8th Biennial ASME Conference on Engineering Systems Design and Analysis
July 4-7, 2006, Torino, Italy

ESDA2006-95387

MODELLING AND SIMULATION OF FLUID POWER SYSTEMS IN OBJECT-ORIENTED PROGRAMMING ENVIRONMENT

Gunnar Grossschmidt, Mait Harf, Taavi Sallaste

Tallinn University of Technology, Institute of Machinery and Institute of Cybernetics
Ehitajate tee 5, 19086 Tallinn, Estonia
E-mail: gunnar.grossschmidt@ttu.ee

ABSTRACT

The paper deals with principles of computer modelling and simulation of behaviour of fluid power systems in object-oriented programming environment. The approach is based on using multi-pole models and signal-flow graphs of functional elements, that enables methodical, graphical representation of mathematical models of large and complicated chain systems. In this way we can be convinced in the correct composing of models. A high-level programming environment NUT is used as a tool for building modelling and simulation systems. Several modelling and simulation systems have been developed using approach described above. Different simulation tasks have been solved on these modelling systems. Methodology described in the paper has several advantages and novelties.

Keywords: modelling and simulation, load-sensing fluid power systems, multi-pole models, object-oriented programming tools.

INTRODUCTION

Modelling and simulation of hydraulic systems has been investigated in Tallinn University of Technology for several decades. General principles of the research have been published in [1].

In particular case we observe hydraulic and electro-hydraulic systems (e.g. automatically regulated fluid power systems of stationary and mobile machines, steering mechanisms of cars, ships and aeroplanes, drives of robots and manipulators, vibration equipment for manufacturing and testing, amortisation and vibro-isolation systems) as objects of application. Principles and methods to use are applicable for simulating technical chain systems with any kind of physical content.

Most of existing simulation tools for fluid power systems are oriented to mathematical models of single causality and to composing and solving large differential equation systems. Equation systems are difficult to modify during calculations. It is not possible to apply different calculation procedures to different components of the system. It is difficult to debug and solve large equation systems.

1. MULTI-POLE MODELS, CAUSALITIES

The mostly used multi-pole models are the two-pole and the four-pole models [2].

The two-pole model expresses the relation between input and output, for example between flow variable and potential variable. We have several forms of mathematical causality by two-pole models. Two-pole models of functional elements (**FE**) are expressed by one equation.

The four-pole models show the relations between two pairs of potential and flow variables. One of the variables in pair must be the input. Only in that way we can take into account the input and the output in the same port. Models of that kind express the physical content of processes with feedback.

Four forms of such four-pole models, or otherwise – four forms of mathematical causality exist. Letters **G, H, Y** and **Z** as in electrical engineering denote them. Elementary **FE** four-pole models are expressed by two equations.

The four-pole models of form **G** and **H** (causality) represent the orientation in both directions (transfer functions $\mathbf{G}_{12}$, $\mathbf{G}_{21}$, $\mathbf{H}_{12}$ and $\mathbf{H}_{21}$) with corrections through the cross dependencies (transfer functions $\mathbf{G}_{11}$, $\mathbf{G}_{22}$, $\mathbf{H}_{11}$ and $\mathbf{H}_{22}$ in **Fig. 1**).

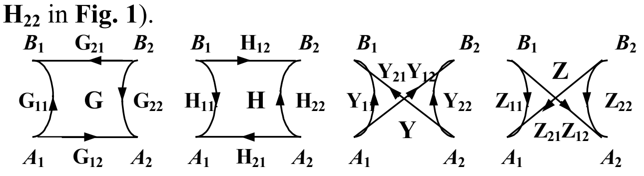

Fig. 1 – Oriented graphs of four-pole model of forms **G, H, Y** and **Z** (causalities).

The four-pole models of forms **G** and **H** of elementary **FE**s for the transient response calculation include the members with differentiation procedure in cross dependencies. Mostly the four-pole models of forms **G** and

H are used for calculation of the statics, steady-state conditions and frequency characteristics.

The four-pole models of forms **Y** and **Z** of the elementary **FE**s change the type of variable, which proceed through other elements of the system. The four-pole model of form **Y** has potential variables as inputs and flow variables as outputs. The four-pole model of form **Z** has conversely flow variables as inputs and potential variables as outputs. The four-pole models of forms **Y** and **Z** of the elementary **FE-**s for the transient response calculation include the transfer functions with integration procedure.

The four-pole model of form **Y** for mechanical and hydraulic inertia expresses only the dynamics. Hydraulic volume elasticity four-pole model of form **Z** is also used only for dynamics. The four-pole model of form **Z** does not exist for mechanical inertia and damping, and for hydraulic inertia and resistance. The four-pole model of form **Y** does not exist for mechanical elasticity and for hydraulic volume elasticity.

As an example of the six-pole model, the model of hydraulic cylinder is considered.

The hydraulic cylinder (**Fig. 2**) has three pairs of variables: p_1, Q_1; p_2, Q_2; x (or v), F; where p_1, p_2 – pressures in the cylinder chambers, Q_1, Q_2 – volume flow rates in cylinder chambers, x, v – position and velocity of the piston rod, F – force on the piston rod.

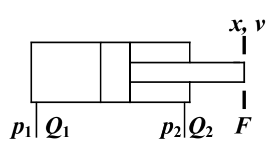

Fig. 2 - Variables of the hydraulic cylinder.

There exist four forms (causalities) of six-pole models for a hydraulic cylinder. Six-pole models will describe the tee couplings. There exist six forms of six-pole models of the tee couplings. Here the dependencies are very simple and, therefore, all these forms can be described through the same non-causal equations. Here the symbolic solution method is used. The six-pole models of elementary **FE-**s are expressed through three equations.

As an example of the seven-pole model, the model of variable displacement axial piston pump is considered in **Fig. 3**.

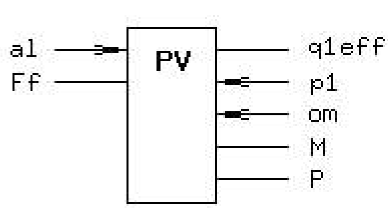

Fig. 3 – Multi-pole model of variable displacement axial piston pump.

The inputs are denoted with arrows and the outputs are denoted without arrows.

