Direct Steady State Calculations of Hydraulic Power Systems

Dieter Scholz
Applied Science
Seevering 53
D - 21629 Neu Wulmstorf

Email: Scholz@tu-harburg.d400.de
http://www.tu-harburg.de/fst/FST.Scholz.html

Jörg Engelhardt
Aircraft Systems Engineering
Technical University Hamburg-Harburg
Neßpriel 5
D - 21129 Hamburg

Email: Engelhardt@tu-harburg.d400.de

Abstract

This paper presents an approach for the steady state calculation of hydraulic power systems. The Linear Theory Method with *p-q*-Equations is selected. It allows the calculation of networks that may form general loops. A set of equations can be generated automatically from the topology of the hydraulic system. Energy converting components are calculated separately from the network. However, the energy converting components show their influence on the network calculation by way of their boundary conditions. The work presented led to the programming of a computer tool called ICaROS (**I**nteractive **Ca**lculations of Hyd**r**aulic **P**ower **S**ystems). This tool showed the feasibility of the proposed methodology.

Keywords: Hydraulic Network, Fluid Power, Steady State, Linear Theory Method.

1 Nomenclature

A	reference area in component	[m²]
C_x'	linearised conductance, Eq. (7), (10)	
K	bulk modulus	[bar]
K_C	damping coefficient	[-]
n	exponent: $\Delta p = R_x \cdot Q_x^n$	[-]
p , p_j	pressure in node	[bar]
Δp	corrective pressure at node	[bar]
Δp_x	pressure loss in component	[bar]
q , q_j	external flow into node	[l/min]
Q , Q_x	internal flow through component	[l/min]
ΔQ	corrective flow in loop or pseudo loop	[l/min]
R_x	resistance, Eq. (15)	
R_x'	linearised resistance, Eq. (9)	
Re	Reynolds number	[-]
V	oil volume in component	[m³]
ρ	density	[kg/m³]
ζ	restistance coefficient	[-]
ν	kinematic viscosity	[m²/s]

2 Introduction

Some fluid power systems show already in the preliminary design phase a considerable complexity. Easily, the designer can get lost in system details and an enormous amount of data. Here, a steady state calculation can help to get quickly an understanding of the dependence between main system parameters. The relationship between external parameters (actuator loads, actuator speeds) and internal parameters (pressures in nodes, flows through components and pressure losses in components) becomes quickly apparent in a steady state calculation. Furthermore, less input data is required to perform the calculation compared with a dynamic simulation. Obviously, even for small hydraulic systems the numerical demand of a steady state calculation surpasses by far the possibilities of a hand calculation. Consequently, a computer implementation is imperative and led to the programming of a computer tool based on the theory presented. Calculations with the computer tool helped to prove the feasibility of the proposed methodology. The

chosen approach is a general one and the results can be used for any fluid power system for which the assumption of internal, one dimensional, incompressible, single phase flow is appropriate. The direct steady state calculation of hydraulic *power systems* is an extension of methodologies already well established for the steady state calculation of hydraulic *distribution systems*. Hydraulic distribution systems can be categorised in this way as a subset of hydraulic power systems.

3 Network Fundamentals

3.1 Type of Networks

Hydraulic networks can be grouped into different categories:

- serial networks,
- branching networks,
- parallel-serial networks,
- general looped networks.

Serial networks consist simply of hydraulic elements connected one after the other. Generally they have one source (node with known pressure), one sink (node with an unknown pressure) and more intermediate nodes (nodes with no external flow). **Branching networks** are tree-like networks without any loops. Usually they have one source, intermediate nodes and more than one sink. **Parallel-serial networks** consist of combinations of elements and subassemblies connected either in series or parallel. In contrast to branching networks they can form loops but these loops can easily be solved by combining parallel and serial elements. **General looped networks** can contain multiple sources and bridge connections. The steady state calculation of serial, branching and parallel-serial networks is usually straight forward [STREETER 85]. The steady state calculation of general looped networks, however, requires special methods.

