Використання Шаблонів для Структур Вхідних Даних в Оптимізаційно-Імітаційних Експериментах

Віра Бігдан, Олена Криковлюк
Інститут кібернетики НАНУ ім.В.М.Глушкова
Київ, Україна
vbigdan@icfcst.kiev.ua

Applying of variables templates for input data structures in Optimization Simulation Experiments

Vira Bigdan, Olena Krykovliuk
Institute of Cybernetics after V.Glushkov of NAS of Ukraine
Kyiv, Ukraine
vbigdan@icfcst.kiev.ua

***Анотація*—досліджується запропонований підхід до уніфікації вхідних даних змішаного типу для оптимізаційно-імітаційних експериментів. Запропоновані шаблони були розроблені для оптимізатора, який використовує еволюційні та мультиагентні стратегії. Тестування проводилося на кластері СКІТ3.**

***Abstract*—this research analyzes proposed approach to mixed type input data unification in optimization and simulation experiments. The proposed templates were developed for optimizer based on evolutionary and multiagent strategies. Testing and verification was done on cluster CKIT3.**

Ключові слова—шаблони даних, стохастичні алгоритми оптимізації, імітаційний експеримент, дискретизація, змішаний тип змінних, кластер

Keywords—data templates, stochastic optimization algorithms, simulation experiment, discretization, mixed-value variables, cluster

I. INTRODUCTION

Institute of Cybernetics of the National Academy of Sciences of Ukraine accumulated certain experience in analyzing complex stochastic systems with the use of the developed NEDISOPT_D system. This system integrates capabilities of simulation methods, optimization and distributed computing technologies. The system optimizer provides directional search in selected optimal modes, using evolutionary and multi-agent strategies. The team also developed parallel versions of optimized strategies which are using cluster architectures [1].

At present further reseach is done on developing the capabilities of the OPTIMIZER module in simulation systems using the architecture similar to NEDISOPT_D.

When choosing methods of optimization, it should be considered that the simulation model of the system being analyzed is a "black box". The input data is a vector of values with a limited set of input factors, and on the output there is a vector containing values of model responses [2,3]. The researcher makes conclusions based on the analysis of output parameters of the model as to which input vector provided more "optimal" functioning of the system according to certain criteria or a set of criteria [3].

Simulation systems are used for modeling diverse range of complex stochastic systems. Therefore, the input data for the created simulation models can be completely different. Firstly, the input data can be quantitatively different, and secondly, the variables themselves can be of a completely different nature: numerical (discrete, continuous, mixed-value variables) or categorical (logical, various categories of devices and equipment with different parameters and properties).

II. UNIFICATION OF INPUT DATA STRUCTURES

The effectiveness of research and design of complex stochastic systems based on technology of optimization-simulation integration can be encreased by bringing the original data of simulation model to a unified form using the developed templates. Such template allows to reduce the time of subsequent optimization and provides opportunities for re-use of large data amount due to the possibility of a simple revision of the data structure and provides a convenient user interface for working with input data. This approach is based on the concepts of optimization and simulation integration and meta-heuristic optimization strategies. Software environment for its support is represented by OPTIMIZER of NEDISOPT-D architecture.

The admissible values for variables of the concrete model (the input data for OPTIMIZER) can be specified in various ways:

- by a finite set of user-defined values;
- by the minimum and maximum values of the numerical interval from which all values are allowable or with some sampling discretization step [4].

Optimization experiments with models use heuristic stochastic algorithms that work with standard built-in data structures for the system [4, 5]. OPTIMIZER of the system assumes that all data that is used in optimization algorithms and is input to these processes is considered as real. Before optimization begins, the process of digitization of real continuous data is performed. At the same time the limit values are determined for the factors values.

Unification of data increases the effectiveness of simulation experiments in case of using an ensemble of optimization algorithms [6]: for the exchange of data processes on a cluster arrays of the same type are used.

Different types of data must be brought to some universal type for the effective use of optimization algorithms [3]. For operation of algorithms of various optimization strategies, it is necessary to specify a set of values of input parameters that can be divided into two groups. The first group of input parameters provides information about factors of the problem, examines their number, maximum number of quantization levels of any factor, type of each factor (integer or real), type of permissible values (continuous, discrete or bounded set of values), range of the allowed range of values, step sampling (for continuous) or multiple values.

