In recent years very powerful means have appeared for programming intended for the imitative simulation of systems. They have given the possibility of effectively investigating the behavior of diverse classes of complex systems and of obtaining interesting information on their functioning processes.

However, rather frequently the need arises for selecting the conditions or parameters characterizing the system, which would ensure that running of the processes being investigated is the best in some specific sense. The solving of problems of such type as applied to systems with a significant number of variable parameters is exceptionally complex and in practice is often reduced to a multiply repeated cycle of simulation, analysis, and estimation of the data obtained, and the parameter corrections. Such a solution method, however, not only is too long and unproductive from the point of view of computer time expended, but also may not always lead to satisfactory results. It is difficult to justify the results thus obtained, since it is not possible to determine to what degree the quality of functioning of the system can be improved further, etc. Therefore, the problem of the program realization of software procedures for a purposeful sorting of the variants for their successive improvement and optimization (the procedure of directed imitative simulation) is a very urgent one.

The problem of estimating the system's parameters can formally be treated as a problem of mathematical programming. We denote the collection of parameters with respect to which the system is to be optimized by the vector $\mathbf{x}$; by $F(\mathbf{x})$ we denote the value of the system's performance quality function, realized for a given collection $\mathbf{x}$ of parameters. Then we have

$$F(\mathbf{x}) \rightarrow \text{min}$$  \hspace{1cm} (1)

under the constraints

$$\mathbf{x} \in X \cap X',$$  \hspace{1cm} (2)

where $X = \{\mathbf{x} : f_i(\mathbf{x}) \leq 0; i = 1, 2, \ldots, n\}$ are constraints given explicitly in the form of inequalities; $X'$ is the set $\{\mathbf{x}\}$ given implicitly (the membership of a vector of argument $\mathbf{x}$ to this set can be ascertained only during the simulation).

The presence of such (similar to X') sets is typical precisely for the problems solvable by imitative simulation means. Let us clarify the nature of these sets by the following illustrative example.

Let a network of computational centers having N points of data receiving and processing effect the control of a certain process in real time. The dynamic characteristics of the process are such that the data processing is required to be terminated no later than by a time T after it has been fed into the network.

When an inquiry comes in at any point the message is processed at the given center and the results are returned to the user. However, in case the computational centers at the place the inquiry comes in are occupied the information is directed to those of the computational centers for which the probability of fastest processing of the order at the given instant is maximal. If the situation with respect to the receiving of the message by the new center has changed, it can once again be transmitted to another point. The total time for processing the message is thus made up of the waiting time in the queue and of the total time of transmitting the order over the network's channels.

The proper allocation of the computational resources available over the distributed network is a typical and very often complex problem.

Let us give a more general statement of the problem, suitable for a significant circle of problems of similar type.

Let there be a queuing system represented by a graph G(I, U) with a vertex set I and an arc set U. There are x_s service facilities attached to each vertex i_s ∈ I, s = 1, 2, ..., n. We denote their productivity by c_s.

At the graph's vertices there are order generators with execution rates y_s(ω) [where y_s(ω) are functions measurable on some probability space]. The orders originating at the i_s-th vertex can either be serviced by the available facilities or, if this is not possible, be directed to the system where they wander about in accordance to a specified scenario in expectation of being serviced. Thus, at the i_s-th vertex there enters orders both generated at the vertex as well as sent to it from the system [we denote them z_s(ω)].

Now if y_s(ω) + z_s(ω) > c_s · x_s, then the unserviced orders are once again returned to the system. The total number of orders directed into the system by vertex i_s is determined by the expression max{0, y_s(ω) + z_s(ω) - c_s · x_s}.

The problem consists in the distribution of the servicing facilities x_i over the graph's vertices in such a way that some target function F(x_1, ..., x_n) is minimized under the fulfillment of the constraints imposed on the vector x = (x_1, x_2, ..., x_n).

In the case given the constraints X are specified by simple relations of type a_i ≤ x_i ≤ b_i, i = 1, ..., n, as well as \( \sum_{i=1}^{n} x_i < A \).

However, the set X' is determined by the specific nature of the requirements on the system's functioning. An example is the requirement to limit the maximal message delay in the system to some amount T.

It is clear that the membership of a vector \( \bar{x} \) in the set described can be verified only during the simulation, which emphasizes the specific nature of similar problems. The nonfulfillment of constraints X' is classified as a situation leading to a violation of the imitative model.

Let us note a circumstance that must be taken into account when developing the algorithms and methods for solving problems of the type being considered.

As a rule the imitative models of real systems are very complex and tens of seconds of machine time on modern electronic computers are required for their single simulation [i.e., for the computation of F(x) for a given \( \bar{x} \)]. Even if we assume that F(\( \bar{x} \)) is continuously differentiable with respect to \( \bar{x} \) (as usually happens in optimal control problems), then in this case neither Newton-type methods nor the usual gradient methods can be used. Indeed, when the dimension of vector \( \bar{x} \) equals 30, it is necessary to compute the value of F(\( \bar{x} \)) at 31 points, i.e., to simulate the system that many times, in order to estimate the derivative. It is clear that such problems cannot be solved by a purely program method.

A number of authors, particularly, V. M. Glushkov, Yu. M. Ermol'sev, T. P. Mar'yanovich, and N. Z. Shor, have repeatedly noted the prospects of attempts to solve problems similar to those described, in a man-computer dialogue mode. In this case the possibility emerges of using during the solving of the problem not only the formal solution methods, whose convergence has been proved, but also the experience and intuition of