3.2 Network Equations

Direct steady state calculations of general looped networks for hydraulic power systems are all based on nonlinear or linearised node equations

$$\sum_x Q_x = 0 \tag{1}$$

and/or loop equations

$$\sum_x \Delta p_x = 0 \quad . \tag{2}$$

These equations form the base for the network equation known as

- Q-Equations (based on flows in components / node and loop equations),
- p-Equations (based on the pressure in all sinks / node equations),
- ΔQ-Equations (based on corrective flows in basic and pseudo loops / loop equations),
- Δp-Equations (based on corrective pressures in each sink / node equations).

These equations have been presented in detail in [SCHOLZ 95].

3.3 Algorithms for Direct Hydraulic Steady State Calculations

A general hydraulic network is described by a set of nonlinear equations as named above. This set of nonlinear equations can only be solved iteratively. Three approaches have been used extensively:

- the Hardy-Cross-Method,
- the Newton-Raphson-Method,
- the Finite-Element-Method also known as Linear-Theory-Method.

The result of an investigation and comparison [BHAVE 91], [SCHOLZ 95] showed the Linear Theory Method to be superior to the other methods.

3.4 Other Approaches to Obtain Hydraulic Steady State Solutions

In principle, a steady state solution of a hydraulic system can also be obtained from a dynamic simulation after simulating for such a long time that all transients have vanished. Two different approaches to a dynamic simulation are commonly used:

- A dynamic simulation can be based on the continuity equation from which the *pressure build-up in each component* is calculated:

$$\dot{p} = \frac{K}{V} \cdot \left(\Sigma Q_{in} - \Sigma Q_{out} \right) \quad . \tag{3}$$

  In order to obtain a steady state solution, the input signal has to be a low-gradient ramp input or a sequence of small steps. [LI 93] reports that a high computational effort is required by this approach to calculate a steady state solution compared to the direct steady state calculation. [HOFFMANN 81] experienced continuous oscillations with systems characterised by low damping which precluded the dynamic simulation from reaching a steady state solution in due time.
- A dynamic simulation can be based on the momentum and continuity equation in each tube of the system. This yields a pair of hyperbolic partial differential equations that are transformed by the *method of characteristics*. The finite

difference method is then applied to solve the characteristics [WYLIE 93]. The speed of sound accounts for the propagation of the pressure waves in the tubes. With selection of a smaller speed of sound, the steady state solution is not altered, but convergence on the steady state can be achieved faster. Nevertheless, [BHAVE 91] concludes

> *... it is inefficient to employ the unsteady flow approach for steady state problems because the unsteady approach takes considerably more computational effort and time and has convergence problems.*

4 *p-q*-Equations Solved by the Linear-Theory-Method

4.1 *p-q*-Equations

The so-called *p-q*-Equations are modified *p*-Equations. *p-q*-Equations solve for unknown external flows *q simultaneously* with unknown nodal pressures *p*. In contrast, *p*-Equations require unknown external flows *q* to be calculated *separately* from the nodal pressure *p*. For that reason, applying *p-q*-Equations, nodes may exist with unknown external flow *and* unknown nodal pressure as long as the whole network remains solvable [SCHOLZ 95]. Unfortunately, as a result *p-q*-Equations produce a larger system of equations. In contrast to loop equations, node equations, like *p-q*-Equations, lend themselves much more readily for modern computer applications with an interface allowing interactive graphic manipulations of the network and automatic generation of the system of equations.

p-q-Equations are written for every node in the system. Detailing Equation (1):

$$\sum_{x=1}^{k} Q_x + q_j = 0, \qquad \text{for all nodes } j = 1, \ldots, J \quad . \tag{4}$$

The J nonlinear equations are used to solve for N unknown pressures p and M unknown external flows q. With

$$Q_x = \left(\frac{p_i - p_j}{R_x} \right)^{\frac{1}{n}} \tag{5}$$

Equation (5) can be written as

$$\sum_{\text{i connected with j by x}} \left(\frac{p_i - p_j}{R_x} \right)^{\frac{1}{n}} + q_j = 0, \quad \text{for all nodes } j \tag{6}$$

