ETLA ELINKEINOELÄMÄN TUTKIMUSLAITOS
THE RESEARCH INSTITUTE OF THE FINNISH ECONOMY

Keskusteluaiheita
Discussion Papers
13 December 2011

No 1261

Method for Throughput Maximization of Multiclass Networks with Flexible Servers

Valeriy Naumov* – Olli Martikainen**

* ETLA – The Research Institute of the Finnish Economy, valeriy.naumov@etla.fi
** ETLA – The Research Institute of the Finnish Economy, olli.martikainen@etla.fi

The authors thank the Technology Industries of Finland Centennial Foundation and Tekes – the Finnish Funding Agency for Technology and Innovation for research funding.

ISSN 0781–6847

Contents

Abstract

In this paper, we study the throughput of multiclass networks featuring several types of flexible servers as well as general constraints both on the number of servers having the same skills and on the number of servers allowed at each station. Each job class is characterized by its processing station, and each station is characterized by the amount of work assigned to that station upon job arrival. Servers may have different skills and efficiencies. We propose a simple method calculating an upper bound for the maximum network throughput achievable with static server allocation.

Key words: Server allocation, flexible server, multiclass network, throughput, bottleneck

JEL: C61, C62, C68

Tiivistelmä

Tässä artikkelissa tutkitaan joustavia erityyppisiä palvelimia käyttävän moniluokkaverkon läpäisyä, kun verkon solmuissa olevien palvelinten osaamisluokat ja kokonaismäärät noudattavat yleisiä rajoitusehtoja. Jokaiseen asiakasluokkaan liittyy verkon solmu, jossa asiakkaita palvellaan, sekä työmäärä, jonka asiakkaat tuovat solmuun. Palvelimissa sallitaan eri osaamisluokkia ja palvelutehokkuuksia. Esitämme yksinkertaisen tavan verkon maksimiläpäisyn ylärajan laskemiseksi staattisella palvelinallokaatiolla.

Asiasanat: Palvelinallokaatio, joustava palvelin, moniluokkaverkko, läpäisy, pullonkaula

1 Introduction

Stochastic network models with flexible servers are widely used for the analysis and optimization of computer and communication networks [11], manufacturing systems [12], and health care services [13]. In such models, "server flexibility" denotes a server's job processing capacity at different network stations. An example of a flexible server is a person working in activities performed at the different network stations, and a person's capacity at a given station is determined by his or her skill at carrying out the task at that station.

There are extensive lists of papers analyzing throughput of systems with flexible servers as exemplified by [2]-[5]. Andradottir et al. [2] propose linear programming (LP) to determine the optimal allocation of a given number of flexible servers in a multiclass network. Al-Azzoni and Down [8] use the same allocation LP model for mapping tasks onto flexible servers. Down and Karakostas [3] extend the LP model in [2] and study server allocation under a constraint on the number of servers at each station. In this paper, we consider a multiclass network featuring general constraints both on the number of servers having the same skills and on the number of servers allowed at a given station. Server skills and efficiency at different stations are represented in the form of a productivity matrix. Our analysis is based on three explicit assumptions: 1) Upon arrival at a station, each job inputs some quantity of work; 2) The aggregate productivity of servers allocated to station is an additive function of the individual servers' productivities; 3) The expected rate of work assigned to each station must not exceed the total productivity of servers allocated to that station. We outline a new formulation for the problem of optimal static server allocation and propose a method for the calculation of an upper bound on the network throughput.

Our formulation generalizes known problem formulations in three ways: A) Servers are divided into groups of servers having the same productivity; B) The number of servers found at a given station can be limited; C) We consider two types of networks: open networks, in which jobs can arrive and depart, and clopen networks, which forbid job arrival but permit job departure.