The multi-pole model of variable displacement axial piston pump has the following input and output variables:

al	position angle of the pump swash plate,
Ff	reaction force of the swash plate,
q1eff	volumetric flow at the pump outlet,
p1	pressure at the pump outlet,
om	angle velocity of the electric motor,
M	reaction moment of the pump,
P	output power of the pump.

2. NUT PROGRAMMING ENVIRONMENT

The NUT system is a programming tool, which supports declarative programming in a high-level language, automatic program synthesis and visual programming.

The NUT programming language rests on two paradigms: procedural object-oriented programming and the automatic synthesis of programs from declarative specifications. The NUT language is object-oriented. Concepts in it are specified as classes, and then used either in computations or for specifying new concepts. There is one big difference between the classes in ordinary object-oriented language and in NUT. The latter contain more knowledge than ordinary classes, and can be directly used as specifications for problem solving. The description of NUT classes may contain specifications of their components, methods, initial values and other properties. Due to an equation solver built into the language processor, the system is able to interpret arithmetic equations as multi-way procedures for computing the unknown components of the equation. Each class can have a visual representation as well, so that much computing can be described visually.

Automatic synthesis of programs is a technique for the automatic construction of programs from the knowledge available in specifications of classes. Having a specification of a class, we are, in general, interested in solving the following problem: find an algorithm for computing the values of components y1,..., yn from the values of components x1, ..., xm. The automatic synthesis of programs, as practised in NUT, is based on proof search in intuitionistic propositional logic.

The NUT graphics facilities include Graphics Editor, a set of graphics functions in the language, and the Scheme Editor. The Scheme Editor is a tool for visual programming that allows the user to define and use classes by means of graphical schemes. In order to draw schemes of problem specifications, we must have, for each class, an icon in the palette and an image, which will represent an object in a scheme. So there must be an icon and an image for every class. This can be done immediately after specifying a class. After specifying all the classes together with their icons and images one can specify and use for computations a number of different schemes using defined classes. There are numerous built-in features of the scheme editor, which support visual programming:

- connection lines between ports which represent equalities binding the ports;
- an interactive zoom-in window can be used for showing or editing of any object of a scheme (this window is

formatted automatically on the basis of the class specification);

- requests for computing elements of a schema can be given from menus.

3. MODELLING AND SIMULATION OF FLUID POWER SYSTEMS

A number of packages for different fluid power systems modelling and simulation have been implemented in the NUT system [3,4]. Multi-pole models of functional elements have been described as NUT classes together with their icons and images. Besides multi-pole model classes, several supporting classes as "clock" for the time, "source" for the disturbance, "process" for organising the whole computing process, have been specified. Using visual specifications of described multi-pole models one can graphically compose models of various fluid power systems. When solving specific simulation problem, model has to be adjusted by evaluating different parameters of the elements and adding sources to elements of the model that describe disturbances of the necessary shape and values.

During the simulation, some elements of the model need parameters, which values cannot be computed at the moment they are required. For computing values of such parameters a special method has been used. When starting the process, approximate values of such critical parameters have been given as initials. At each step of the simulation process we try to refine initial approximate parameters using a special iteration procedure. We use the NUT system to synthesize programs for re-computing some parameters and try to re-compute them iteratively until precise values of the parameters have been attained.

A special element “disp” is used in the scheme for graphical displaying of dependencies we are interested in.

The simulation is organized as computing of static, steady state motion and dynamics of the hydraulic device. In general, model for static and steady-state condition differs from the model for dynamic responses. Nevertheless fragments for static calculations can expand the model for dynamic responses.

4. COMPUTING PROCESS ORGANIZATION

Using visual specifications of described multi-pole models one can graphically compose a number of various fluid power systems. It is possible to solve a great number of various computing tasks on each fluid power system evaluating some components as inputs and computing some other components as outputs.

The whole computing process is organised by supporting class “process”. State variables are introduced for every functional element to characterise the features of the element at the current simulation time step. Simulation process starts from initial state and includes calculation of following state (*nextstate*) from previous states (from *oldstate* and *state*). As a result of simulation final state (*finalstate*) is computed.

5. MODELLING OF A HYDRAULIC LOAD-SENSING SYSTEM

5.1. Mechanical-hydraulic load-sensing system

As an example we consider modelling and simulation of a hydraulic load-sensing system. Fluid power systems, in which working pressure (pressure in pump output) is kept proportional to load, are called hydraulic load-sensing systems. Such systems are mainly used in mechanisms containing numerous drives to run with purpose to save energy. These are quite complicated automatically regulating systems and until now optimal solutions for such systems have not been found.

The scheme of a hydraulic load-sensing system of Bosch GmbH, discussed in more details in [5], is shown in **Fig. 4**.