Usually $n = 2$ is chosen. Introducing the linearised conductance

$$C_x' = \frac{|p_i - p_j|^{\frac{1}{n}-1}}{R_x^{\frac{1}{n}}} \tag{7}$$

converts the set of nonlinear equations into a set of linear equations

$$\sum C_x' (p_i - p_j) + q_j = 0, \quad \text{for all nodes } j = 1, \ldots \tag{8}$$

4.2 Initialisation of the Iteration

Since C_x' depends on the unknown p_j , the system of equations can only be solved iteratively. The linearised conductance C_x' has to be guessed for the first iteration step. The Linear Theory Method converges no matter which start-up values were selected. A good guess, however, speeds up convergence. With the linearised resistance

$$R_x' = R_x \, |Q_x|^{n-1} \tag{9}$$

the linearised conductance can be obtained from

$$C_x' = \frac{1}{R_x \cdot |Q_x|^{n-1}} \quad . \tag{10}$$

Initialisation of C_x' with Equation (10) can be done by automatically specifying for each component either

- an initial flow Q_x [ISSACS 80],
- an average flow velocity v_x [ISSACS 80] and hence $Q_x = v_x \cdot d^2 \cdot \pi / 4$,
- a Reynolds number Re [COLLINS 75] and hence $Q_x = \pi/4 \cdot Re \cdot d \cdot \nu$.

4.3 Damping of the Iteration

From the second iteration step onwards, the linearised conductance ${}_{t+1}C_x'$ for the Iteration step $t + 1$ is calculated form

$${}_{t+1}C_x' = \frac{|{}_tp_i - {}_tp_j|^{\frac{1}{n}-1}}{R_x^{\frac{1}{n}}} \quad . \tag{11}$$

It has been observed [COLLINS 75] that the linearised conductance can oscillate about the correct value during the iteration. By introducing a damping factor K_C , convergence can be reached faster. A linearised conductance ${}_{(t+1),in}C_x'$ for entering iteration $t + 1$ is then

calculated from a linearised conductance at the beginning (${}_{t,in}C_x'$) and at the end (${}_{t,out}C_x'$) of iteration t

$$ {}_{(t+1),in}C_x' = {}_{t,out}C_x' + K_C \cdot \left({}_{t,in}C_x' - {}_{t,out}C_x' \right) \quad . \tag{12} $$

The damping factor is defined in the range $0 \leq K_C < 1$. [ISSACS 80] and [COLLINS 75] propose $K_C = 0.5$. Own calculations indicate that usually $K_C < 0.5$ produces faster convergence.

4.4 Automatic Generation of a Set of *p-q*-Equations

For generating a set of *p-q*-Equations, in **Step 1** a system of equations is generated based on Equation (8). In **Step 2** the system of equations is sorted for known and unknown values.

Step 1: *Generation of the unsorted set of equations.*
The unsorted set of equations has the form $A_1 \ \mathbf{x_1} = \mathbf{b_1}$. A_1 is called the system matrix and is composed of the linearised conductances. $\mathbf{x_1}$ includes the pressures and $\mathbf{b_1}$ includes the external flows of each node. Three rules can be formulated which allow the generation of the unsorted set of equations:

- The elements on the main diagonal a_{ij} with $i = j$ are calculated as sum of the conductances of those elements which meet in node i.
- The off-diagonal elements a_{ij} with $i \neq j$ are zero if node i and j are not connected. Otherwise, the elements a_{ij} are equal to the negative value of the linearised conductance connecting nodes i and j.
- The system matrix is symmetrical.

The set of equations can hardly be solved directly because a pressure p_j has to be known for at least one node in a hydraulic network and also an external flow q_j has to be unknown for at least one node.

Step 2: *Dividing known and unknown values.*
The set of equations has to be sorted in such a way that know and unknown values are properly divided. From $A_1 \ \mathbf{x_1} = \mathbf{b_1}$ it is generated $A_2 \ \mathbf{x_2} = \mathbf{b_2}$. For the formalised procedure of dividing known and unknown values, again a set of rules can be formulated which allows the generation of the sorted set of equations $A_2 \ \mathbf{x_2} = \mathbf{b_2}$.

