Journal of Engineering and Applied Sciences 11 (6): 1243-1247, 2016
ISSN: 1816-949X

The Mathematical Model of Building a Multi-Level Topology of Computer Network for Distributed Corporate System Based on the Inverse Problem

S.P. Vorobyov
M.I. Platov South-Russian State Technical University (NPI),
Prosveshcheniya Str. 132, Novocherkassk, Russian Federation

Abstract: This study describes the formulation of a mathematical model for optimization of multi-level topological structure of a distributed enterprise information system computer network. To solve this problem, it is proposed to use an approach based on a modified genetic algorithm. The results of numerical experiments are presented in the study.

Key words: Mathematical model, computer network, genetic algorithm, multi-level topology, Russia

INTRODUCTION

Within the framework of systems theory, the inverse problem or the synthesis problem is associated with definition of the structure, parameters and properties of the system within a given range of possible conditions. Upon that, it is assumed that firstly system quality evaluation criteria are to be selected, then a system 0 model is built, corrected and studied; then values of absolute indices of its properties are determined, simulation results are analyzed and a decision about the quality of the synthesized system is made. Thus, the system will be obtained as a result of synthesis that corresponds to the selected quality criteria.

Modern enterprise information system is built on the basis of a distributed complex of network software and hardware means which include a plurality of interconnected and interacting components solving various functional tasks within the framework of a consistent approach to achieve a common goal both in terms of high-speed and high-quality data transmission and meeting to a set of system-wide and functional requirements which specifically define performance, bandwidth, reliability, etc. figures. Accounting all the characteristics upon the designing of such systems is reduced to a rather complex multi-criteria optimization problem that has no generally effective solution algorithm. Most of the proposals involve a process of creation of a distributed information system which represents a multi-step iterative procedure generally comprising separate design for technical, software, information, administrative and organizational aspects and not taking into account the need for unity of technical and political decisions. In addition, afterwards there are problems of distribution of computing, information, functional and other resources that are already solved on the basis of the proposed system architecture and consider options for improving the structure in order to improve performance, reliability, etc. in a fairly rigid existing structural constraints that do not always provide the best solution. A promising complex tool for building architectural solutions and distributed enterprise information systems is an approach based on multilevel model of representation of overlapping distributed network infrastructure (Vorobyov, 2009).

Multi-Level topological structure is a representation of a network structure with the full description of the network connections between objects at different levels of the model that links into a unified whole the aspects of data communication control with support of QoS protocols, technical and programmatic implementation, information space and management.

A multi-Level architectural decision in accordance with the Cisco concept is currently used in the capacity of a basic version for implementation of an enterprise information system infrastructure; it includes access, distribution and core levels, the local server and server farm, as well as a storage system or storage area network. The result is a flexible, scalable, reliable and stable system which consists of typed blocks and provides the required functionality. Access level provides connectivity user workstations to the network. The distribution level provides connectivity through the use of access policies and defines the boundaries of regions and segments. The core level provides optimal transport between distributed data network sites. The server farm provides shared storage of corporate information resources within the SAN and query processing.

MATERIALS AND MATHODS

Generalized problem statement: In this case the pair ($O^{(l)}$, $T^{(l)}$ is introduced to describe the distributed enterprise information system S in terms of a multi-level approach; it comprises a set $O^{(l)}$ which contains all the necessary objects S and $T^{(l)}$ is the family of subsets of the set O that define a topological structure of system S at l level l = {al, dl, cl, sl}, respectively, for the levels of access, distribution, core and server farm SAN. In addition, there are also introduced: a set of functional subsystems FSS = {FSSi, 1 = $\overline{1,m}$ }; a set of information resources of a functional subsystem FSSiIMi ={Imij, j =1, $\overline{1,kir}$ }, $i=\overline{1,m}$; multiple servers of the enterprise information system SR = {SRj, j = $\overline{1,ks}$ }. Function FTR ($O^{(l)}$,$T^{(l)}$) defines the volume of transit traffic at the l level.