The second group of input data sets the values of the control parameters for each algorithm. Common parameters for all algorithms are the size of population (or the number of individuals for multi-strategy strategies) and the limitation on number of sampling levels.

The number of quantization levels and the sampling step are set by a researcher of the model. Input data which is the same for all strategies is provided separately it represents always real numbers (conversion into a valid type occurs automatically) and parameters for the selected strategy are also specified separately.

To enter input data that is intended to be used as variables in optimization simulation experiments, an intermediate template is created to represent the input data. This template provides

- automatic factor number setting;
- selection type of data and the way it is set using the appropriate switches.

After entering the data for all factors of the optimization experiment, an internal data structure of a single format is formed for all optimization strategies that will be used as input by OPTIMIZER for the chosen strategy. In this structure there is a factor number, the number of levels of its quantization, parameters of the quantization level of the factor (maximum, minimum, step).

III. Unified Scheme for Implementation of Optimization-Simulation Experiments

A specifically developed unified scheme for the implementation of optimization simulation experiments provides necessary flexibility in study of complex systems within the framework of the NEDISOPT_D [7] architecture. The scheme defines the methodology of planning and technology for the implementation of multi-stage optimization-simulation experiments.

In general case, each study with simulation model should include three stages, which are implemented sequentially. At the first stage (modeling), significance of the input data of the model (factors) that were originally introduced and related to the selected meta-heuristic optimization strategy is investigated. Factors are selected in a way to fully describe the simulated system. At this stage, it is possible to simplify the model, by eliminating those factors that are not significant. At the second stage (simulation), tasks of tactical planning of simulation experiments are determined. At the third stage (replication) an estimation of reliability of the results of experiments is carried out.

The development of scenarios for the first stage simulation experiments is determined by strategic planning algorithms, which essentially depend on the adopted strategy for finding optimal solutions. For the meta-heuristic optimization strategies, the main issues are:

- to define evaluated alternatives, characteristics of which form composition and structure of decisions;
- to form the initial population or to identify the number of individuals, taking into account the limits and quantization levels of the factors determined at the first phase of the experiment;
- to determine control parameters for the process of searching for optimal solutions depending on the optimization strategy used;
- to define responses and values of fitness functions that characterize the estimated and optimal alternatives.

Carrying out of researches at the second stage requires to solve the following tasks:

- to classify the response-output of the simulation model and to form the main and additional set of responses. Composition of the main set, as a rule, includes indicators of effectiveness of the functioning of systems which are being analyzed. These indicators are used in strategies for finding optimal solutions. The additional set includes responses containing various sorts of detailed information necessary for more accurate understanding of the specifics of functioning of system which is being analyzed;

- to identify (allocate) significant (dominant) factors that have the greatest influence on the main responses of the simulation model;
- to determine the initial conditions for experiments;
- to determine the composition of control and input information for purely imitative runs;
- to verify and validate the simulation model. Note that regardless of the optimization strategy being used, the search for optimal solutions can be carried out only on the basis of the model for which the verification and validation procedures are being performed [5-7];
- to analyze sensitivity of the simulation model and to define constraints and to set permissible intervals for responses and factors changing;
- to determine the confidence interval for modeling, the interval of the model overclocking and the step with which the modeling process will be carried out;
- to define formulas for calculation of target functions on the basis of responses of corresponding runs of the simulation model, factors, all sorts of cost and penalty characteristics.

The third stage is the stage of replication runs and is connected with solving the issues of validation of modeling results and is intended to confirm reliability and stability of the obtained results.

IV. OPTIMIZER STRATEGIES

The development of scenarios for simulation experiments for the first stage is determined by algorithms of strategic planning, which essentially depend on the strategy of finding the optimal solutions. OPTIMIZER which is being developed:

- has no permanent connection with the simulation model;
- the target function is set by user and can include both input and output variables;
- the range and step of changing the parameter values is set;
- there is a possibility for user to set the initial plan of the experiment;
- uses both evolutionary and multi-agent optimization strategies;
- the criteria for stopping the optimizer's work is to specify the total number of estimated alternatives or the time limit;
- one simulation session runs with a fixed selected values of input parameters;
- the process of finding optimal solutions is logged;
- the optimization strategies work both under Windows and under Linux;
- the web interface is being developed.