- Generation of vector $\mathbf{x_2}$:
  $\mathbf{x_2}$ includes the J unknown of the system. The first N lines of vector $\mathbf{x_2}$ are allocated to the N unknown pressures p in the same sequence as allocated in the vector $\mathbf{x_1}$. The remaining M lines of vector $\mathbf{x_2}$ contain the M unknown external flows q in the same sequence as allocated in vector $\mathbf{b_1}$.

- Generation of matrix A_2 :
  - The N columns of the system matrix A_1 which are allocated to the unknown pressures are taken as columns 1 to N in matrix A_2 .
  - The remaining M columns $N + 1$ to J in matirx A_2 are allocated to the unknown external flows according to the ordering in vector $\mathbf{x}_2$. Each of these columns of matrix A_2 contains -1 in the line that was originally occupied by the external flow q in vector $\mathbf{b}_1$. The remaining lines of column $N + 1$ to J in matirx A_2 are set to zero.
- Generation of vector $\mathbf{b}_2$:
  - A vector $\mathbf{b}_1^*$ with J lines is generated. If the line of the original vector $\mathbf{b}_1$ was occupied with a known external flow, then this external flow is also carried over into vector $\mathbf{b}_1^*$. The remaining lines are set to zero.
  - Based on matrix A_1 a matrix A^*_1 is generated that only includes the columns of known pressures and preserves the sequence of columns in A_1 .
  - The vector $\mathbf{x}_1$ is converted into a vector $\mathbf{x}_1^*$ with N lines that include the known pressures.
  - The vector $\mathbf{b}_2$ is calculated from

$$\mathbf{b}_2 = \mathbf{b}_1^* - A_1^* \cdot \mathbf{x}_1^* \quad . \tag{13}$$

4.5 Solving the Set of Equations

The set of equations

$$A_2 \, \mathbf{x}_2 = \mathbf{b}_2 \tag{14}$$

can now be solved. Gaussian elimination with scaling and partial pivoting is the fastest method for systems with up to 100 equations. For bigger systems an iterative procedure can be faster [HOFFMAN 92]. Relaxation methods require however a diagonal dominant matrix. A diagonal dominant matrix can be generated if the nodes in the hydraulic system are internally renumbered in a way that adjacent nodes i and j will be allocated numbers i, j with $|i - j|$ is as small as possible.

The linearised resistances R_x' (Equation (9)) and conductances C_x' (Equation (10)) depend on the flows Q through the components. Consequently, the linearised conductances have to be recalculated after each step during the iteration:

- Constant resistances: C_x' can simply be recalculated using Equation (11).
- Resistances with a variable resistance coefficient ζ: The Reynolds number is calculated from the new internal flows Q . The resistance coefficient ζ is calculated depending on the way the component is modelled (see below). R_x is calculated from

$$R_x = \zeta \cdot \frac{\rho}{2 \cdot A^n} \tag{15}$$

and R_x' from Equation (9) and finally C_x' again from Equation (10).

- Components that are characterised by a Δp-Q relationship: C_x' is calculated from

$$C_x' = Q \,/\, \Delta p \quad . \tag{16}$$

The system matrix A_2 is provided with these new conductances and the set of equations is solved again.

The iteration concludes if a minimum number of iterations have been performed and the unknown pressures and external flows being calculated do not change significantly anymore from one iteration to the next one:

$$\left| \frac{\left({}^{t+1}x_{1_i} - {}^{t}x_{1_i} \right)_{max}}{{}^{t+1}x_{1_i}} \right| \leq \epsilon \quad . \tag{17}$$

5 Integration of Hydraulic Components

So far, in the method presented, components have only been considered as *generic* components. In reality components have to be modelled individually depending on their physical appearance and the amount of information being available to model the components.