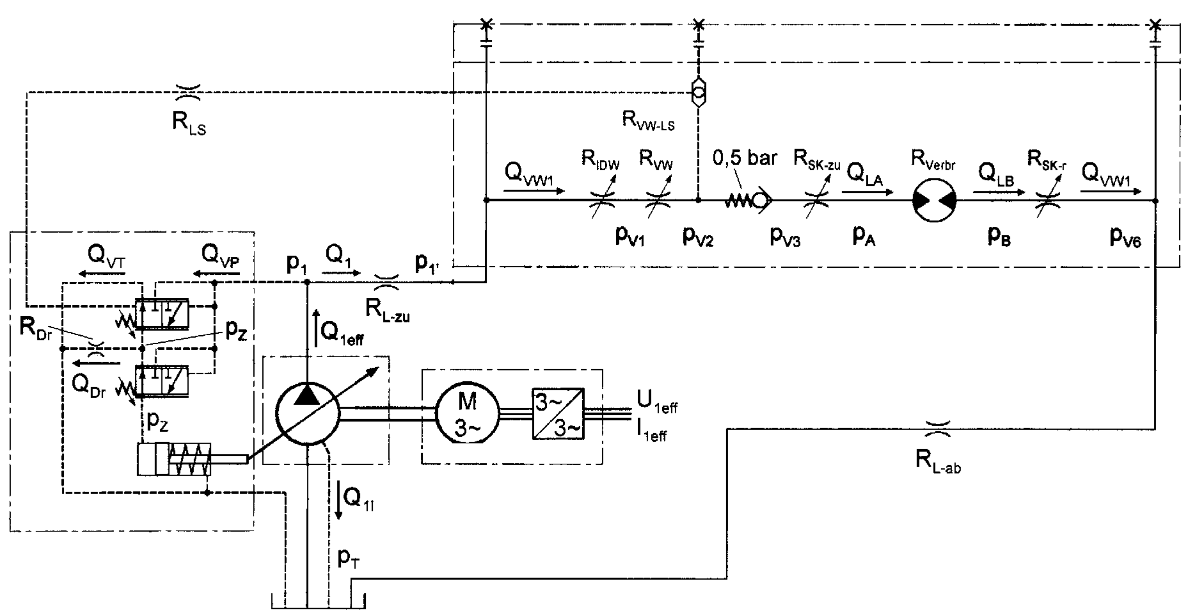

Fig. 4 - The scheme of a hydraulic load-sensing system.

The variable displacement axial piston pump is driven by an electric motor ***M***. Hydraulic-mechanical control of the pump volumetric flow is performed by regulating valves and hydraulic cylinder. The feeding chain of the hydraulic motor ***Rverbr*** contains tube ***Rl-zu***, pressure compensator ***Ridw***, measuring valve ***Rwv***, check valve, meter-in throttle edge ***Rsk-zu***, connection elements, etc.
The output chain of the hydraulic motor contains a meter-out throttle edge ***Rsk-r***, and tube ***Rl-ab***. The device contains load-sensing pressure feedback with resistance ***Rls***.

The scheme of a hydraulic-mechanical controller (**Fig.5**) contains spool valve (effective area *Av*) with inflow and outflow slot, constant resistor (volumetric flow *Qdr*), positioning cylinder (effective area *Az*), and swash plate with spring.

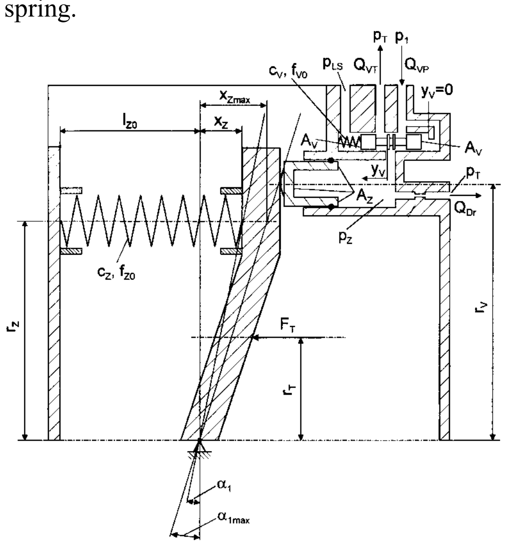

Fig. 5 - The scheme of a hydraulic-mechanical controller.

The valve block is shown in **Fig. 6**. Main part of the valve block is directional valve with measuring throttle edge *Rvw,* meter-in throttle edge *Rvs-zu* and meter-out throttle edge *Rsk-r*. The valve block also contains pressure compensator with throttle edge *Ridw* and check valve with pressure drop *0,5 bar.*

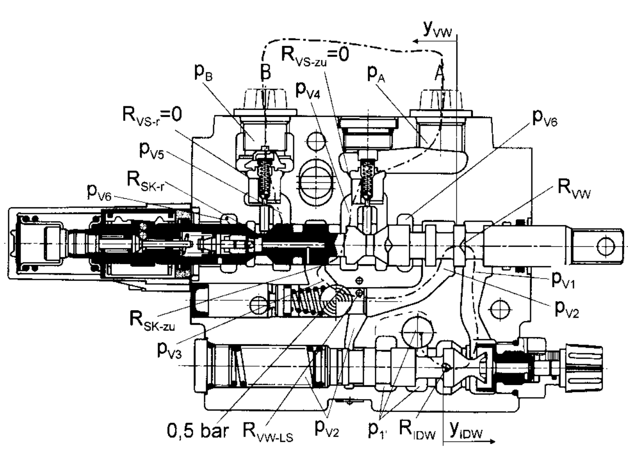

Fig. 6 - The valve block.

5.2. Configuring the multi-pole model

To build up the multi-pole model it is necessary to break the scheme of the load-sensing system into logical components and subsystems and define connecting poles.

Models of the following components of the load-sensing system have been developed: variable displacement axial piston pump, electric motor, hydraulic-mechanical controller, relief valve, valve block, hydraulic motor, tubes and multiple tube connection elements.

Model of the hydraulic-mechanical controller includes models of spool valve, meter-in throttle edge, meter-out throttle edge, resistor and positioning cylinder with swash plate.

Model of the valve block includes measuring valve with pressure compensator, check valve with meter-in throttle edge, and meter-out throttle edge.

The multi-pole model of the whole load-sensing system has been built up from the components models. First, necessary components have been connected trough connecting poles. Second, variables of connection poles have been defined as inputs or outputs for every component depending of required causalities [5].

All the models of the load-sensing system components have been described as NUT classes together with their images and icons.

The model of the whole load-sensing system (**Fig. 7)** has been composed based on the scheme of a hydraulic load-sensing system (**Fig. 4**). The model contains 18 component models: variable displacement axial piston pump *PV* (8), electric motor *ME* (11), pump controlling device that consists of pump control spool valve *VP* (1), meter-in throttle edge of the spool valve *RVP* (2*)*, meter-out throttle edge of the spool valve *RVT* (6) interface element *IEH* (3), constant resistor *REL* (5) and positioning cylinder *ZV* (4), relief valve *VD* (7), measuring valve with pressure compensator *RIDVW* (14), meter-in throttle edge for the hydraulic motor *RSKZ* (21), hydraulic motor *MH* (22), meter-out throttle edge for the hydraulic motor *RSKA* (23), tubes *Rtu* (12, 13, 25), interface elements *IEH* (10, 16).

In order to perform simulations on the model of the load-sensing hydraulic system we need to adjust the model, test all the components and set up system parameters.

5.3. Adjusting the multi-pole model

Adjusting the model includes several stages.

First, we need to choose main components of the system: hydraulic motor, hydraulic pump and electric motor.