2 Description of the system

We consider a network composed of *N* stations, *K* job classes and *M* server types. The stations represent the job processing stage, and each station consists of infinite buffers as well as several servers that work in parallel. A job's class uniquely identifies the job's station, and a given job can change class after each processing stage. We use $\mathcal{C}(n)$ to denote the set of job classes processed at station *n*. Upon the completion of service, a job in class i is either routed to class j with probability p_{ij} or leaves the network with probability $1-\sum_{j=1}^{K} p_{ij}$. The transition from class i to class j may correspond to a transition of the job from one station to another, or it may represent the job's transition to another class within the same station. Recall that a network is open if jobs can both enter and leave the system. We call a network clopen if there are no arrivals, so that jobs can only leave the system. We assume that the matrix $\mathbf{I}-\mathbf{P}$ is invertible, which implies that each job class has a finite expected time to leave the network.

We assume that a job transmits to a station some volume of work that must be performed by servers, which are defined as independent exponentially distributed random variables. We denote by v_j the expected volume of work that each job of class j brings to a station. We consider networks having multi-skilled servers that are split into *M* types according to their functionality. We denote by b_m the number of servers of type *m*. A server's productivity is represented by a constant value that describes the volume of work that the server is able to process at a station per unit of time. We denote the productivity of type *m* servers at station n by π_{nm}. Servers of type *m* having zero productivity (i.e., for which $\pi_{nm} = 0$) will not be operational at station n. Thus, a server can receive jobs at a given station only if its productivity at the station is positive. For that matter, a server can be allocated to any station where it is operational. For example, servers of type 1 may be operational only at stations 1, 2 and 3, whereas servers of type 2 may be operational only at stations 2, 3 and 4. The $N \times M$ -matrix $\mathbf{\Pi} = [\pi_{nm}]$ will be called the productivity matrix corresponding to a given set of servers and stations. The resulting skill matrix $\mathbf{\Sigma} = [\sigma_{nm}]$ is defined by setting $\sigma_{nm} = 1$ if $\pi_{nm} > 0$ and by setting $\sigma_{nm} = 0$, if $\pi_{nm} = 0$. If a productivity matrix consists entirely of the elements 0 and 1, then it coincides with the skill matrix.

As defined, the productivity matrix $\mathbf{\Pi}$ has N rows and *M* columns, where N is the number of network stations and *M* is the number of server types. Each positive entry π_{nm} in $\mathbf{\Pi}$ corresponds to a skill pattern. For example, a positive value in the productivity matrix ($\pi_{nm} > 0$) indicates that each server of type *m* can process tasks at the station *n*. The number L of skill patterns is bounded as follows:

$$\max\{N, M\} \le L \le NM .$$

A productivity matrix can be represented by its fundamental digraph [1]. As depicted in Figure 1, the fundamental digraph $\mathcal{F}(\mathbf{\Pi})$ of $\mathbf{\Pi}$ has N vertices $\{s_1, s_2, \ldots, s_N\}$ representing network stations, *M* vertices $\{t_1, t_2, \ldots, t_M\}$ representing server types, and L arcs $\{a_1, a_2, \ldots, a_L\}$ representing skill patterns and connecting stations to server types. Note that in the digraph $\mathcal{F}(\mathbf{\Pi})$, there is an arc from vertex s_n to vertex t_m if and only if servers of type *m* are capable to process jobs at station *n*, i.e., if $\pi_{nm} > 0$. For each skill pattern *l,* we denote the indices of the tail s_n and the head t_m of arc a_l as $n = tail(l)$ and $m = head(l)$, respectively. Note that the fundamental digraph of the productivity matrix $\mathcal{F}(\mathbf{\Pi})$ and the fundamental digraph of the skill matrix $\mathcal{F}(\mathbf{\Sigma})$ are the same. We call this digraph the digraph of skill patterns.