5.1 Resistance Components

Simple hydraulic resistances are tubes, bends, orifices, contractions and expansions. These components are characterised by a resistance coefficient ζ. The Resistance R_x is calculated from Equation (15). The resistance coefficient depends only on the geometry of the component and the Reynolds number. The empirical relationship between geometry, Reynolds number and resistance coefficient can be taken from handbooks like [IDEL'CHIK 94].

In practice many components exist which have a unique shape so that hydraulic characteristics cannot be obtained from handbooks. Various models for the resistance of these components can be applied to incorporate the measurements that often exist with different level of detail:

- Modelling a ζ-Re-relationship by
  - o a constant ζ-value,
  - o a spline $\zeta = f(Re)$,
  - o calculating ζ according to [WILL 95] from $\zeta = k_1 / Re + k_2$.

  Evaluation: R_x from Equation (15) and C_x' from Equation (11).
- Modelling a Δp-Q relationship by
  - o a spline $\Delta p = f(Q)$
    evaluation: $C_x' = Q / \Delta p$,
  - o a pair $(\Delta p_{nom} ; Q_{nom})$
    evaluation: $R_x = \Delta p_{nom} / Q_{nom}^2$, $C_x' = const$ from Equation (11).

5.2 Valves

Implementation of Check Valves, Pressure Relief Valves, and Priority Valves. The steady state characteristics of the valves are entered into the program in form of Δp-Q-diagrams. Additionally, the opening pressure is required. For a pressure differential greater than opening pressure, the valve is treated like a normal hydraulic resistor. In the other case, the modified conductance C_x' is set to zero.

Implementation of Four-Way Servo Valves. A servo valve can be modelled from four variable hydraulic resistors in bridge connection. These four variable hydraulic resistors can automatically be generated by the computer according to the selected valve opening. The required valve opening is either given or can be iteratively determined (if e.g. hydraulic systems are tested by demanding certain motor speeds and actuator piston velocities at given loads). In the latter case, the first iteration starts at maximum valve opening. If the resulting actuator speed is less than required, no practical solution is possible. Otherwise, the valve setting will automatically be reduced to achieve the demanded actuator speed at given load.

5.3 Energy Converting Components

For fluid power systems, energy converting components can be modelled with a high degree of flexibility if the energy converting components are calculated separately from the rest of the network and merely the interface to these components with the remaining system is managed. This will be demonstrated for some selected components:

Implementation of Pumps. Five different pump models are considered:

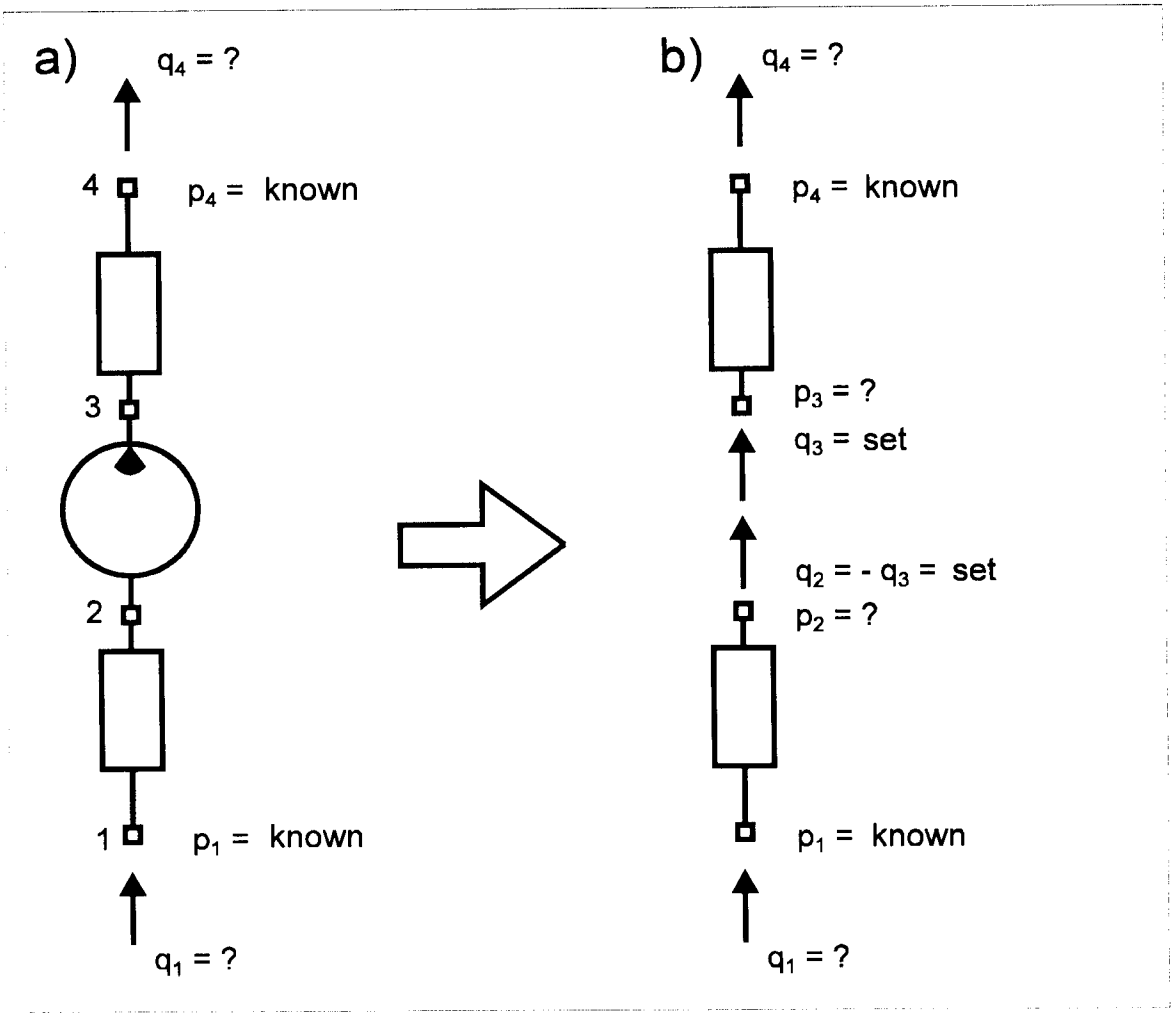
1.) pumps with constant overall efficiency;

2.) pumps with variable overall efficiency depending on the pressure increase over the pump, Δp, and pump speed n calculated from $\eta = f(\Delta p, n)$;

3.) pumps described by Δp-Q-diagrams;

4.) Bavendiek-modelled pumps [BAVENDIEK 87] with the efficiency depending further on oil viscosity ν and swashplate setting α : $\eta = f(n, \Delta p, \nu, \alpha)$;

5.) pressure regulated pumps.



Firgure 1: Pump Cut Off a Simple Network for Separate Calculation

Figure 1a shows a pump in a simple network. The pump is connected on both sides to a pipe. Let us assume that node 1 and 4 are linked to a reservoir. Hence, the pressure in node 1 and node 4 are known, however, the external flows are unknown. The pump is now taken off the system and considered separately. Two separate networks emerge with two new boundary nodes. For pump models 1.) through 4.), these new nodes are taken as having unknown pressure. Assuming no external leakages, effective outlet flow must equal effective inlet flow: $q_3 = - q_2$ (**Figure 1b**). For each iterative step during network

calculation, a procedure as follows is initiated:

1.) calculate a new pump efficiency η depending on the pump model (for the first iteration an efficiency will be assumed);
2.) calculate an effective pump flow Q_{eff} from pump efficiency, pump displacement, and pump rotational speed;
3.) set external flows on new boundary nodes: $q_3 = Q_{eff}$ and $q_2 = -q_3$;
4.) calculate Δp for the next calculation of pump efficiency;
5.) perform the next iteration step of the network calculation.For pressure regulated pumps, p_3 and q_2 in Figure 1b are considered to be known with $q_2 = - Q_{eff}$. Accordingly, the calculation procedure changes slightly. Q_{eff} and pump outflow pressure p_3 taken from the last two iterations are averaged. This introduces damping to the iteration and ensures a stable solution.

