2017, [Online]. http://www.epa.gov/nrmrl/wswrd/dw/epanet.html

[18] The Thermoflow Software. Fabruary 21, 2017, [Online]. https://www.thermoflow.com/products_appspecific.html

[19] L. S. Popyrin, V. I. Samusev and V. V. Epel'shtein, *Automation of mathematical modeling of thermal power plants*, Moscow: Nauka, 1981, 236 p. (in Russian).

[20] A. V. Alexeyev, N. N. Novitsky, V. V.Tokarev and Z. I. Shalaginova, “Principles of development and software implementation of the information and computing environment for computer modeling of pipeline and hydraulic systems”, in *Pipeline systems*

of power engineering. Methods of mathematical modeling and optimization, Novosibirsk: Nauka, 2007, pp 221-230. (in Russian).

[21] E. Gamma, R. Helm, R. Johnson and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*, Boston: Addison-Wesley, 1994, 395 p.

[22] R.C. Martin, *Agile Software Development: Principles, Patterns, and Practices*, Upper Saddle River, New Jersey: Pearson Education, 2002, 529 p.

[23] M. Fowler, Patterns of Enterprise Application Architecture, Boston: Addison-Wesley, 2002, 533 p.

[24] M. M. Gorbunov-Posadov, *Expandable programs*, Moscow: Poliptikh, 1999, 336 p. (in Russian).

[25] L. V. Massel, N. N. Makagonova, V. V. Triputina, A. Yu. Gornov, E. A. Boldyrev, G. G. Massel, A. P. Dem'yanchik and D. V. Podkamenny, *Integration of Information Technologies in Systemic Energy Studies*, Novosibirsk: Nauka, 2003, 320 p. (in Russian).

[26] D. Knuth, *The Art of Computer Programming. 1: Fundamental Algorithms*, 3rd ed., Boston: Addison-Wesley Professional, 1997, 650 p.

[27] E. W. Dijkstra, A *Discipline of Programming, Upper Saddle River*, New Jersey: Prentice Hall. 1976. 217 p.

[28] A. V. Aho, J. E. Hopcroft and J. D. Ullman, *Data Structures and Algorithms*, Massachusetts: Addison-Wesley, 1983, 427 p.

[29] R. Sedgewick and K. Wayne. *Algorithms*, 4th ed., Massachusetts: Addison-Wesley Professional, 2011, 992 p.

[30] S. Skiena, *The Algorithm Design Manual*, 2nd ed., Berlin: Springer, 2010, 730 p.

[31] T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein, *Introduction to Algorithms*, 3rd ed., Massachusetts: The MIT Press. 2009. 1312 p.

[32] V. A. Stennikov, E. A. Barakhtenko and D.V. Sokolov, "A methodological approach on the basis of Model-Driven Engineering and ontologies for the development of a software for heat supply system design," *Informatsionnye tekhnologii*, 2015, vol. 21, no. 3, pp. 201-209. (in Russian).

[33] J. Gosling, B. Joy, G. Steele, G. Bracha and A. Buckley, *Java Language Specification, Java SE 8 Edition*, Boston: Addison-Wesley Professional, 2014, 792 p.

[34] M. Kay, XSLT 2.0 and XPath 2.0 *Programmer's Reference*, 4th Edition. New York: Wrox, 2008, 1376 p.

[35] I. Forman and N. Forman, *Java Reflection in Action*, Greenwich: Manning Publications Co., 2005, 273 p.

Valery A. Stennikov is Professor, Director of the Melentiev Energy Systems Institute of Siberian Branch of the Russian Academy of Sciences, Corresponding Member of the Russian Academy of Sciences. He received his PhD degree in 1985 and the degree of Doctor of Technical Sciences in 2002. His research interests include: the methodology, mathematical models and methods for development of heat supply systems taking into account reliability and controllability requirements; energy effective technologies and equipment; energy saving; methods and algorithms of calculation of tariffs for heat energy; intelligent integrated energy system.

Evgeny A. Barakhtenko received Dr. of Eng. degree from the Irkutsk State Technical University in 2011. He is Senior Researcher at the Melentiev Energy Systems Institute of Siberian Branch of the Russian Academy of Sciences. His research interests include: methodical bases and software for solving the problems of optimal development of pipeline systems and intelligent integrated energy system.

Dmitry V. Sokolov received Dr. of Eng. degree from the Melentiev Energy Systems Institute of Siberian Branch of the Russian Academy of Sciences in 2013. He is Senior Researcher at the Melentiev Energy Systems Institute of Siberian Branch of the Russian Academy of Sciences. His research interests include: optimization methods, development of high-speed algorithms and software.