Consider three matrices, an $N \times L$ -matrix $\mathbf{V}$, an $L \times L$ -matrix $\mathbf{U}$, and an $L \times M$ -matrix $\mathbf{W}$ defined by

$$\mathbf{V} = \begin{bmatrix} \delta_{1,tail(1)} & \delta_{1,tail(2)} & \cdots & \delta_{1,tail(L)} \\ \delta_{2,tail(1)} & \delta_{2,tail(2)} & \cdots & \delta_{2,tail(L)} \\ \cdots & \cdots & \cdots & \cdots \\ \delta_{N,tail(1)} & \delta_{N,tail(2)} & \cdots & \delta_{N,tail(L)} \end{bmatrix},$$

$$\mathbf{U} = \begin{bmatrix} \pi_{tail(1),head(1)} & & & \\ & \pi_{tail(2),head(2)} & & \mathbf{0} \\ & & \ddots & \\ \mathbf{0} & & & \pi_{tail(L),head(L)} \end{bmatrix},$$

$$\mathbf{W} = \begin{bmatrix} \delta_{head(1),1} & \delta_{head(1),2} & \cdots & \delta_{head(1),M} \\ \delta_{head(2),1} & \delta_{head(2),2} & \cdots & \delta_{head(2),M} \\ \cdots & \cdots & \cdots & \cdots \\ \delta_{head(L),1} & \delta_{head(L),2} & \cdots & \delta_{head(L),M} \end{bmatrix},$$

where $\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$. Matrix $\mathbf{V}$ contains a single 1 in each column, matrix $\mathbf{W}$ contains a single 1 in each row, and matrix $\mathbf{U}$ is diagonal. It is easy to verify that matrices $\mathbf{V}$, $\mathbf{W}$ and $\mathbf{U}$ provide a factorization of the skill matrix $\mathbf{F}$ and the productivity matrix $\mathbf{\Pi}$ as

$$\mathbf{F} = \mathbf{VW}, \quad \mathbf{\Pi} = \mathbf{VUW}.$$

Servers are assumed to be flexible in the sense that the sets of stations where servers are operational may overlap and each server can be allocated to any station at which that server is operational. The allocation of servers to stations can be described by an integer $N \times M -$ matrix $\mathbf{X} = [x_{nm}]$, where x_{nm} is the number of type *m* servers allocated to station n. Because the total number of allocated type *m* servers cannot exceed b_m, the matrix $\mathbf{X}$ must satisfy the following constraints:

$$\sum_{n=1}^{N} x_{nm} \leq b_m, \; m = 1,2,\ldots,M. \tag{1}$$

Let $\mathcal{N} = \{1,2,\ldots,N\}$ represent the set of all stations, and $\mathcal{G} = \{S_1, S_2, \ldots S_L\}$ denote a collection of non-empty subsets of $\mathcal{N}$. We assume that for each set of stations $S_i \in \mathcal{G}$, a positive number B_i is specified serving as an upper limit for the number of servers allocated to stations belonging to the set $S_i \in \mathcal{G}$. Therefore, only those server allocations are feasible that satisfy the constraints

$$\sum_{n \in S_i} \sum_{m=1}^{M} x_{nm} \leq B_i, \; i = 1,2,\ldots,L. \tag{2}$$

For example, if $\mathcal{G} = \{\{1\},\{2\},\ldots,\{N\},\mathcal{N}\}$, then the number of servers at station *n* has the upper bound B_n, $n = 1,2,\ldots,N$, whereas the total number of servers in the network cannot exceed B_{N+1}.

We assume that service discipline at each station is work-conservative, i.e., that no server is left idle at a station unless its queue is empty. Servers are assumed to cooperate in the sense that if there are multiple servers allocated to a station, then these servers pool their efforts, so that the aggregate productivity of all servers allocated to station n can be calculated as

$$\eta_n(\mathbf{X}) = \sum_{m=1}^{M} \pi_{nm} x_{nm} \text{ , } n = 1,2,\ldots,N \text{ .} \quad (3)$$