It is necessary to define such a set $T^{(al)}_{i_{opt}}$ on the level of access that will include the location of servers and services, functional subsystems and their informational resources optimally in terms of the selected criteria:

$$g^{(al)}\left(T^{(al)}_{iopt},SR^{(al)},FSSL^{(al)},IM^{(al)}\right)$$
$$\rightarrow i\min\max T^{(al)}_{iopt}=\left\{T^{(al)}\right\}$$

Subject to the constraints $h^{(al)}_j\left(T^{(al)}_{iopt},SR^{(al)},FSS^{(al)},IM^{(al)}\right) \le H^{(al)}_j, j=\overline{1,m^{(al)}}$ where m is a number of constraints imposed on the topological structure of the computer network at the access level of the selected interaction model.

In the same way it is required to determine at the distribution level such a subset $T^{(dl)}_{iopt}$ of the family T^{dl} which will describe entering the elements of active network equipment $X^{(dl)}$ in the distributed structure based on switches as well as the necessary servers and services and information resources of functional subsystems at:

$$g^{(dl)}\left(T^{(dl)}_{i_{opt}},SR^{(dl)},FSS^{(dl)},IM^{(dl)}\right)\underset{i}{\rightarrow}\min\max$$

Upon compliance with functional constraints and also constraints based on networking standard:

$$h_j^{(dl)}\left(T^{(dl)}_{i_{opt}},SR^{(dl)},FSS^{(dl)},IM^{(dl)}\right)\le H_j^{(dl)}, j=\overline{1,m^{(dl)}}$$

At the level of the core, set $X^{(cl)}$ includes elements of active network equipment, as well as the necessary servers and services and information resources of functional subsystems and it is required to determine a subset $T^{(cl)}_{iopt}$ at the optimal criterion value:

$$g^{(cl)}\left(T^{(cl)}_{i_{opt}},SR^{(cl)},FSS^{(cl)},IM^{(cl)}\right)\rightarrow\min\max$$

and compliance with the relevant functional constraints, including quality of service:

$$h^{(cl)}_j\left(T^{(cl)}_{i_{opt}},SR^{(cl)},FSS^{(cl)},IM^{(cl)}\right).\le H^{(cl)}_j, j=\overline{1,m^{(cl)}}$$

At the server farm level, set $O^{(sl)}$ includes elements that specify the necessary active network equipment, servers, storage systems, etc. During building the topology $T^{(sl)}_{iopt}$ it is required to determine the optimal distribution of elements of the set on the elements of the network infrastructure at:

$$g^{(sl)}\left(T^{(sl)}_{i_{opt}},SR^{(sl)},FSS^{(sl)},IM^{(sl)}\right)\rightarrow\min\max$$

taking into account existing constraints from the software and systems:

$$h^{(sl)}_j\left(T^{(sl)}_{i_{opt}},SR^{(sl)},FSS^{(sl)},IM^{(sl)}\right)\le H^{(sl)}_j, j=\overline{1,m^{(sl)}}$$

Mathematical model of optimization: Detailed mathematical model includes a set of workstations WS = {WS_i, $\overline{1,n_{ws}}$ =}; a set of active network equipment elements (switches, routers, etc.) SW = {SW_i, i =} and a set of storage systems SS = {SSi, i=}. As a result, a lot of O = WSUSRUSWSS

To build the model, the following variables are introduced:

x^{al}={1, if the set element o belongs to the accept level T^{al}; o-otherwise

x^{dl} = {1, if the set element o belongs to the distribution level T^{dl}; o-others

x^{cl} = {1, if the set element o belongs to the core level T^{cl}; o-otherwise

x^{sl} = {1, if the set element o belongs to the server farm level T^{sl}; o-otherwise

