Computer Science Journal of Moldova, vol.**10**, no.2(29), 2002

Configuration of local area network set of servers

Ion Bolun, Anatol Ciumac

Abstract

Multiserver LANs, that offer many categories of services with absolute priority discipline, are investigated. The flows of user requests to servers are considered Poissonian. The optimization problem for LAN server set configuration is formulated; the goal is to minimize costs with servers, without exceeding the established mean response time for all users' requests of different categories. Analytical solutions for particular cases are obtained and an algorithm for solving the general problem is proposed.

Key words: server set, configuration, optimization, response time, algorithm, user requests, request flow.

1 Introduction

Local area networks are the most efficient way of using informatics means in companies. Primordial aspects in designing and using LANs are the minimization of costs and the insurance of required QoS [1, 3-5, 7]. From this point of view it is important to determine the set of servers needed to process the user requests and to distribute request flows among the net's servers. When determining the server set, it is usually considered the total request flows' laboriousity, and the total servers' capacity must exceed this laboriousity by a specified value [3, 7]. Such an approach is acceptable when inoperative requests predominate in the net [2]. But, in the modern data networks, the implementation of services that involve an intensive processing of graphics, video, video on demand etc. is more and more solicited. In such conditions, the need for operative request processing is high, and operative request flows predominate, as a rule, in the net. Meeting such requests at reasonable costs involves the utilization of disciplines with priorities in serving requests of different categories. In the known models, used for configuring the server set that processes requests on a priority basis, the requests are usually considered as having the same laboriousity. But in fact, requests' characteristics differ in most of cases.

In this paper, aspects referring to the configuration of local area networks server set are investigated. The goal is to minimize costs with servers' set configuration, without exceeding the established mean response time for requests of different categories. The given optimization problem is formulated; analytical solutions for particular cases are obtained and an algorithm for solving the general problem is proposed.

2 The server set functioning model

Let a broadcast LAN consist of N user workstations. User requests need $i = \overline{1,n}$ categories of services from servers. Servers of m types may be used in LAN. Each server could offer one or more of n categories of services. Services are offered on a priority basis. An absolute discipline with continued service is used; the smaller the value of index i – the higher the priority of the respective request category.

Costs and required QoS are major characteristics in computer systems, inclusively in computer networks, investigations [3, 4, 6]. As to QoS, the main factors used are, primarily, the time of response to user requests and, in a few cases depending on the case, the viability, the security etc. There are some specific requirements too, such as, assuring an isochronal data traffic, for example [7].

In this paper, costs and mean response time to user requests are considered as basic characteristics. Frequently, the solutions referred to other characteristics are invariants, compared to the enumerated ones, and could be solved separately or need minor modifications of the solution referred to the obtained characteristics. Used notices:

u_i – the mean processing laboriousity of a user request to the $i = \overline{1,n}$ service (category i request). Here n is a total number of user request categories in the network;

T_i, T_{di} – the real and, respectively, the established top limit of the mean response time to a request of i category;

γ_{li} – the flow rate of category i requests generated by the net station l;

Λ_i – the total flow rate of category i requests in the network;

M_{ej}, M_j, C_j, U_j – the number of existent servers, the total number of needed servers, the cost and, respectively, the productivity (capacity) of type $j = \overline{1,m}$ server;

λ_{ji} – the total flow rate of category i requests served by server of type j.

The following suppositions are accepted:

1) the net data transfer medium is low charged with traffic. The most used LANs – Ethernet, operates, par example, in such conditions. Thus, it is considered that the data transfer time is much shorter, than the data processing time in the net;

2) the flows of user requests, generated in the net, are stationary Poissonian, which really takes place, usually, in networks [1, 4, 5, 8];

3) the distribution of time that a server need to process the requests of different categories is general. But in some cases the exponential service distribution is also investigated; this distribution frequently is peculiar for real computer systems [1, 4, 5].

The mean time τ_{ji} (rate μ_{ji}) needed for a server of category j to process a request of category i is determined as

$$\tau_{ji} = \frac{1}{\mu_{ji}} = \frac{u_i}{U_j}, \quad j = \overline{1,m}, \quad i = \overline{1,n}, \tag{1}$$

and the total flow rate Λ_i of category i requests in the net is

$$\Lambda_i = \sum_{l=1}^{N} \gamma_{li}, \quad i = \overline{1,n.} \tag{2}$$

Also, at a general service distribution and the absolute service discipline with continued service, the following relations take place [5]:

$$T_{ji} = \frac{\tau_{ji}}{1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}} + \frac{\sum\limits_{k=1}^{i}\lambda_{jk}\tau_{jk}^{(2)}}{2\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)\left(1-\sum\limits_{k=1}^{i}\lambda_{jk}\tau_{jk}\right)},$$
$$j=\overline{1,m},\quad i=\overline{1,n}, \tag{3}$$

where T_{ji} is the mean response time and $\tau_{ji}^{(2)}$ is the 2nd moment of service time of an i category request to a j category server.

In the case of exponential service time of the requests of different categories, one has [5]:

$$\tau_{jk}^{(2)} = 2\tau_{jk}^2; \tag{4}$$

$$T_{j1} = \frac{1}{\mu_j - \lambda_j},\quad j=\overline{1,m}; \tag{5}$$

$$T_{ji} = \frac{\tau_{ji}}{1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}} + \frac{\sum\limits_{k=1}^{i}\lambda_{jk}\tau_{jk}^2}{\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)\left(1-\sum\limits_{k=1}^{i}\lambda_{jk}\tau_{jk}\right)},$$
$$j=\overline{1,m},\quad i=\overline{1,n}. \tag{6}$$

When formulating and solving diverse problems, there will be taken into consideration that the investigated system must operate in a stationary regime. For a j type server, the stationary regime exists if the server charge is lower then 1, i.e.

$$\sum_{i=1}^{n}\lambda_{ji}\tau_{ji} < 1,\quad j=\overline{1,m}. \tag{7}$$

Supplementary statements and conventions:

1) server types will be so defined that

$$U_j < U_{j+1}, \quad j = \overline{1, m-1}; \tag{8}$$

2) referring to the ratio “cost/productivity”, it may be three cases:

$$\begin{aligned} \frac{C_j}{U_j} &> \frac{C_{j+1}}{U_{j+1}}, \quad j = \overline{1, m-1}. & (9)\\ \frac{C_j}{U_j} &= \frac{C_{j+1}}{U_{j+1}} = a, \quad j = \overline{1, m-1}. & (10)\\ \frac{C_j}{U_j} &< \frac{C_{j+1}}{U_{j+1}}, \quad j = \overline{1, m-1}. & (11) \end{aligned}$$

Case (9) is a feebler form than the one defined by the Grosh law [4], proposed in the 60’. This law states that the cost of a computer operation is inversely proportional to square root of computer productivity; for modern technologies this dependence is feebler. Case (10) is used by L.Kleinrock and others, but in the case of data circuits [4].