Second, we need to set up system parameters. Initially, approximate values of system parameters have been chosen:

- Required range of pressures 6…12 bars for pump volumetric flow from *max* to *min* has been found, depending on the parameters of the pump.
- The difference of pump outlet pressure and feedback (load-sensing) pressure for pump regulation from *max* to *min* has been taken 15…20 bars.

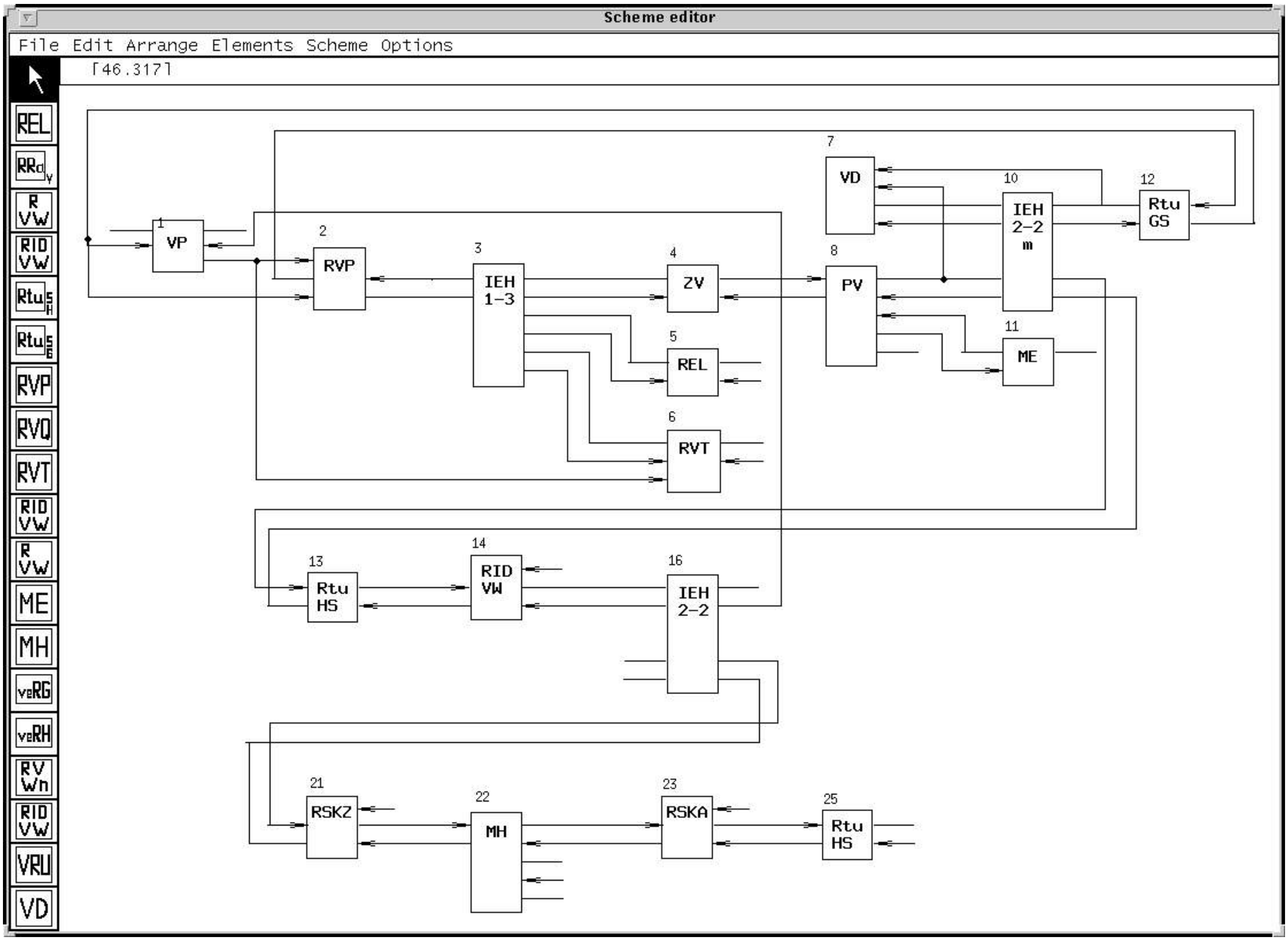

Fig. 7 - Model of the whole load-sensing hydraulic system.

- Shapes and sizes of pump control spool valve throttle edges have been designed to get the range of pump controlling pressure 6…12 bars.
- The minimal volumetric flow of the pump has to be adjusted to guarantee feeding of the pump regulating system.
- Pressure drop in measuring valve has been taken 5…7 bars, which was guaranteed by parameters of the pressure compensator.
- Throttle edges of the measuring valve have been designed to attain the correspondence of volumetric flows to pressure drops.
- Throttle edges of the pressure compensator have been designed to attain the summary pressure drops in measuring valve and pressure compensator that corresponds to volumetric flows.

Third, all the models of components must be tested separately. For this purpose, for every component the simulation problem has been composed, approximate input signals have been chosen and finally, action of the component has been simulated in order to be sure component's model works correctly.

Fourth, the separately tested components models have been connected into more complicated subsystems and tested in behaviour. At this and following stages, problems of growing difficulty arise. Typical is the appearance of feedback chains, which make dependencies between parameters essentially complicated and more difficult to observe.

5.4. Modelling subsystems

5.4.1. Pump with electric motor. Simulation problem description for testing variable displacement axial piston pump *PV* with electric motor *ME* is shown in **Fig. 8**.

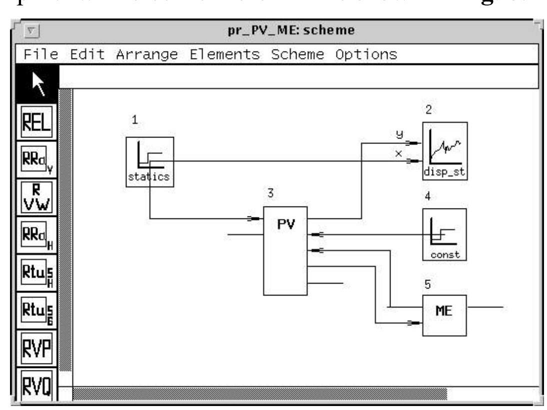

Fig. 8 - Variable displacement axial piston pump with electric motor

Input value of pump pressure (4) is given as *const* and a range of values for position angle of the pump swash plate is given as *static* (1). The volumetric flow depending on the position angle of the pump swash plate has been calculated and visualized by *disp* (2). The result is shown in **Fig. 9**.