In the first version of OPTIMIZER, meta-heuristic strategies were used, which were based on the classical genetic algorithm [9, 10]. Each generation of population of chromosome solutions has its own vector of fitness values. These values are calculated on the basis of the expression for the fitness function of the formulated optimization task (problem), taking into account the responses of the corresponding simulation model.

Now the evolution strategy is supplemented by new strategies: evolutionary algorithms are extended by the island model of genetic algorithm [11, 12], as well as multi-agent strategies based on the algorithm of ant colony, algorithm of particle swarm and the algorithm of bee colonie are added [9]. The iterative process in evolutionary algorithms is based on changing the population of chromosomes by applying the crossover and mutation operators. The function of the first one is to preserve the genetic material – chromosomes accumulated in the population. The size of population is limited, and the mutation operator allows to search for new chromosomes on the whole space of allowed values. The island model of genetic algorithm is based on multi-population models. Each of these optimization strategies has its own specific structure of input data.

To test and evaluate the work of the classical genetic algorithm which was implemented, several test functions with different variables and several local and global minima were created [14]. These functions also included local and global maxima.

During implementation, the characteristics of test functions were to the maximum approximated to the behavior of the complex system model. Therefore, important characteristics of these functions are:

- stochastic nature of fitness value (to the accurately calculated function one or more random variables were added with a normal or even distribution, but with different versions of parameters);
- the presence of one or more global "extremums", the values of which for one function may differ in several runs,
- the presence of several local "extremums";
- all "extremums" are scattered randomly throughout the admissible area, without overlapping in areas of attraction;
- functions are defined (computational) within the entire permissible range, but, depending on the types of their inputs, they can be estimated (calculated) at (on) a finite set of points.

Conducting of experiments to find out the most effective values of the evolution algorithm parameters were carried out in two stages. At the first stage, the mean fitness value for each of the chromosome planes was obtained.

Only an estimate of the best chromosomes found in the population after finishing the work of the algorithm was considered [6], and it was not taken into account whether at least one local minimum was found in the process of optimizing the test function.

At the second stage, all examined chromosomes were sorted out according to their fitness values (from smaller to bigger, and for further analysis, 20 best and 20 worst chromosomes were considered in order to find probable trends in meanings of both individual parameters and their combinations with each other. As a result, conclusions were drawn for each of the parameters under investigation [11].

Experiments conducted to increase the efficiency of search for optimal solutions by changing the values of control parameters, showed that the quality of the algorithm's work greatly depends on values taken by its parameters. The analysis of the above results allowed to identify the range of values for each of the studied parameters, at which the maximum efficiency of the algorithm is achieved.

The cluster complex of Institute of Cybernetics after V.Glushkov of the National Academy of Sciences of Ukraine was used for the study.

To evaluate the effectiveness of the selected chromosome plans, the island model of evolutionary algorithm was used without exchange of chromosomes between the populations [5, 8, 11].

The experiment consisted in the simultaneous launch of 10 and 20 parallel processes (the numbers 10 and 20 were chosen for convenience). In each process an evolutionary algorithm was used with its experiment plan to be evaluated. For better results reliability, 12 replication experiments were conductedfor each plan and the arithmetical mean of fitness values of 10 average runs (skippung the best and worst) was used as the final evaluation of the set efficiency.

CONCLUSIONS

In imitation experiments, the number of possible combinations of factors is very large, even if the number of factors is small, because quantization of continuous factors can be done with an arbitrary precision. As a result, a large dimensionality problem arises for optimization tasks. Therefore, the most important task of simulation experiment is to specify key factors that should be identified as quickly and as accurately as possible from the set of possible ones, which can significantly reduce the dimensionality of optimization tasks.

Separation of input data into data for direct optimization algorithms, which are identical for all optimization strategies and bringing them to a unified form and data that are used only in a particular strategy, increases the efficiency of the optimization process. Bringing the input data to a unified form based on the developed template for input data and providing the capability of compositing various methods significantly improves the efficiency of the optimization phase.