In order to implement a distributed information system structure the following conditions and constraints are introduced: Each element belongs to only one level:

$$\sum_{o_j}\left(x^{al}+x^{dl}+x^{cl}+x^{sl}\right)=1$$

Connecting workstations to access level switches

$$\sum_{i=1}^{nws}\sum_{j=1}^{nsw} xws_i sw_j x^{al}_{sw_j}=1$$

where, $x_{ws}i_{sw}j$ = {1, if the workstation Ws_i is connected to a switch Sw_j. A workstation may be connected to only one switch Sw_j:

$$\sum_{j=1}^{n_{SW}} xws_i sw_j x^{al}_{sw_j} = 1, i = \overline{1, n_{ws}}$$

Connection of an access level active network equipment to the level of distribution taking into account the availability of back-up connections and trunk channels:

$$\sum_{i=j}^{n_{sw}} \sum_{j=1}^{n_{sw}} x_{sw_i sw_j} x^{al}_{swi} x^{dl}_{sw j} = res$$

where res-coefficient of redundancy of communication channels between switches of access and distribution levels. Servers can be connected to an active network equipment of the distribution level:

$$\sum_{J=1}^{n_{SW}} Xsr_i sw_j x^{dl}_{sw j} = 1, i = \overline{1, ks}$$

The connection between the active equipment at the distribution level, taking into account the availability of back-up components:

$$\sum_{i=1}^{n_{sw}} \sum_{j=1}^{n_{sw}} x_{SW_i SW_j} x^{dl}_{sw j} = res^{dl}$$

Connection of the active distribution network $T^{(cl)}_{iopt}$ equipment level to the level of the core based on the availability of backup connections and trunk channels:

$$\sum_{i=1}^{n_{sw}} \sum_{j=1}^{n_{sw}} x_{sw_i sw_j} x^{dl}_{swj} x^{cl}_{sw j} = res$$

where, resc coefficient of redundancy of communication channels between switches of distribution and core levels. Servers can be connected to an active network equipment of the core level:

$$\sum_{j=1}^{n_{sw}} x_{sr_i sw_j} x^{cl}_{swj} = 1, i = \overline{1, ks}$$

The connection between the active equipment at the core level taking into account the availability of back-up connections:

$$\sum_{i=j}^{n_{sw}} \sum_{j=1}^{n_{sw}} x_{sw_i sw_j} x^{cl}_{swi} x^{cl}_{swj} = res^{cl},$$

Connection of the active network equipment of a server farm level to the core level taking into account the availability of backup connections and trunk channels:

$$\sum_{i=1}^{n_{sw}} \sum_{j=1}^{n_{sw}} x_{sw_i sw_j} x^{cl}_{swi} x^{cl}_{sw_j} = ress$$

where, ress coefficient of redundancy of communication channels between switches of server farm and core levels. Servers can be connected to an active network equipment at the server farm level:

$$\sum_{j=1}^{n_{sw}} x_{sr_i sw_j} x^{sl}_{sw_j} = 1, i = \overline{1, ks}$$

Storage systems can be connected to an active network equipment of the server farm level:

$$\sum_{j=1}^{n_{sw}} x_{ss_i sw_j} x^{sl}_{sw j} = 1, i = \overline{1, n_{ss}}$$

The connection between the active equipment at the server farm level taking into account the availability of backup connections:

$$\sum_{i=1}^{n_{sw}} \sum_{j=1}^{n_{sw}} x_{sw_i sw_j} x^{sl}_{sw_i} x^{sl}_{sw j} = res^{sl},$$

The effectiveness of implemented topological structure is determined by the following Equation:

$$FT\begin{pmatrix} x^{al}, x^{dl}, x^{cl}, x^{sl} x_{, ws_i sw_j,} x_{sr_i sw_j,} x_{ss_i sw_j,} \\ x_{sw_i sw_j,} x^{al}_{sw_i,} x^{dl}_{sw_j,} x^{sl}_{sw_i} x^{cl}_{sw_j} \end{pmatrix} \to \min\max$$

RESULTS AND DISCUSSION

The NP-complete problems are included in the process of designing a computer network of a distributed information system can be solved by exact methods: branch and bound algorithm, branch-and-prune algorithms and dynamic programming. But with the increase in the tasks dimension their computational complexity grows exponentially. Therefore, Genetic Algorithms (GAs) which belong to the class of evolutionary algorithms are most successfully applied to solve such problems.