The case (11) is more rarely used; the opportunity for resource concentration in this case is caused, as a rule, by the facilitation of the administration of the respective equipment resources and databases, as well as by the fact that at the same arrival flow rate λ the following relation takes place:

$$T\left(U_j, \lambda\right) > T\left(U_{j+1}, \lambda\right).$$

In case of systems for operative requests service, when it is important to assure a small response time, the use of a server with a higher productivity is reasonable even in conditions (11), if [2]

$$\frac{C_j}{\overset{\frown}{\lambda}_j} \geq \frac{C_{j+1}}{\overset{\frown}{\lambda}_{j+1}},$$

where $\overset{\frown}{\lambda}_j$, $\overset{\frown}{\lambda}_{j+1}$ are saturation (predominating [2]) flows for servers, respective, j and $j+1$.

Some problems for the server set M_j, $j = \overline{1,m}$ configuring and for the user request flows Λ_i, $i = \overline{1,n}$ distributing among servers are investigated below.

3 A mathematic formulation of the general problem

For the defined notices and accepted in p. 2 suppositions, the following values are known: Λ_i, T_{di}, $i = \overline{1,n}$; M_{ej}, τ_{ji}, $\tau_{ji}^{(2)}$, $j = \overline{1,m}$, $i = \overline{1,n}$. It is required to determine the server set M_j and the distribution λ_{jri}, $j = \overline{1,m}$, $r = \overline{1,M_j}$, $i = \overline{1,n}$ of request flows Λ_i, $i = \overline{1,n}$, that would assure the minimal total costs C with network servers

$$C = \sum_{j=1}^{m} C_j M_j \rightarrow \min \tag{12}$$

and satisfy the restrictions:

$$T_{jri} \leq T_{di}, \quad j = \overline{1,m}, \quad r = \overline{1,M_j}, \quad i = \overline{1,n}; \tag{13}$$

$$\sum_{j=1}^{m} \sum_{r=1}^{M_j} \lambda_{jri} = \Lambda_i, \quad i = \overline{1,n}; \tag{14}$$

$$M_j \geq M_{ej}, \quad j = \overline{1,m}. \tag{15}$$

In inequality (13) durations T_{jri}, $j = \overline{1,m}$, $r = \overline{1,M_j}$, $i = \overline{1,n}$ are determined according to formula (3) for general service distribution, and according to formula (6) – for exponential service distribution of user requests, replacing in these T_{ji} by T_{jri} and λ_{ji} by λ_{jri}. To mention that:

1) restriction (13) specifies the fulfillment of requirements in respect of how operatively are the user requests processed;

2) condition (14) provides that all user requests are processed by servers;

3) restriction (15) requires that the existent servers are loaded with traffic.

4 Preliminary considerations and particular aspects of the problem

The variables M_j, $j = \overline{1,m}$ are of integer type, and variables λ_{jri}, $j = \overline{1,m}$, $r = \overline{1,M_j}$, $i = \overline{1,n}$ are of continue type. Dependencies (3) and (6), used in restriction (13), are nonlinear. So, the problem (12)-(15) is one of nonlinear programming in integers. It's solving through the existent methods isn't guarantied. Using the particularities of the problem, an algorithm for its solving at a not very large number m of server types is proposed in p. 5.2. Solutions for some particular cases of the problem are obtained too.

In order to solve the general problem, first (p. 4) certain particularities are emphasized, basing on which the certain rules that allow to reduce the initial problem dimension are determined. The problem of user requests flow distribution among the servers of the same type is investigated in p. 4.1. Then, the expressions for calculating the server saturation flows are determined in p. 4.2, and an algorithm for determining the possibilities of a given server set to serve user request flows is described in p. 4.3.

The problem is solved in two stages. First, M_j, $j = \overline{1,m}$ is determined (p. 5), and afterwards the reasonable values of rates λ_j, $j = \overline{1,m}$ are concretized. A particular case, when a server set can be constituted only of servers of the same type M_j, is investigated in p. 5.2.

In many cases the results of sufficient adequacy are obtained when only one category of requests is considered ($n = 1$). This case is investigated in p. 5.3.1. The problem for a general distribution of requests‘ service time is formalized there. The problem takes a simpler form for an exponential distribution of user requests service time. In this case it takes the form of an integer linear programming one. Moreover, if m is large and values μ_j, $j = \overline{1,m}$ are relatively uniform distributed in the interval $[\mu_1; \mu_m]$, then the analytical solution of the problem is obtained.

An algorithm for solving the general problem is described in p. 5.3.2.

4.1 Request flows distribution among servers of the same type

Let requests flows Λ_{ji}, $i = \overline{1,n}$ be served by M_j servers of j type. It is required to determine the repartition λ_{jri}, $r = \overline{1,M_j}$, $i = \overline{1,n}$ of flows Λ_{ji}, $i = \overline{1,n}$ among the M_j servers of j type. As optimization criteria the following is used:

1) minimum of the mean response time T_{ji} to requests of each of the $i = \overline{1,n}$ categories, these being provided in the order of the request categories' priority. The case is investigated in p. 4.1.1;

2) maximization of the request flow rates that can be served. The case is investigated in p. 4.1.2.