Implementation of Hydraulic Motors. Three different motor models are considered:

1.) motors with constant efficiency;
2.) motors with variable efficiency depending on pressure drop Δp and motor speed n: $\eta = f(\Delta p, n)$;
3.) Bavendiek-modelled motors [BAVENDIEK 87] with efficiency depending further on oil viscosity ν and swashplate setting α : $\eta = f(n, \Delta p, \nu, \alpha)$.

Again, this component is calculated separately from the rest of the network. At motor outlet, the external flow q_3 of boundary node 3 and at motor inlet pressure p_2 are considered to be known. For each iterative step during network calculation, a procedure as follows is initiated:

1.) calculate a new motor overall efficiency η depending on the motor model (for the first iteration an efficiency will be assumed);
2.) calculate an average outlet pressure p_3 from the last two iterations; calculate motor inlet pressure p_2 from outlet pressure and required differential pressure Δp due to motor loading;
3.) set $q_3 = - q_2$ and effective flow $Q_{eff} = q_3$;
4.) calculate motor speed n from effective flow Q_{eff} , motor displacement, and volumetric efficiency;
5.) calculate the required differential pressure Δp (used in Step 1) from hydromechnical efficiency, effective torque, and geometric displacement;
6.) perform the next iteration step of the network calculation.

The calculation of linear actuators follows the same principles as given for the hydraulic motor.

6 The Program ICaROS

ICaROS stands for **I**nteractive **Ca**lculations of Hyd**r**aulic **P**ower **S**ystems. ICaROS was programmed to check the ideas presented in this paper. The program can be used in batch operation with input files describing the hydraulic system and ist components in question. A much more intuitive access to the program can be obtained by using the graphical user interface that allows a graphical and direct manipulation of the hydraulic system being considered. A screen shot of the ICaROS user interface is given in **Figure 2.**