The basic idea of a genetic algorithm is to create a population of individuals each of which is represented as a chromosome. Any chromosome is a possible solution to the considered optimization problem. Only an objective function value (or fitness function) is used to find the best solutions. The value of the fitness function for an individual indicates how an individual described by this chromosome is well-suited to solve the problem. A chromosome consists of a finite number of genes presenting the genotype of the object, i.e., the sum of its hereditary characteristics. The process of evolutionary search is conducted only on the genotype level. Basic biological operators (crossing, mutation, inversion, etc.) are applied to a population. A population is constantly updated with the help of generation of new species and destruction of the old ones and each new population becomes better and depends only on the previous one.

Application of genetic algorithm requires development of a method of solution encoding. In this problem, a solution is understood as the permissible network topology being a connected graph. When coding a solution of the problem in a bit string

we will use the incidence matrix which shows the connection of working stations WS, servers SR and storage systems SS with switches SW and connection of switches among themselves. The network topology is encoded with a bit string of size $n_{ws}(n_{ws}+n_{sr}+n_{ss}+n_{sw})$ where a non-zero value in the corresponding bit means availability of a communication channel between the network nodes, 0 absence of a communication channel. Adaption of genetic algorithm to solve a particular optimization problem involves a feasible solution coding (in terms of acceptable values) that is defined by boundary conditions of the problem and its decryption, i.e., representation of the solution obtained by the genetic algorithm to the easy to read view.

Fitness function of genetic algorithm is the optimality criterion of a topological structure. After stopping the algorithm, the fitness function value of the best chromosome is used in the capacity of a fitness function assessment value for the whole task. The criterion for stopping the algorithm may be a convergence of the population (the difference between the values of the function for several generations is below a certain threshold), the achievement of a certain number of generations, the running time, etc. It is important to note that for some classes of problems the same genetic algorithm can provide the optimal solution while for others it turns out to be ineffective or totally unacceptable. For this reason, in recent years in a number of studies aimed at improving the efficiency of the genetic algorithm, the ideas on creating an adaptive genetic algorithm have taken a significance. They are mainly associated with parallelization of a genetic algorithm, structuring of a population, migration of chromosomes and populations, etc (Back *et al*., 1997).

Modifications of the genetic algorithm which can change its parameters during operation are of separate interest. They were a continuation of development of the idea on generational algorithms which change the size of a population in the process of their work. Such adaptive algorithms are able to change not only the size of the population, but also the essence of genetic operators (crossover probability, mutation probability). As a rule, these changes occur by choosing the parameter of several options identified before the start of the algorithm. The idea of adaptive genetic algorithm gets its expression in the concept nGA representing a multi-level genetic algorithm (Goldberg, 1989). The lower level of this algorithm directly performs the task of improving the population of solutions. The upper levels are genetic algorithms solving the optimization problem to improve the low-level algorithm parameters. At the same time speed of the low-level algorithm and its speed of improvement of the population from generation to generation is commonly used in the capacity of the objective function.

To increase the speed of convergence of the genetic algorithm and improve its operation resistance it is modified taking into account the peculiarities of the task to which it is applied. The study (Lobanov, 2008) proposes three new modification of the genetic algorithm: the use of automats with flags, the algorithm which restores relations between states and states sorting algorithm in order of their use.

Based on the researches (Goldberg, 1966; Kureichik, 1988, 1998, 2002) the problem of building an optimal topology stated in this study is achieved by using a modified genetic algorithm.

Modification of a standard genetic algorithm is using an initial structuring of a population on the basis of the objective function, using the N-point crossover operator model (number of intersection points is selected) and universal stochastic selection, activation of the strategy of rapid partitioning the search space into areas with high values of the utility function, introduction of an adaptive filter clipping solutions with low value of the utility function.

The need for the initial structuring of the population is due to the fact that in view of the potentially large dimension of the problem to be solved in this work, the classic genetic algorithm converges quite rare (<10% of cases). In experimental studies, it was found that the main cause of this problem is an incorrect initial population obtained on the basis of a random number generator work excluding any special features of the data domain. Therefore, the algorithm was implemented on generation of the most correct initial population in terms of meeting critical constraints of the problem.