4.1.1 Ensuring the minimum of mean response time to requests

Let request flows Λ_{ji}, $i = \overline{1,n}$ to be served by M_j servers of j type. It is required to determine the repartition λ_{jri}, $r = \overline{1,M_j}$, $i = \overline{1,n}$ of flows Λ_{ji}, $i = \overline{1,n}$ among the M_j servers of j type so, that there be assured the minimum of the mean response time $\tilde{T}_{ji}$ consecutively for each of the $i = \overline{1,n}$ request categories: for the category 1, then for the category 2 etc. Such a consecutiveness of the request flows repartition among servers is justified by the service priority of request categories with a smaller value of i index. Thus, consecutively for each $i = \overline{1,n}$, it is required

$$\tilde{T}_{ji} = \frac{1}{\Lambda_{ji}} \sum_{r=1}^{M_j} T_{jri}\lambda_{jri} \rightarrow \min, \tag{16}$$

for observing restriction

$$\sum_{j=1}^{M_j} \lambda_{jri} = \Lambda_{ji}. \tag{17}$$

Here r indicates server r from the total number of M_j servers. The response time T_{jri} to requests of i category, that are served by the r server of j type, is determining according to formula (3), by replacing in this T_{ji} by T_{jri} and λ_{ji} by λ_{jri}.

The solutions of the problem (16)-(18) for the cases $i = 1$ and $i = 2$, obtained using Lagranjian multipliers method, showed that the optimal distribution is the uniform one, that is $\lambda_{jr1} = \Lambda_{j1}/M_j$, $r = \overline{1, M_j}$ and, respectively, $\lambda_{jr2} = \Lambda_{j2}/M_j$, $r = \overline{1, M_j}$. It is necessary to demonstrate that the uniform distribution is optimal for the general case too. Let the uniform distribution be optimal for $k = \overline{1, i-1}$. Then, taking into consideration relation (3), T_{jri} is determined as follows:

$$\begin{gathered} T_{jri} = \\ = \frac{\tau_{ji}}{1 - \sum\limits_{k=1}^{i-1} \lambda_{jk}\tau_{jk}} + \frac{\lambda_{jri}\tau_{ji}^{(2)} + \sum\limits_{k=1}^{i-1} \lambda_{jk}\tau_{jk}^{(2)}}{2\left(1 - \sum\limits_{k=1}^{i-1} \lambda_{jk}\tau_{jk}\right)\left(1 - \lambda_{jri}\tau_{ji} - \sum\limits_{k=1}^{i-1} \lambda_{jk}\tau_{jk}\right)}, \\ r = \overline{1, M_j}. \end{gathered} \tag{18}$$

It is necessary to demonstrate that the uniform distribution is optimal for the case i too. The Lagranjian function L is

$$L = \tilde{T}_{ji} + \chi\left(\Lambda_{ji} - \sum_{r=1}^{M_j} \lambda_{jri}\right) \rightarrow \text{ min}, \tag{19}$$

where χ is the Lagranjian multiplier. So

$$\begin{cases} \frac{\partial L}{\partial \lambda_{jri}} = \frac{\partial \tilde{T}_{ji}}{\partial \lambda_{jri}} + \chi = \frac{1}{\Lambda_{ji}} \frac{\partial (T_{jri}\lambda_{jri})}{\partial \lambda_{jri}} + \chi = 0, r = \overline{1, M_j} \\ \frac{\partial L}{\partial \chi} = \Lambda_{ji} - \sum\limits_{r=1}^{M_j} \lambda_{jri} = 0. \end{cases} \tag{20}$$

Replacing T_{jri} in the equation of the first line of system (20) by expression (18), afterwards taking the derivation and solving the respective equation by λ_{rji}, one could obtain

$$\lambda_{jri}^{(1,2)} =$$

$$= \frac{1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}}{\tau_{ji}}\left(1 \pm \sqrt{\frac{\tau_{ji}\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}^{(2)}-\tau_{ji}^{(2)}\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)}{\tau_{ji}\left[2\tau_{ji}^{2}-\tau_{ji}^{(2)}+2\chi\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)\right]}}\right),$$

$$r = \overline{1, M_j}. \tag{21}$$

For the stationary functioning of the server system, it is necessary to satisfy condition (7), which in this case takes the form

$$\lambda_{jri} < \frac{1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}}{\tau_{ji}}, \tag{22}$$

That's why in expression (21) from "±" operations only the case of using the "minus" operation is reasonable. Replacing λ_{jri}, $r = \overline{1, M_{ej}}$ in system's (20) second line equation by expression (21), further to some simple transformations, one has

$$\sqrt{\frac{\tau_{ji}\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}^{(2)}-\tau_{ji}^{(2)}\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)}{\tau_{ji}\left[2\tau_{ji}^{2}-\tau_{ji}^{(2)}+2\chi\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)\right]}} =$$

$$= 1 - \frac{\Lambda_{ji}\tau_{ji}}{M_{ej}\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)}. \tag{23}$$

Replacing the expression contained in square root of (21) by that of the right part of the equation (23), one could obtain

$$\lambda_{jri} = \lambda_{ji} = \frac{\Lambda_{ji}}{M_j}, r = \overline{1, M_j}, \tag{24}$$

which was necessary to demonstrate. So, the optimal repartition of flows Λ_{ji}, $i = \overline{1,n}$ among the M_j servers of j type, assuring mean response time minimum $\tilde{T}_{ji}$ consecutively, in accordance to the service priority, for each of the $i = \overline{1,n}$ request categories, is the uniform one.

4.1.2 The maximization of the rate of request flows which can be served by a server set of the same type

The uniform repartition is the most reasonable also from the point of view of the maximization of the total flow Λ_{ji} of requests that could be served by a given set M_j of the same j type servers, when rates Λ_{jk}, $k = \overline{1, i-1}$ are known, for $i = \overline{1,n}$. The truth of this affirmation is demonstrated in this section. The given problem – dual comparatively to the one from p. 4.1, consists of the following. For each $i = \overline{1,n}$ consecutively, it is required

$$\Lambda_{ji} = \sum_{j=1}^{M_j} \lambda_{jri} \to \max \tag{25}$$

in respecting the restriction

$$\tilde{T}_{ji} \leq T_{di}, \tag{26}$$

where $\tilde{T}_{ji}$ is determined by T_{jri}, $r = \overline{1, M_j}$ according to (16).