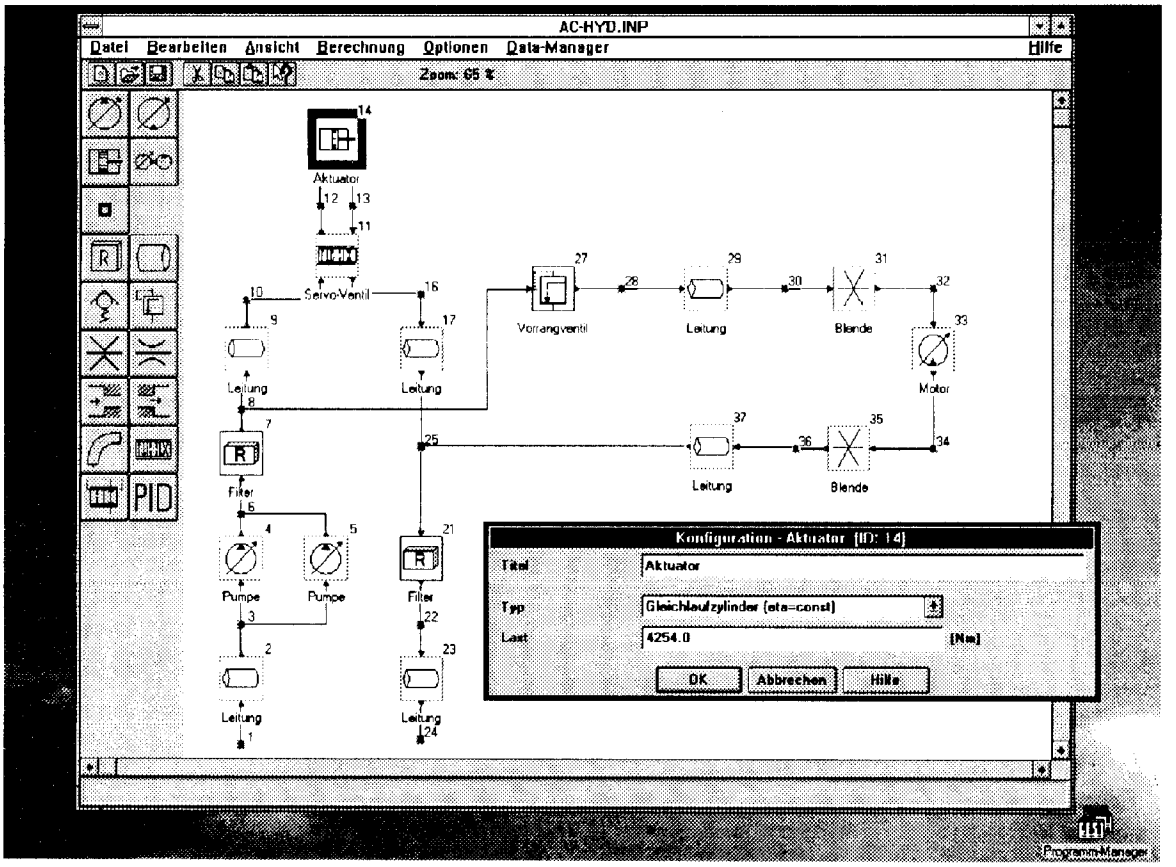


Figure 2: ICaROS, Graphical User Interface

7 Conclusions

This paper has presented an approach for the steady state calculation of hydraulic power systems. A steady state calculation is useful to get a first overview of system parameters with little input data requirements. The Linear Theory Method was selected for the

calculation of networks forming general loops. A set of equations can be generated automatically from the topology of the hydraulic system. Energy converting components are calculated separately from the network calculation. However, the energy converting components influence the network calculation by means of their boundary conditions. The feasibility of the proposed approach has been tested with a computer program called ICaROS.

8 Literature

[BAVENDIEK 87] BAVENDIEK, R.: *Verlustkennwertbestimmung am Beispiel von hydrostatischen Maschinen in Schrägachsenbauweise*, VDI-Forschungsbericht, Reihe 7, No. 122, Düsseldorf : Verein Deutscher Ingenieure, 1987.

[BHAVE 91] BHAVE, P.R.: *Analysis of Flow in Water Distribution Networks*, Lancaster : Technomic, 1991.

[COLLINS 75] COLLINS, A.G.; JOHNSON, R.L.: Finite-Element Method for Water-Distribution Networks. In: *Journal American Water Works Association*, 67(7), 1975, 385-389.

[HOFFMANN 81] HOFFMANN, W.: *Dynamisches Verhalten hydraulischer Systeme, automatischer Modellaufbau und digitale Simulation*, Aachen, Rheinisch-Westfälische Technische Hochschule, Fakultät für Maschinenwesen, Diss., 1981.

[IDEL'CHIK 94] IDEL'CHIK, I.E.: *Handbook of Hydraulik Resistance*, Boca Raton : CRC Press, 1994.

[ISSACS 80] ISSACS, L.T.; MILLS, K.G.: Linear Theory Methods for Pipe Network Analysis. In: *Journal of the Hydraulics Division*, American Soc. of Civil Engineers, 106(HY7), 1980, 1191-1201.

[LI 93] LI, X. ET. AL.: Unity Between Steady and Transitional State Simulation for Hydraulic Systems. In: JAPANESE HYDRAULICS AND PNEUMATICS SOCIETY (Organizer): *Proceedings of the Second JHPS International Symposium on Fluid Power*, JHPS, 1993, 545-550.

[SCHOLZ 95] SCHOLZ, D.: *Computer Aided Engineering zur Auslegung von Flugsteuerungs- und Hydrauliksystemen*, (10. Fachtagung Hydraulik und Pneumatik, 5th and 6th October 1995), Dresden : Technische Universität, 1995.

[STREETER 85] STREETER, V.L.; WYLIE, E.B.: *Fluid Mechanics*, New York: McGraw-Hill, 1985.

[WILL 95] WILL, D.; WILL, R.; WILL, T.: HydCalc - Hydraulikberechnungen unter Windows. In: *O+P Ölhydraulik und Pneumatik*, Limburg : Vereinigte Fachverlage, 39, No.1, 1995, 27-31.