The use of a N-point crossover operator in the proposed algorithm is caused by the fact that in the large their more efficient solution (in terms of speed) via genetic

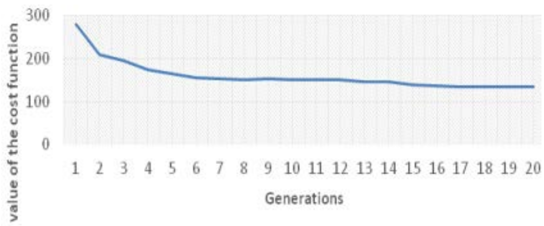

Fig. 1: Dependency diagrams of the optimized function values on the number of algorithm generations

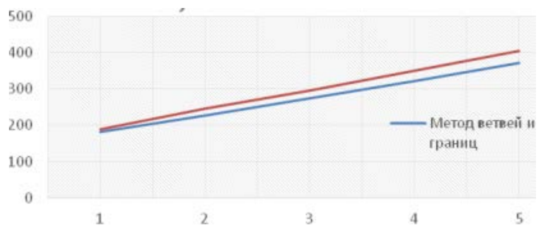

Fig. 2: Dependencies of the objective function value on sets with a different number of switches used

algorithm one or two intersection points of chromosomes are insufficient. Therefore, the number N of these points is based on a stochastic choice for every generation depending on the ratio of the objective function average value among all individuals in a population to the function value for its best individual. The closer these values on the real axis, the greater must be N. This makes it possible to obtain new alleles which may su dimension problems a chromosome length is large enough and for bsequently improve the quality of future generations. Versatility of the described stochastic selection is shown in the fact that it does not depend on the population size and the type of individuals.

Using a strategy of the rapid partitioning of the search space into regions with high values of the utility function is due to the potentially large dimension of the problem. A quick sort algorithm is used to sort the individuals (Algorithms, 2005; Knuth, 2012).

Figure 1 shows dependency diagrams of the optimized function values on the number of algorithm generations for fixed values of other parameters derived on the basis of experimental results using simulation and optimization program.

Figure 2 shows dependency diagrams of the objective function on the sets with different number of switches used for the genetic algorithm and the branch and bound method. Analysis shows that when solving the problem using a modified genetic algorithm deviation of the objective function value from the optimum is within 3-7%. Branch and bound method.

CONCLUSION

The main conclusion obtained as a result of experiments on test cases is that the developed genetic algorithm allows finding a valid or optimal (from the point of view of constraints in the problem) solution thus, being a kind of search techniques with elements of chance, for significantly less time (on the order of magnitude) than by the branch and bound method.

ACKNOWLEDGEMENTS

The results were obtained with the support of the project No. 1.2690.2014/K “Methods for solving inverse problems of complex systems diagnostics (in engineering and medicine) based on natural-model experiment” implemented within the framework of a project part of the state task.

REFERENCES

Back, T., D.B. Fogel and Z. Michalewuz, 1997. Handbook of Evolutionary Computation. Oxford University Press, New York, Pages: 1130.

Goldberg, D.E., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Addison-Wesley Publishing Co. Inc., USA., Pages: 412.

Knuth, D.E., 2012. Art of Computer Programming. M.Williams, USA., Pages: 824.

Kureichik, V.M., 1988. Genetic Algorithms. TRTU Publishing House, Monograph, Taganrog, Pages: 242.

Kureichik, V.M., 1998. Methods for Genetic Research. TRTU Publishing House, Taganrog, Pages: 118.

Kureichik, V.M., 2002. Genetic Algorithms and their Application. 2nd Edn., TRTU Publishing House, Taganrog, Pages: 242.

Vorobyov, S.P., 2009. Possible directions of use for the concept of multi-level topology and optimization of distributed corporate systems. Questions Mod. Sci. Pract., 8: 131-143.