Values λ_{jri}, $r = \overline{1, M_{ej}}$ are continuous, and $\tilde{T}_{ji}$ is a continuous raise function towards Λ_{ji}. That is why in relation (25) $\max \Lambda_{ji}$ will be assured for the case of equality in restriction (26). Taking into consideration these facts, problem (25)-(26) could be solved using the Lagranjian multipliers method. The Lagranjian function is

$$L = \sum_{r=1}^{M_{ej}} \lambda_{jri} + \chi \left(\frac{1}{\Lambda_{ji}} \sum_{r=1}^{M_{ej}} T_{jri} \lambda_{jri} - T_{di} \right) \to \text{ max.} \tag{27}$$

So

$$\begin{cases} \frac{\partial L}{\partial \lambda_{jri}} = 1 + \frac{\chi}{\Lambda_{ji}} \frac{\partial (T_{jri}\lambda_{jri})}{\partial \lambda_{jri}} = 0, r = \overline{1, M_{ej}} \\ \frac{\partial L}{\partial \chi} = \frac{1}{\Lambda_{ji}} \sum\limits_{r=1}^{M_{ej}} T_{jri}\lambda_{jri} - T_{di} = 0. \end{cases} \tag{28}$$

Taking into consideration (18), from the system (28) first line equations, as a result of some transformations, one could obtain

$$\lambda_{jri}^{(1,2)} = \frac{S}{\tau_{ji}} \left(1 \pm \sqrt{\frac{\chi\left[\tau_{ji}^2 (1-2S) - \tau_{jk}^{(2)}\right] - \tau_{ji} S^2 (2S\Lambda_{ji} - H)}{\chi\left(2\tau_{ji}^2 - \tau_{ji}^{(2)}\right) + 2S\tau_{ji}\Lambda_{ji}}} \right), \quad r = \overline{1, M_{ej}}, \tag{29}$$

where

$$S = 1 - \sum_{k=1}^{i-1} \lambda_{jk}\tau_{jk}; \tag{30}$$

$$H = \sum_{k=1}^{i-1} \lambda_{jk}\tau_{jk}^{(2)}. \tag{31}$$

By the same reasons as in the case of the expression (21), for $\lambda_{jri}^{(1,2)}$, $r = \overline{1, M_j}$ in expression (29) from the "±" operations only the case of using the "minus" operation is reasonable. By replacing λ_{jri}, $r = \overline{1, M_{ej}}$ in the system (28) second line equation by the expression (29), one could obtain

$$\sqrt{\frac{\chi\left[\tau_{ji}^2 (1-2S) - \tau_{jk}^{(2)}\right] - \tau_{ji} S^2 (2S\Lambda_{ji} - H)}{\chi\left(2\tau_{ji}^2 - \tau_{ji}^{(2)}\right) + 2S\tau_{ji}\Lambda_{ji}}} = 1 - \frac{\tau_{ji}}{SM_{ej}}. \tag{32}$$

Finally, by replacing the expression included in square root of (29) by that of the right part of the equality (32), relation (24) is obtained, which was required to demonstrate. Thus, the optimal repartition of flows Λ_{ji}, $i = \overline{1,n}$ among M_{ej} servers of j type, that assure the maximum flow rate Λ_{ji}, for each of the request categories $i = \overline{1,n}$, is the uniform one.

Consequence 1 (based on results obtained in pp. 4.1.1, 4.1.2). A necessary condition for the optimal request flows distribution among servers is the uniform distribution of the flows Λ_{ji}, $i = \overline{1,n}$ among the M_j servers of the same j type for $j = \overline{1,m}$. So, the general problem (12)-(15) is reduced to determining the flow rates λ_{ji}, $j = \overline{1,m}$, $i = \overline{1,n}$, rather than the flow rates λ_{jri} ($j = \overline{1,m}$, $r = \overline{1,M_j}$, $i = \overline{1,n}$).

4.2 Server saturation flows

First there will be given the definition of the canonic saturation (predominating [2]) flows. **The flows (flow rates)** $\bar{\lambda}_{ji}$, $i = \overline{1,n}$ are **canonic saturation flows** for the j server, if they satisfy restrictions (13), when the flows

$$\lambda_{ji} = \bar{\lambda}_{ji} + \Delta\lambda_{ji}, i = \overline{1,n},$$

don't satisfy these restrictions any more, no matter how small are values $\Delta\lambda_{ji} > 0$, $i = \overline{1,n}$. That is, flows $\bar{\lambda}_{ji}$, $i = \overline{1,n}$ are predominating in the sense of [2] – flows of the highest rate, for which the restrictions regarding the mean response time to requests of different categories, taking into consideration the application of these conditions first for the requests of category 1, then for that of category 2 etc. (in concordance with their service priority). Evidently, in this case inequalities (13) are transformed in equalities.

A separate flow could be also of saturation one. A flow $\overset{\frown}{\lambda}_{ji}$, at known rates of flows λ_{jk}, $k = \overline{1,i-1}$ that satisfy restriction (13), is named a **saturation flow** for the j server, if this satisfies the respective restriction from (13), but the flow

$$\lambda_{ji} = \overset{\frown}{\lambda}_{ji} + \Delta\lambda_{ji}$$

does not satisfy this restriction any longer, no matter how small is the value $\Delta\lambda_{ji} > 0$. Evidently, for the saturation flows the respective inequalities from (13) become equalities.