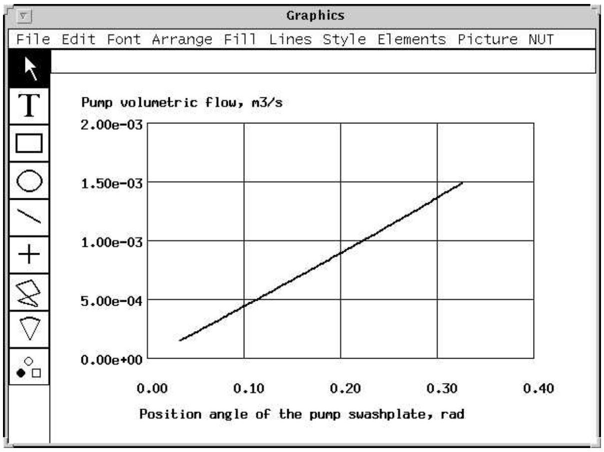

Fig. 9 - The pump volumetric flow depending on the position angle of the pump swash plate.

5.4.2. Hydraulic-mechanical controller with pump and electric motor. Simulation problem description for testing hydraulic-mechanical controller (see **Fig. 5**) together with pump and electric motor is shown in **Fig. 10**. The scheme contains variable displacement axial piston pump *PV* (9), electric motor *ME* (12) and pump controlling device that consists of pump control spool valve *VP* (2), inflow spool valve slot *RVP* (4), outflow spool valve slot *RVT* (8), interface element *IEH* (5), constant resistor *REL* (7) and positioning cylinder *ZV* (6).

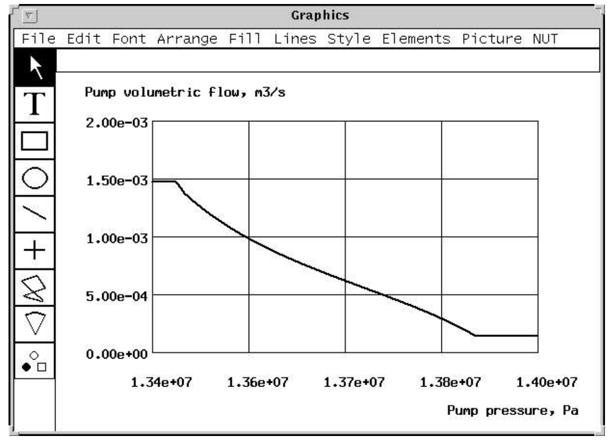

Fig. 11 - The pump volumetric flow depending on the pump pressure.

Input values load-sensing pressure (3) and controller output pressure (10) are given as *const* and a range of values for pump pressure is given as *statics* (1).

In **Fig. 11** the graph of pump volumetric flow depending on pump pressure is shown. The volumetric flow value is maximal if the difference of pump pressure and load-sensing pressure is less than approximately 14.5 bars. The volumetric flow value is minimal if the difference of pump pressure and load-sensing pressure is more than approximately 18.6 bars.

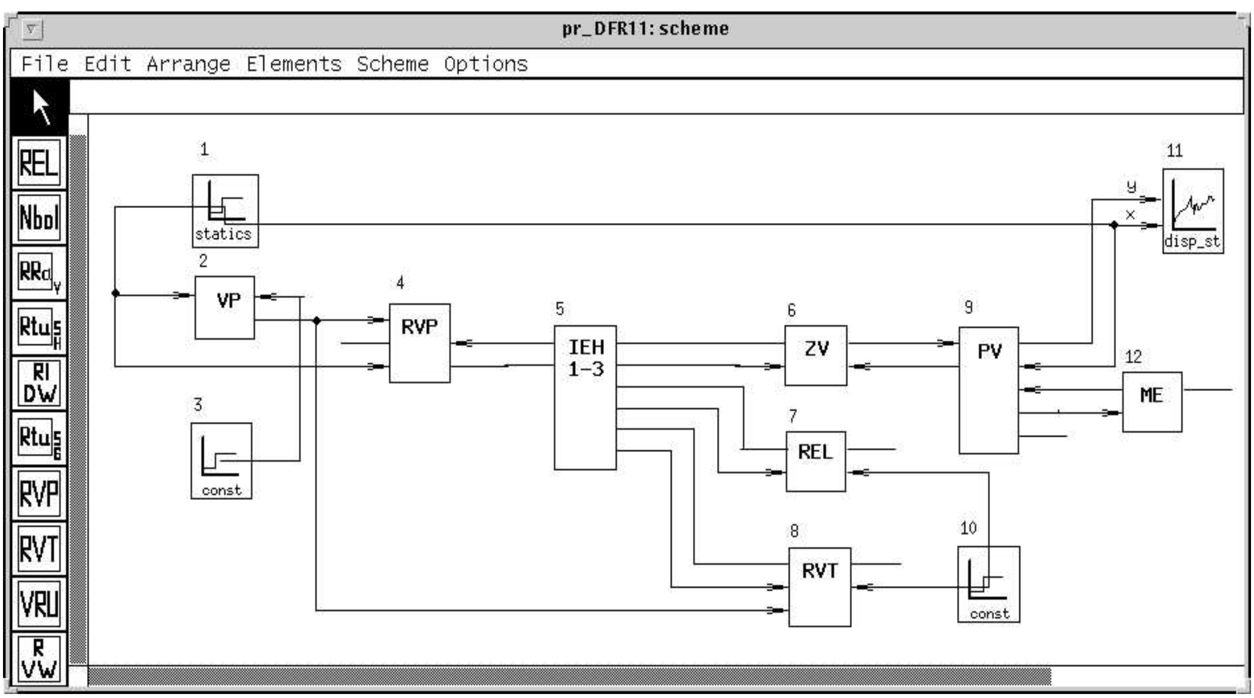

Fig. 10 – The scheme of a hydraulic-mechanical controller with pump and electric motor.