If $\pi_{nm} = 0$ for some station *n* and server type *m*, then the corresponding number of allocated servers should also have zero value (i.e., $x_{nm} = 0$) because the allocation of type *m* servers to station *n* cannot improve that station's performance. Therefore, $\mathbf{X}$ is only characterized by those entries corresponding to the L positive entries of the productivity matrix $\mathbf{\Pi}$. This implies that instead of needing $N \times M$ unknowns x_{nm}, we must only search for a vector **x** of length L that specifies the positive entries of $\mathbf{X}$. The server allocation vector $\mathbf{x} = (x_1, \ldots, x_L)$ corresponding to the server allocation matrix $\mathbf{X}$ is defined by

$$x_l = x_{nm},$$

where

$$n = tail(l),\ m = head(l).$$

The server allocation vector constraints on the total quantity of servers (1)-(2) and formula (3) can be rewritten as

$$\sum_{l=1}^{L} w_{ml} x_l \le b_m,\ m = 1,2,\ldots,M,$$

$$\sum_{n \in S_i} \sum_{l=1}^{L} v_{nl} x_l \le B_i,\ \ i = 1,2,\ldots,K,$$

$$\eta_n(\mathbf{X}) = \sum_{l=1}^{L} \varphi_{nl} x_l \text{ , } n = 1,2,\ldots,N,$$

where the matrix $\mathbf{\Phi} = [\varphi_{nl}]$ is defined by $\mathbf{\Phi} = \mathbf{VU}$.

3 Server allocation

3.1 Open networks

Assume that arriving jobs are routed to class j with probability α_j, where $\sum_{j=1}^{K} \alpha_j = 1$. The expected number of visits γ_j to class *j*, called the visiting ratio, can be uniquely determined by solving the following linear system [10]:

$$\gamma_j = \alpha_j + \sum_{i=1}^{K} \gamma_i p_{ij} \text{ , } j = 1,2,\ldots,K \text{ .}$$

The total expected workload at station *n*, which is the expected volume of work that each job places into station *n* during the job's lifetime, is given by

$$w_n = \sum_{j \in \mathcal{C}(n)} \gamma_j v_j \text{ .} \quad (4)$$

We assume that w_n is positive for each station n, i.e., that the system does not have superfluous stations.

In the load-proportional server allocation approach, servers of each type are assigned to every station at which they are operational, directly proportional to the workload w_n and inversely proportional to the server productivity at the station. Special multipliers k_n depend on the feasibility of the constraints from (2). In other words, in the load-proportional server allocation approach, the number of servers of type m allocated to station n is calculated as

$$x_{nm} = k_n b_m d_{nm}, \tag{5}$$

where

$$k_n = \begin{cases} 1, & \text{if } D_j \le B_j \text{ for all } S_j \text{ containing } n, \\ \min\limits_{\substack{1 \le i \le K \\ n \in S_i}} \left(\dfrac{B_i}{D_i} \right), & \text{otherwise,} \end{cases} \tag{6}$$

$$D_i = \sum_{n \in S_i} \sum_{m=1}^{M} b_m d_{nm}, \quad d_{nm} = \begin{cases} \dfrac{\left(\dfrac{w_n}{\pi_{nm}} \right)}{\sum\limits_{\substack{1 \le i \le N \\ \pi_{im} > 0}} \left(\dfrac{w_i}{\pi_{im}} \right)}, & \pi_{nm} > 0, \\ 0, & \text{otherwise}. \end{cases} \tag{7}$$

If the constraints from (2) are omitted, i.e., if $\mathcal{G} = \varnothing$, then $k_n = 1$ for all $n = 1,2,\ldots,N$ in (5). Load-proportional server allocation accounts for the constraints (1) and (2), but it also ignores the effect of the aggregate productivity expressed by formula (23).