To mention that if all flows λ_{ji}, $i = \overline{1,n}$ are of saturation, then they are canonic of saturation, that is

$$\overset{\frown}{\lambda}_{ji} = \bar{\lambda}_{ji}, \qquad i = \overline{1,n}. \tag{33}$$

As a rule, for a specific system with a multidimensional request flow and high requirements to the serve requests operativity, only one flow is of saturation, or maximum two, of which one is of low operativity. These are the saturation flows that impose system server performance requirements. Therefore, in analyzing and configuring computer systems it is important to determine the saturation flows. Because inequality i from (13) for the saturation flow $\overset{\frown}{\lambda}_{ji}$ is transforming in equality and, taking into consideration (3), the following takes place

$$T_{di} = \\ = \frac{\tau_{ji}}{1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}} + \frac{\overset{\frown}{\lambda}_{ji}\tau_{ji}^{(2)} + \sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}^{(2)}}{2\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)\left(1-\overset{\frown}{\lambda}_{ji}\tau_{ji}-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)}, \\ j = \overline{1,m}, i = \overline{1,n}, \tag{34}$$

from where

$$\overset{\frown}{\lambda}_{ji} = \frac{2\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)\left[T_{di}\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)-\tau_{ji}\right]-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}^{(2)}}{\tau_{ji}^{(2)}+2\tau_{ji}\left[T_{di}\left(1-\sum\limits_{k=1}^{i-1}\lambda_{jk}\tau_{jk}\right)-\tau_{ji}\right]}, \\ j = \overline{1,m}, i = \overline{1,n}. \tag{35}$$

Evidently, formula (35) can be used also in determining the canonic saturation flows $\bar{\lambda}_{ji}$, $i = \overline{1,n}$, if in this equation to replace $\overset{\frown}{\lambda}_{ji}$, $i = \overline{1,n}$ and λ_{ji}, $i = \overline{1,n}$ by $\bar{\lambda}_{ji}$, $i = \overline{1,n}$.

4.3 Determining the possibilities of a given server set

Frequently it is necessary to determine if a known server set M_j, $j = \overline{1,m}$ (par example, an existent server set M_{ej}, $j = \overline{1,m}$) is able to serve a given request flows Λ_i, $i = \overline{1,n}$, assuring the mean response time T_{di} established for each request category $i = \overline{1,n}$. In the case when not all requests of flows Λ_i, $i = \overline{1,n}$ can be served, priority is given to the flows of category with a smaller i index, and the flows

$$\Delta\Lambda_i = \Lambda_i - \overset{\frown}{\Lambda}_i, \qquad i = \overline{1,n} \tag{36}$$

are not served; additional servers are needed to serve these flows. Here $\overset{\frown}{\Lambda}_i$, $i = \overline{1,n}$ are request flows that can be served by servers M_j, $j = \overline{1,m}$.

If service priority is offered to requests with smaller i index, then the optimal distribution λ_{ji}, $j = \overline{1,m}$, $i = \overline{1,n}$ (in the sense of maximizing the flow rates of requests of highest priorities that could be served) of request flows Λ_i, $i = \overline{1,n}$ among servers M_j, $j = \overline{1,m}$ could be realized conform to **algorithm A1**:

1^{o}. $i = 1$.

2^{o}. There is determined, $\overset{\frown}{\lambda}_{ji}$, $j = \overline{1,m}$ according to (35) and, also,

$$\overset{\frown}{\Lambda}_i = \sum_{j=1}^{m} M_j \overset{\frown}{\lambda}_{ji}, \quad \Delta\Lambda_i = \Lambda_i - \overset{\frown}{\Lambda}_i \ .$$

3^{o}. If $\Delta\Lambda_i \geq 0$, then $\lambda_{ji} = \overset{\frown}{\lambda}_{ji}$, $j = \overline{1,m}$. Otherwise $\Delta\Lambda_i = 0$ and, taking into consideration relation (24),

$$\lambda_{ji} = \frac{\Lambda_i}{M_{ej}}, \qquad j = \overline{1,m}.$$

4^{o}. If $i < n$, then $i := i + 1$ and switch to p. 2^{o}.

5^{o}. The searched values of flows λ_{ji}, $\Delta\Lambda_i$, $j = \overline{1,m}$, $i = \overline{1,n}$ are obtained. The flows $\Delta\Lambda_i$, $i = \overline{1,n}$ cannot be served by servers M_j if need to satisfy the restriction (13). If

$$\Delta\Lambda = \sum_{i=1}^{n} \Delta\Lambda_i > 0,$$

then additional servers to serve the flows $\Delta\Lambda_i$, $i = \overline{1,n}$ are needed. Otherwise ($\Delta\Lambda = 0$), the servers M_j, $j = \overline{1,m}$ assure the service of all requests respecting the requirements (13) and supplementary servers are not needed.

5 Determining the server set

5.1 The general problem reduced form

The results obtained in p. 4 allow reducing the dimension and, respectively, the complexity of the problem formulated in p. 3. Based on consequence 1 (p. 4.1.2), restrictions (14) can be replaced by restrictions

$$\sum_{j=1}^{m} M_j \lambda_{ji} = \Lambda_i, \qquad i = \overline{1,n}. \tag{37}$$

Moreover, when determining the server set for serving user requests, restrictions (13) and (37) can be replaced (based on results obtained in p. 4.2) by restrictions

$$\sum_{j=1}^{m} M_j \overset{\frown}{\lambda}_{ji} \geq \Lambda_i, \qquad i = \overline{1,n}. \tag{38}$$

Also, based on results obtained in p. 4.3, restriction (15) can be eliminated too, if algorithm 1 from p. 4.3 is firstly used for determining the request flows that need to be served by additional servers and later on, only the problem with these flows is examined. Evidently, at the second stage, which relates to flows redistribution with a view to reducing the response time to requests, all servers (existent and additional ones) and all request flows will be considered.

So, the problem examined in this section consists in determining the server set M_j, $j = \overline{1,m}$, needed to serve the users' request flows

Λ_i, $i = \overline{1,n}$, in assuring the minimum of the goal function (12) and observing restrictions (38). The saturation flows $\overset{\frown}{\lambda}_{ji}$, $j = \overline{1,m}$, $i = \overline{1,n}$ from (38) are determined according to (35), but at an exponential service distribution of user requests – relation (4) is considered. One could observe from (35) that saturation flows $\overset{\frown}{\lambda}_{ji}$, $j = \overline{1,m}$ depend on flows λ_{jk}, $j = \overline{1,m}$, $k = \overline{1,i-1}$, which, in their turn, depend on values M_j, $j = \overline{1,m}$ and Λ_i, $i = \overline{1,n}$.