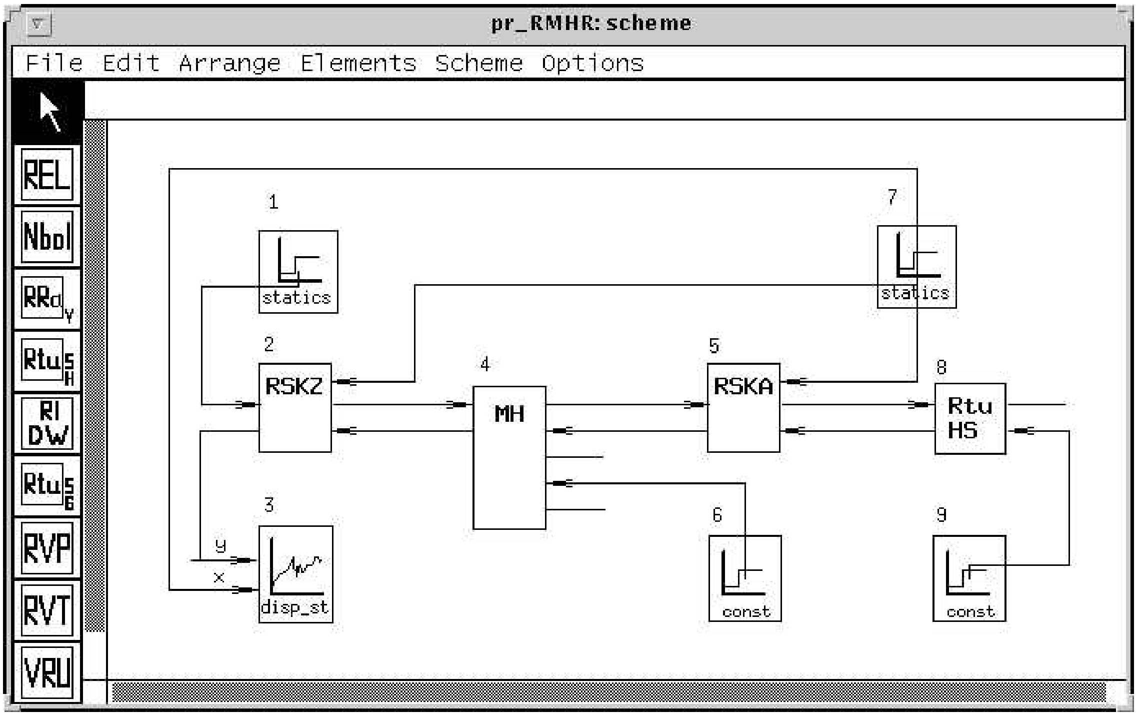

Fig. 12 - Hydraulic motor with throttle edges.

5.4.3. Hydraulic motor with throttle edges. Simulation problem description for testing hydraulic motor *MH* (4*)* together with meter-in throttle edge *RSKZ* (2), meter-out throttle edge *RSKA* (5*)* and outlet tube *RtuHS* (8) is shown in **Fig. 12.**

Input values for load torque of the motor (6) and outlet pressure (9) are given as *const* and ranges of values for inlet volumetric flow (1) and direction valve displacement (7) are given as *statics*.

In **Fig. 13** the graph of inlet pressure of the meter-in throttle edge depending on displacement of the direction valve is shown.

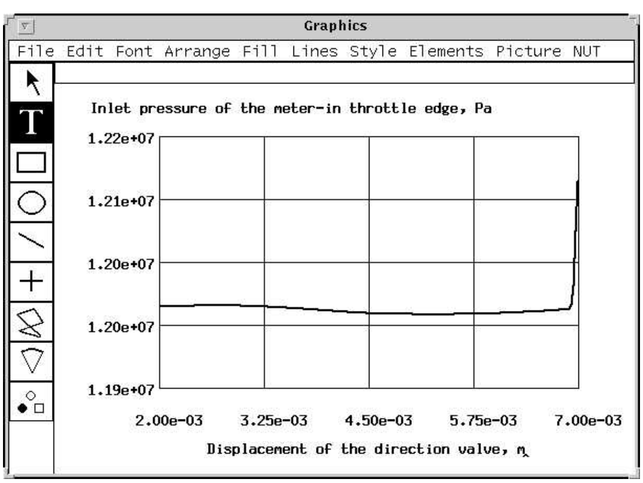

Fig. 13 – The inlet pressure of the meter-in throttle edge depending on the displacement of the direction valve.

The shape and size of meter-out throttle edge has been chosen to make the pressure before the meter-in throttle edge not to depend on the volumetric flow. Here the simulation proceeds from directional valve displacement value 7.00e-03 to 2.00e-03. As approximate values of pressures, given as initials, are not precise enough, the results of the first 4 steps of simulation are not reliable.

5.5. Modelling the whole load-sensing system

The simulation problem description of the whole load-sensing hydraulic system steady-state calculations is shown in **Fig. 14.**

The model contains 19 component models and several constants and static parameter components: variable displacement axial piston pump *PV* (8), electric motor *ME* (11), pump controlling device which consists of pump control spool valve *VP* (1), inflow spool valve slot *RVP* (2*)*, outflow spool valve slot *RVT* (6) interface element *IEH* (3), constant resistor *REL* (5) and positioning cylinder *ZV* (4), relief valve *VD* (7), measuring valve with pressure compensator *RIDVW* (14), meter-in throttle edge *RSKZ* (21), hydraulic motor *MH* (22), meter-out throttle edge *RSKA* (23), tubes (12, 13, 25), interface elements *IEH* (10, 16), efficiency coefficients calculator *WG* (18), inputs *const* (9, 15, 24), range input for displacement of the direction valve *static* (19) and range input for hydraulic motor load moment *static* (24).

Pressure drop from pump to hydraulic motor (load-sensing pressure drop) between given maximum and minimum volumetric flow must change linear. It must be achieved by measuring valve with pressure compensator *RIDVW.* Linear dependence of pressure drop in measuring valve and pressure compensator *RIDVW* on the displacement of the direction valve is shown in **Fig. 15**.