The process of studying models of complex systems involves implementation of a large number of simulation experiments. The output data of each experiment is accumulated for further analysis. The processes of research and design of complex systems are accompanied by generation and accumulation of large volumes of heterogeneous information that forms the basis of practical modeling experience in the relevant applied field. Further analysis of the accumulated data requires the use of methods of intellectual analysis to search for useful trends/patterns and to make feasible forecasts.

ЛІТЕРАТУРА REFERENCES

[1] V.A. Pepelyaev, “Planning optimization-simulation experiments, *Kibernetika i Sistemnyi Analiz*, No. 6, November–December 2006, pp. 112–125

[2] Robert E. Shannon, “Systems Simulation: The Art and Science” *Prentice-Hall*, 1975, P.418

[3] Nikolay P. Buslenko, “Simulation complex systems”, Moscow, *Science* 1978, P.400 (in Russian)

[4] V.B. Bigdan, O.O. Krykovliuk, V.A. Pepeliaev, “Structure input data unification of Optimization Algorithms in Simulation Experiments,” *Computer mathematics*, Kyiv, Ukraine, 2017, No 1, pp. 45 – 55 (in Ukrainian)

[5] V.B. Bigdan, O.O. Krykovliuk, V.A. Pepeliaev, “Input Data Mixed-value Variables Templates for Optimization Algorithms in Simulation Experiments,” in *Modern Computer Science: Problems, Achievements and Prospects of Development:* The International Scientific Conference, Kyiv,Ukraine, 2017, pp. 185-187

[6] I.V. Sergienko, V.P. Shylo , O.V. Shylo “Algorithm Unions In Parallel Optimization,” in *Modern Computer Science: Problems, Achievements and Prospects of Development:* The International Scientific Conference , devoted, Kyiv,Ukraine, 2017, pp. 139

[7] V.B. Bigdan, V.A. Pepeliaev, Yu.M.Tchornyy. “Unified scheme for the implementation of optimization-simulation experiments,” *Problems in Programming*. Scientific Journal, Kyiv, Ukraine, No 2-3, pp. 728-733 (in Ukrainian)

[8] V.B. Bigdan, O.O. Krykovliuk, V.A. Pepelyaev, Yu.M.Tchornyy, “Control of Evolutionary Optimization Algorithms in Simulation Experiments,” *Computer mathematics*, Kyiv, Ukraine, 2016, No 6, pp. 102 – 113 (in Ukrainian)

[9] Subbotin S.O., Oleynik A.O. Oleynik O.O. “Non-iterative, evolutionary, and multi-agent methods for synthesizing indistinct logical and neural network models,” Zaporizhia, Ukraine, *ZNTU*, 2009, P. 375 (in Ukrainian)

[10] V.B. Bigdan, V.A. Pepelyaev Yu.M.Tchornyy, “Applying of genetic algorithm for optimization in simulation experiments,” in *Decision making:* VII The International school-seminar, Uzhogorod, Ukraine, 2016, pp.46-47 (in Ukrainian)

[11] V.B. Bigdan, O.O. Krykovliuk, Yu.M.Tchornyy, “Applying of island model of genetic algorithm for optimization in simulation experiments,” in *Decision Theory:* VII The International school-seminar, Uzhogorod, Ukraine, 2016, pp.44-45 (in Ukrainian)

[12] O.O. Krykovliuk “Applying of island model of genetic algorithm for scheme realization of parallelizing processes of directed search on cluster architectures,” in *Problems of decision making under uncertainties,* PDMU: XXIII International conference, Mukachevo, Ukraine, 2014, pp. 120 (in Ukrainian)

[13] V.B.Bigdan “Models of data flow control by flow of data in distributed search of optimal solutions,” *The theory of optimal solutions*, 2009, No 8, pp. 105-113 (in Rssian)

[14] Vl Pepelyaev, Yu.M.Czornyy, “The principlesof fitness functions bulding for global optimizationalgorithms testing,” *Computer mathematics*, Kyiv, Ukraine, 2017, No 2, pp. 62 – 71 (in Ukrainian) pp.62-71 (in Rssian)