To mention, that problem (12), (38) contains $\left(m + \sum\limits_{j=1}^{m} M_j\right) \Big/ 2m$ times less variables and $(1+m\sum\limits_{j=1}^{m} M_j+m/n)$ times less restrictions than problem (12)-(15). But, although the problem's dimension is considerably reduced, it is still a problem of integer nonlinear programming.

As a rule, number m of server types used in LANs is small: 1-5. Sometimes it is required to use only a single type of servers; this case is investigated in p.5.2. The general case is investigated in p.5.3. In p.5.3.1 there is also investigated an hypothetical case, that assume the possibility to select servers from a very large number of server types ($m \to \infty$). This case could serve for appraising the tendencies with respect to solving real problems, including the influence of m value on the problem-solving path.

5.2 Server set must be constituted of servers of the same type only

Let there be known values: T_{di}, Λ_i, μ_{ji}, $i = \overline{1,n}$; $j = \overline{1,m}$. It is required to determine the type j and the number M_j of servers of this type for serving the request flows Λ_i, $i = \overline{1,n}$, which would assure the minimum total cost of servers in observing restrictions (38) when using servers of j type only; that is:

$$C = C_j M_j \to \min_{j=\overline{1,m}} \tag{39}$$

in observing restrictions

$$M_j \overset{\frown}{\lambda}_{ji} \geq \Lambda_i, \qquad i = \overline{1,n}. \tag{40}$$

Problem (39), (40) is an integer linear programming one. In order to solve it, the known methods could be used - par example, Homori method or “branch and bound” method. But, due to the small dimension of the problem, the solution could be simply obtained by comparing all possible rational alternatives. The respective **algorithm A2** consists of the following:

1^{O}. One determines the server types interval $[\overset{\smile}{j}\,;\overset{\frown}{j}\,]$ from which it is reasonable to select server types for solving the problem. $\overset{\frown}{\lambda}_{0i} := \infty$, $i = \overline{1,n}$ and one calculates:

1) consecutively for $i = 1$, $i = 2$, ..., $i = n$, the saturation flows $\overset{\frown}{\lambda}_{ji}$, $j = \overline{1,m}$ according to expression (35);

2) the server type $\overset{\smile}{j}$, which can still be used for serving the request flows Λ_i, $i = \overline{1,n}$, by the condition of satisfying the restrictions (13)

$$\overset{\smile}{j} = \min_{j=\overline{0,m}} \left\{ j \middle| \overset{\frown}{\lambda}_{ji} > 0, \qquad i = \overline{1,n} \right\}.$$

If $\overset{\smile}{j} = 0$, then the problem has no solution.

3) The minimal number $\overset{\frown}{M}_j$ $\left(j = \overline{\overset{\smile}{j}\,, m} \right)$ of j type servers needed to serve the flows Λ_i, $i = \overline{1,n}$

$$\overset{\frown}{M}_j = \left] \max_{i=\overline{1,n}} \left\{ \frac{\Lambda_i}{\overset{\frown}{\lambda}_{ji}} > 0 \right\} \right[, \qquad j = \overline{1, \overset{\frown}{j}}\,;$$

4) taking into account relations (9) and (10), the type $\overset{\frown}{j}$ of servers of the largest capacity that can be rational to use for solving the problem

$$\widehat{j}= \min_{j=\overline{\underset{\smile}{j},m}}\Big\{j|\ \widehat{M}_j= 1\Big\}.$$

2$^{\text{O}}$. The type j^* and the optimal number $M^*_{j^*}$ of servers are determining as

$$M^*_{j^*} = \left\{\widehat{M}_{j^*}\ |C_{j^*}\ \widehat{M}_{j^*}= \min_{j=\overline{\underset{\smile}{j},\widehat{j}}}\Big\{C_j\ \widehat{M}_j\Big\}\right\}.$$

3$^{\text{O}}$. Since only one server type is used, the optimal (in respect of min T_{j^*i}, $i = \overline{1,n}$) distribution $\lambda^*_{j^*i}$, $i = \overline{1,n}$ of request flows Λ_i, $i = \overline{1,n}$ among servers $M^*_{j^*}$ is determined (see (24)) as

$$\lambda^*_{j^*i} = \frac{\Lambda_i}{M^*_{j^*}}, \qquad i = \overline{1,n}.$$

So, there has been obtained the type j^* and optimal number $M^*_{j^*}$ of servers and, additionally, the distribution $\lambda^*_{j^*i}$, $i = \overline{1,n}$ of request flows Λ_i, $i = \overline{1,n}$ among servers.

5.3 The server set could be constituted of servers of m types

5.3.1 There is only one request category: $n = 1$

It is required to determine the server set M_j, $j = \overline{1,m}$, which will serve the request flow $\Lambda_1 = \Lambda$. Since there is only one request category, index 1 will be omitted in all respective cases (par example, $\tau_{j1} = \tau_j$, $\lambda_{j1} = \lambda_j$). Taking into consideration (1), from relations (3), (13) for $n = 1$ one has:

$$C \;\; = \;\; \sum_{j=1}^{m} C_j M_j \to\ \min \qquad (41)$$

$$\sum_{j=1}^{m} M_j \overset{\frown}{\lambda}_j \;\; \geq \;\; \lambda. \tag{42}$$

For a single request category, values $\overset{\frown}{\lambda}_j$, $j = \overline{1, m}$ do not depend on $M_j, j = \overline{1, m}$ and are determined (see (35)) as

$$\overset{\frown}{\lambda}_j = \frac{2\,(T_d - \tau_j)}{\tau_j^{(2)} + 2\tau_j\,(T_d - \tau_j)}, j = \overline{1, m}. \tag{43}$$

Thus, problem (41)-(43) is an integer linear programming one. In order to solve it, one can use known algorithms, par example, the ones that realize the Homori method or the "branch and bound" method, taking into consideration the problem particularities. Below some particular cases are investigated, for which the problem (41)-(43) takes a simpler form.