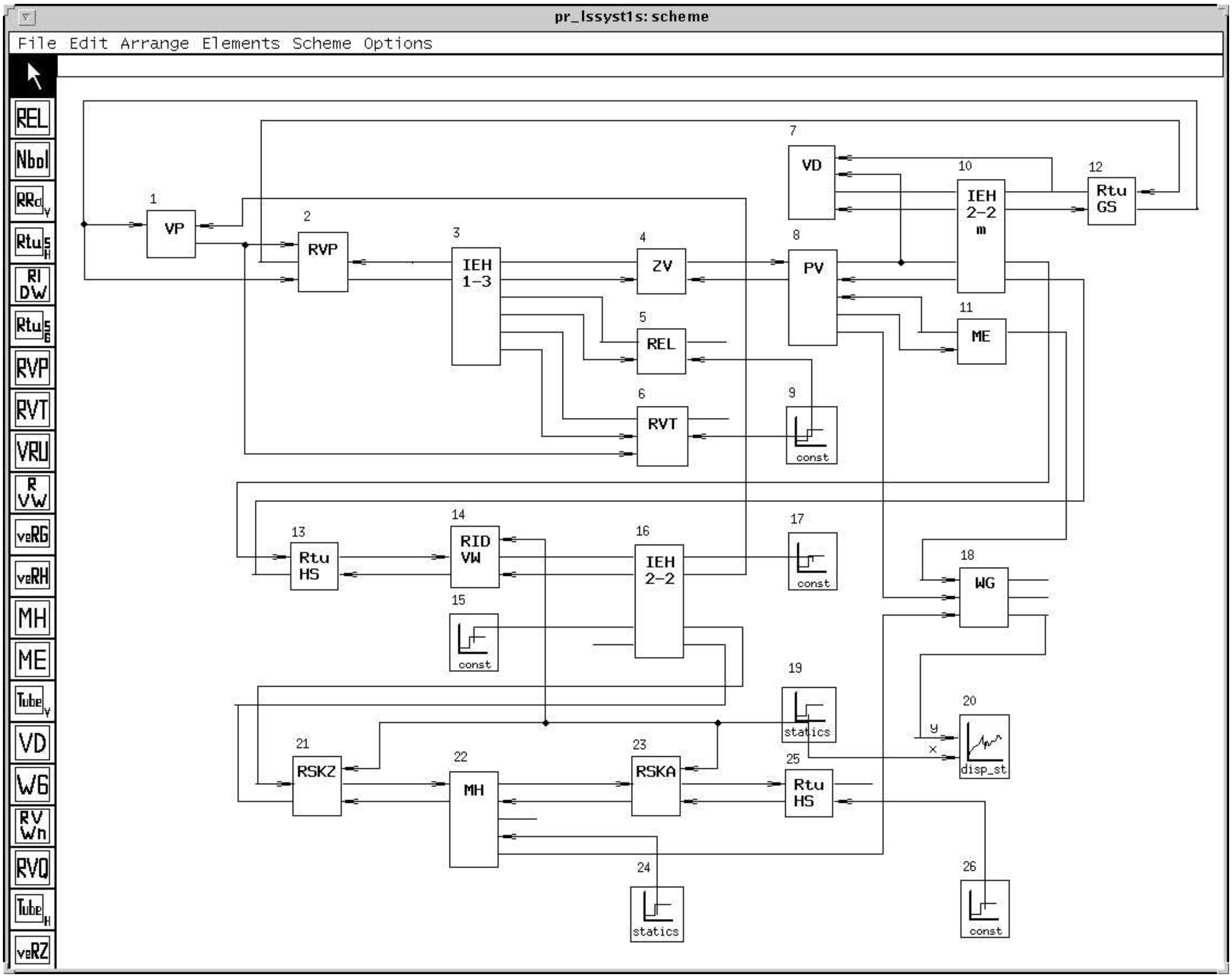

Fig. 14 - Simulation task description of the whole load-sensing hydraulic system steady state calculations.

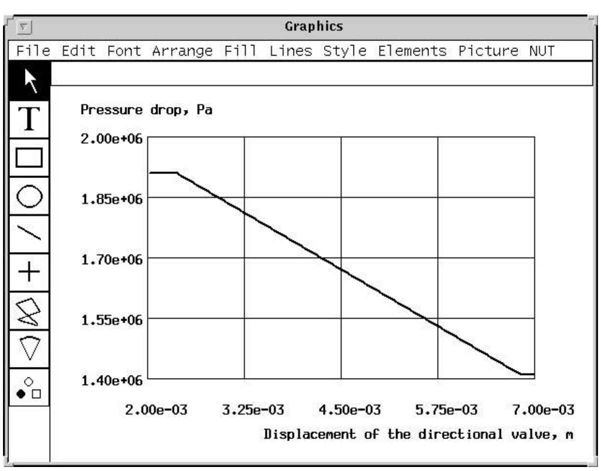

Fig. 15 – Pressure drop in measuring valve and pressure compensator *RIDVW* depending on the displacement of the direction valve.

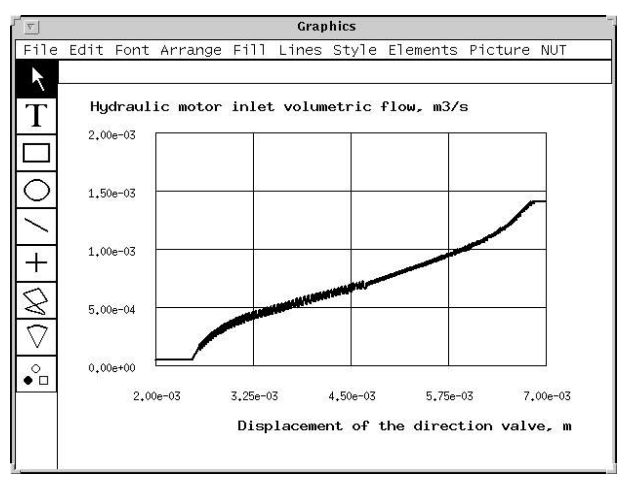

Fig. 16 - Hydraulic motor inlet volumetric flow depending on the displacement of the direction valve.