At an exponential response time distribution and linear dependence of servers' cost on their capacity (see (10)), one has

$$C_j = aU_j = au\mu_j = b\mu_j, \qquad j = \overline{1, m}, \tag{44}$$

where $b = au$ is a coefficient, and from (20.1), taking into consideration (3.1) –

$$\overset{\frown}{\lambda}_j = \mu_j - \frac{1}{T_d}, \qquad j = \overline{1, m} \tag{45}$$

and then problem (41)-(43) is reducing to

$$C = b\sum_{j=1}^{m} M_j \mu_j \rightarrow \text{ min} \tag{46}$$

observing restriction

$$\sum_{j=1}^{m} M_j \mu_j - \frac{1}{T_d}\sum_{j=1}^{m} M_j \geq \Lambda. \tag{47}$$

Problem (46)-(47) is an integer linear programming one, but as a rule, of small dimensions. It could be solved also by comparing all possible rational variants.

From (46), (47) one could observe that the solution requires the use of servers with the largest possible capacity (if more than one server is needed) and, possibly, there will also be needed one or several servers of smaller capacity. First, one determines the maximum number $\widehat{M}_m$ of servers with the largest capacity μ_m. From (47), one has

$$\widehat{M}_m = \left]\Lambda \Big/ \left(\mu_m - \frac{1}{T_d}\right)\right[, \tag{48}$$

where][signify the roundup to integer of expression in square parentheses. The solution will contain, as a rule, $\widehat{M}_m$ or $\widehat{M}_m\ -1$, rarely $\widehat{M}_m\ -2$, servers of $j = 1$ type. There will be investigated the rational alternatives for $M_m \in [0; \widehat{M}_m]$ and servers of less capacity, taking into consideration relations (46), (47). One calculates the criteria C value for $\widehat{M}_m$, then for $\widehat{M}_m\ -1$ and, if necessary – for $\widehat{M}_m\ -2$ etc. Comparing the examined rational alternatives, the one that will assure the minimal cost C of server set will be determined.

The case when the number m of server categories is large and the values μ_j, $j = \overline{1, m}$ are relatively uniformly distributed in the interval $[\mu_1; \mu_m]$. For $m \to \infty$, it could be accepted that $\mu_j = f(j)$ is a continuous function of j. Then the following relations take place:

$$\lambda_j = \widehat{\lambda}_j, \qquad j = \overline{1, m}, \tag{49}$$

inequality (47) is transforming in equality

$$\sum_{j=1}^{m} M_j\ \widehat{\lambda}_j = \Lambda, \tag{50}$$

and the goal function (31), taking into consideration the equalities (20.3) and (36), becomes

$$C = b\left(\sum_{j=1}^{m} M_j\ \widehat{\lambda}_j + \frac{1}{T_d}\sum_{j=1}^{m} M_j\right) = b\left(\Lambda + \frac{1}{T_d}\sum_{j=1}^{m} M_j\right) \to \text{ min.} \tag{51}$$

Since values b, λ and T_d are known, the problem is reduced to minimizing expression $\sum\limits_{j=1}^{m} M_j$ in observing restriction (36). Evidently, the solution is

$$M_m = \left[\frac{\Lambda}{\overset{\frown}{\lambda}_m}\right], \quad M_{j_o} = 1, \tag{52}$$

where

$$j_o = \left\{ j | \overset{\frown}{\lambda}_j = \Lambda - M_m \overset{\frown}{\lambda}_m \right\} \tag{53}$$

or, in the case of the discrete form ($m \neq \infty$),

$$j_o = \min_{j=\overline{1,m}} \left\{ j | \overset{\frown}{\lambda}_j \geq \Lambda - M_m \overset{\frown}{\lambda}_m \right\}. \tag{54}$$

The other values of M_j are 0. In equation (52), the square parentheses [] signify the integer part of the value of expression in parentheses. In general, for $m \to \infty$ there exist an infinite number of solutions, from which only one is presented above. They are:

$$M_m = \left[\frac{\Lambda}{\overset{\frown}{\lambda}_m}\right] - 1 \tag{55}$$

and for each one server of categories $\{j_1; j_2\}$ that satisfy the condition

$$\overset{\frown}{\lambda}_{j_1} + \overset{\frown}{\lambda}_{j_2} = \Delta\Lambda, \tag{56}$$

where

$$\Delta\Lambda = \Lambda - (M_m - 1) \overset{\frown}{\lambda}_m < 2 \overset{\frown}{\lambda}_m .$$

To mention, that any value in the interval $\left[\overset{\frown}{\lambda}_{j_0}; \overset{\frown}{\lambda}_m\right]$ can be attributed to the rate $\overset{\frown}{\lambda}_{j_1}$ from (56), and $\overset{\frown}{\lambda}_{j_2} = \Delta\Lambda - \overset{\frown}{\lambda}_{j_1}$. This result can be used when there are many values for μ_j, $j = \overline{1,m}$ – the error will be small (it can be appraised).

5.3.2 The general case: there can be m server types and many request categories

Diverse methods can be used to solve the problem. For a not very large number m of server types, all possible rational alternatives can be compared. When forming the alternatives, one begins from servers with larger capacity. **Algorithm A3** for solving the problem (12), (38) consists of the following:

1^{O}. This step is the same as the p. 1^{O} of algorithm A2 from p. 5.2.

2^{O}. We note $m = \overset{\frown}{j} - \overset{\smile}{j} + 1$ and renumber server types in such a way, that type $\overset{\smile}{j}$ to be already type 1. So, the problem solution will be searched from server types $j = \overline{1, m}$.

3^{O}. $j := m$; $C^* := \infty$; $M_m := \overset{\frown}{M}_m$; $M_k := 0$, $k = \overline{1, m-1}$.

4^{O}. According to (12), the goal function value C is calculated. If $C < C^*$, then the information related to the new variant is saved, because it is better: $C^* := C$; $M^*_{j^*} := M_m$; $M^*_k := 0$, $k = \overline{1, m-1}$.

5^{O}. If $M_j = 0$, then we switch to p. 11^{O}.

6^{O}. $M_j = M_j - 1$. Consecutively for $i = 1$, $i = 2$, ..., $i = n$, according to algorithm A1, we calculate the request flows $\Delta\Lambda_{ji}$, $i = \overline{1, n}$ not yet distributed (the servers M_k, $k = \overline{j, m}$ being unable to serve them). These create flows $\Lambda_{j-1,i} = \Delta\Lambda_{ji}$, $i = \overline{1, n}$ remained for distribution among servers of types $j = \overline{1, j-1}$:

6.1^{O}. $i := 1$.

6.2^{O}. One determines the saturation flow $\overset{\frown}{\lambda}_{ji}$ (Λ_{ji}) according to (35) and, also,

$$\Delta\Lambda_{ji} = \Lambda_{ji} - M_j \overset{\frown}{\lambda}_{ji}$$

6.3^{O}. If $\Delta\Lambda_{ji} \geq 0$, then $\lambda_{ji} = \overset{\frown}{\lambda}_{ji}$. Otherwise $\Delta\Lambda_{ji} = 0$ and, taking into consideration relation (15), $\lambda_{ji} = \Lambda_{ji}/M_j$.

6.4$^{\rm O}$. If $i < n$, then $i := i + 1$ and switch to p. 6.2$^{\rm O}$.

6.5$^{\rm O}$. $\Lambda_{j-1,i} := \Delta\Lambda_{ji}, \qquad i = \overline{1,n}$.

7$^{\rm O}$. One determines the minimal number $\overset{\frown}{M}_{j-1}$ of $j - 1$ type servers needed to serve the flows $\Lambda_{j-1,i}$, $i = \overline{1,n}$ (algorithm A2 at $m = 1$ can be used):

7.1$^{\rm O}$. The saturation flows $\overset{\frown}{\lambda}_{j-1,i}$, $i = \overline{1,n}$ are determined according to expression (35), consecutively for $i = 1$, $i = 2$, ..., $i = n$.

7.2$^{\rm O}$. $\overset{\frown}{M}_{j-1} = \left]\max\limits_{i=\overline{1,n}}\left\{\frac{\Lambda_{j-1,i}}{\overset{\frown}{\lambda}_{j-1,i}} > 0\right\}\right[$.

8$^{\rm O}$. $M_{j-1} := \overset{\frown}{M}_{j-1}$. $M_{\rm k} := 0$, $k = \overline{1, j-2}$. The goal function value C is calculated according to (12). If $C < C^*$, then the information related to the new variant is saved, for it is better (the modifications only are saved): $C^* := C$; $M^*_{j^*} := M_j$; $M^*_{j-1^*} := M_{j-1}$; $M^*_k := 0$, $k = \overline{1, j-2}$.

9$^{\rm O}$. If $j > 2$, then $j := j - 1$.

10$^{\rm O}$. Switch to p. 5$^{\rm O}$.

11$^{\rm O}$. If $j < m$, then $j := j + 1$ and switch to p. 5$^{\rm O}$.

12$^{\rm O}$. The searched values $M^*_{j^*}$, $j^* = \overline{1,m}$ and C^* are obtained.

6 Conclusions

The problem for configuring LAN set of servers is formulated as a problem of nonlinear programming in integers. Solving it through the existent methods isn't guarantied. Then some particularities are emphasized, based on which there are determined certain rules that allow to considerably reducing the initial problem dimension.

Analytical expressions for calculating the server saturation flows are obtained and an algorithm for determining the possibilities of a given server set to serve user request flows is proposed. For the case when the server set must be constituted only from servers of the same type, the problem is reduced to an integer linear programming one. Taking into consideration the small dimension of this problem, an algorithm based on comparing all the possible rational alternatives is proposed.

A hypothetical case, that assumes the possibility to select servers from an infinite number of server types ($m \to \infty$) at a single request category is examined also; the analytical solution for this case is obtained. This solution could serve for appraising the tendencies with respect to solving real problems, including the influence of m value on the problem-solving path.

An algorithm for solving the general problem is proposed.

The analytical solutions obtained and the algorithms elaborated can be used to configure and assure an efficient utilization of the server set resources in LANs. There is elaborated a software application to be used for determining the server set and a rational distribution of user request flows among servers in LANs.

References

[1] G.Bolch, St.Greiner, H.Meer, K.S.Trivedi. *Queueing Networks and Markov Chains: Modeling and Performance Evaluation with Computer Science Applications.* 1998.

[2] I.Bolun. *Macrosinteza retelelor de calculatoare.* Chisinau: Editura ASEM, 1999.

[3] H.M.Ghafir, C.B.Silio, Jr., "Performance Analysis of a Multiple-Access Ring Network", *IEEE Transactions on Communications*, vol. **41**, no. **10**, October, 1993, pp. 1494–1506.

[4] L.Kleinrock, *Queueing Systems, Vol. II: Computer Applications*, New York: John Wiley Sons, Inc., 1976.

[5] T.G.Robertazzi. *Computer networks and systems: queuing theory and performance evaluation.* New York: Springer-Verlag, 1994.

[6] C.E.Spurgeon. *Ethernet The Definitive Guide.* O'Reilly&Association, Inc., 2000.

[7] W.Stallings, *Local and Metropolitan Area Networks*, 5th Edition, New York, NY, Macmillan Publishing Co., 1997.

[8] G.Thomas. "Improving the Performance of Ethernet Networks"/ *http://ethernet.industrial–networking.com/articles/i06articleimproving.asp*

Ion Bolun, Anatol Ciumac, Received April 25, 2002

Ion Bolun
Academy of Economic Studies,
Moldova, Chishinau
Phone: 22–27–65
E–mail: *bolun@ase.md*

Anatol Ciumac
Accent Electronic S.A.,
Moldova, Chishinau
Phone: 23–45–69
E–mail: *aciumac@accent.md*