In **Fig. 16** the dependence of the hydraulic motor inlet volumetric flow on the displacement of the direction valve in the case of constant load moment is shown. In ideal case the dependence must be of linear form. In our case in both, initial and final area, the dependence shape a little differs from linear.

In **Fig. 17** the dependence of the efficiency coefficient of the whole system on the displacement of the direction valve in the case of constant value of hydraulic motor load moment is shown. The efficiency coefficient of the whole system increases together with increasing of the hydraulic motor speed.

Roughness of the curves is caused by restrictions applied to system parameters deviations that were necessary to achieve convergence of the iterations.

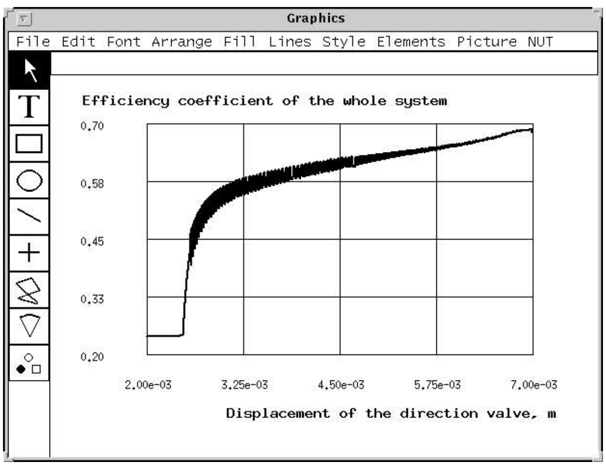

Fig. 17 - Efficiency coefficient of the load-sensing system depending on displacement of the direction valve.

As a result of solving a number of simulation tasks a set of working parameters of the load-sensing system has been adjusted as follows.

- Required range of pump controlling pressure 5.9…11.3 bars for hydraulic motor volumetric flow from q_{max} = 0.001416 m^3/s to q_{min} = 0.000057 m^3/s has been found.
- Pressure drop in measuring valve has been found to be from 5.22e+05 bars to 6.76e+05 bars.
- Pressure drop variation in measuring valve and pressure compensator for pump control from q_{max} to q_{min} has been found to be 14.1…19.1 bars.

5.6. Size and complexity of the modelling system.

Size and complexity of the modelling and simulation system of hydraulic load-sensing systems can be characterised by following numbers.

The modelling system contains 112 classes, including:

- 51 load-sensing system component classes;
- 18 supporting classes;
- 43 element, subsystem and system simulating task classes.

The simulation task of the whole load-sensing system considered above contains:

- 26 classes, including 16 load-sensing system component classes;
- 1026 variables;
- 10 variables that have to be iterated during the computations;
- 59 bindings between system components.

The automatically constructed problem-solving algorithm contains:

- 143 operators (equations, methods, assignments, etc.)
- 32 separate algorithms for solving different subtasks.

All the simulations were performed on the Sun workstation in the UNIX environment.

6. CONCLUSIONS

Hydraulic load-sensing systems are complicated regulating systems with great number of components and several feedbacks. Feedbacks make system very sensible and difficult to simulate. Very precise parameter setting, especially for resistances of hydraulic valve spools, is required to make the system function.

Object-oriented approach and especially automatic program synthesis, used in the research, are original methods in simulation. Using object-oriented approach enables in flexible way to compose and experiment with various large and complicated models. Automatic program synthesis allows describe and solve great number of simulation tasks in order to find out better solutions to design problems.

As a result of the current research a simulation system is proposed that enables to provide computer experiments at the first stage of design of complicated fluid power systems.

Using the simulation system proposed above helps the designer to build up and provide computational experiments with fluid power systems of different configuration and parameters. Results of simulations can be used as initial data while building trial versions of real load-sensing fluid power systems. Using computer simulation system proposed above enables to reduce time and costs of final experimental design.

The main features of the approach proposed in the paper are as follows.

- Mathematical models of the functional elements are composed as multi-pole models taking into account signal propagation in both directions.
- Used multi-pole models can have various causalities.
- Mathematical model of the fluid-power system contains models of functional elements and carries the full information about connections of input/output variables, which express the considered mathematical causalities and guarantees the completeness of the model.
- Modelling and simulation is built up in object-oriented way using the NUT programming environment. This enables to automate and visualise the simulation process.

- Simulation is performed step by step, starting from simulation of components and moving to more complicated subsystems.
- Calculations are performed separately for each multi-pole model. Iteration methods are used in cases of loop dependencies that may appear between component models when they are connected together into more complicated ones.

ACKNOWLEDGEMENT

This research was partially supported by Estonian Scientific Foundation (Grant No. 5867).

REFERENCES

[1] Grossschmidt G., 1997, "Principles of Modelling and Simulation of Technical Chain Systems", Proc. of the 9th European Simulation Symposium & Exhibition, Simulation in Industry, October 19-23, 1997, Passau, Germany, pp. 705-712.

[2] Grossschmidt G., and Vanaveski J., 1998, “Causality of Mathematical Models of Technical Systems”, Proc. of the 12th European Simulation Multiconference ESM´98, as part of the 50th anniversary celebrations of the University of Manchester, Simulation – Past, Present and Future, June 16-19, Manchester, United Kingdom, 1998, pp. 191-195.

[3] Grossschmidt G., Vanaveski J., and Harf M., 2000, "Simulation of Hydraulic Chains using Multi-pole Models in the NUT Programming Environment", Proc. of the 14th European Simulation Multiconference ESM’2000, Simulation and Modelling: Enablers for a better quality of life, May 23-26, 2000, Ghent, Belgium, pp. 709-713.

[4] Grossschmidt G., and Harf M., 2003, "Multi-pole Modelling and Simulation of an Electro-hydraulic Servo-system with a Non-linear Regulator in NUT Environment", Fünftes Deutsch-Polnisches Seminar “Innovation und Fortschritt in der Fluidtechnik”, 18. – 19. September 2003,Warszawa, pp. 146-161.

[5] Grossschmidt G., and Harf M., 2005, "Modelling and Simulation of Hydraulic Load-Sensing Systems using Object-oriented Programming Environment", Proc. of the 19th European Conference on Modelling and Simulation ECMS'2005, "Simulation in Wider Europe", June 1-4, 2005, Riga, Latvia, pp. 605-609.