The total expected service time required for processing a job at a station *n* over all its visits to the station can be calculated as

$$\tau_n(\mathbf{X}) = \frac{w_n}{\eta_n(\mathbf{X})}. \tag{8}$$

and the saturation rate of station n can be calculated as

$$\mu_n(\mathbf{X}) = \frac{\eta_n(\mathbf{X})}{w_n}. \tag{9}$$

At each station of a stable network, the job arrival rate a cannot be larger than the saturation rate [9]-[10]. Therefore, job arrival rate a is feasible only if it satisfies the inequality $a \le \lambda(\mathbf{X})$, where the network throughput $\lambda(\mathbf{X})$ is defined as

$$\lambda(\mathbf{X}) = \min_{1 \le n \le N} \mu_n(\mathbf{X}).$$

Equivalently, the job arrival rate a must satisfy the following constraints:

$$a w_n \le \eta_n(\mathbf{X}), \; n = 1,2,\ldots,N. \tag{10}$$

These constraints demand that the total expected amount of work placed into each station per unit of time does not exceed the total productivity of the servers allocated to the station.

These inequalities in (10) compose a necessary but not sufficient condition for network stability. Non-stable networks satisfying (10) have been observed in practice for a long time; see [14] for an example of a

non-stable communications network. Necessary and sufficient conditions for the stability of various stochastic networks can be found in the survey [15].

For any feasible job arrival rate a, the utilization of station n may be defined as follows:

$$\rho_n(\mathbf{X}) = \frac{a}{\mu_n(\mathbf{X})} \tag{11}$$

and the utilization of all type *m* servers may be defined as

$$u_m(\mathbf{X}) = \frac{1}{b_m} \sum_{n=1}^{N} \rho_n x_{nm} \,. \tag{12}$$

A bottleneck station is defined to be any station n for which the saturation rate $\mu_n(\mathbf{X})$ attains its minimum value $\lambda(\mathbf{X})$ [9]. It follows from (6) that bottleneck stations have the highest utilization in the system. We formulate the Throughput Maximization problem for an Open network (TMO) as the following integer programming problem:

(TMO)

maximize

$$\lambda(\mathbf{X}) = \min_{1 \le n \le N} \frac{1}{w_n} \sum_{m=1}^{M} \pi_{nm} x_{nm} \tag{13}$$

subject to:

$$\sum_{n=1}^{N} x_{nm} \le b_m \,, \qquad m = 1,2,\ldots,M \,, \tag{14}$$

$$\sum_{n \in S_i} \sum_{m=1}^{M} x_{nm} \le B_i \,, \quad i = 1,2,\ldots,K \,, \tag{15}$$

$$x_{nm} \in \mathbb{N}, \; n = 1,2,\ldots,N \,, \; m = 1,2,\ldots,M \,. \tag{16}$$

Here, $\mathbb{N}$ denotes the set of natural numbers, and the expected workloads w_n are calculated using (4). For an optimal allocation of servers, specified by a solution $\tilde{\mathbf{X}}$ to TMO, the value $\lambda(\tilde{\mathbf{X}})$ gives the maximum throughput achievable with static server allocation.

3.2 Clopen networks

Now consider a clopen network, for which at time $t = 0$ there are q_i class i jobs for $j = 1,2,\ldots,K$. Let $Q(t)$ represent the number of jobs in service within the network at time t and $\Theta = \inf_{t \ge 0} \{Q(t) = 0\}$ denote the time until the network is empty, or time-to-empty. We want to find an allocation of servers that minimizes the expected network time-to-empty, $\theta = \mathbb{E}\Theta$.

The expected number of visits of servers entering class j from class i, denoted γ_{ij}, can be uniquely determined by solving the linear system

$$\gamma_{ij} = \delta_{ij} + \sum_{k=1}^{K} \gamma_{ik} p_{kj} \ , \ i, j = 1,2,\ldots,K \ .$$

Taking into account the network's initial state, the expected number of class j jobs processed by the network before it becomes empty can be calculated as

$$Q_j = \sum_{k=1}^{K} q_k \gamma_{kj} \ . \tag{17}$$

This value also can be determined directly from the linear system

$$Q_j = q_j + \sum_{k=1}^{K} Q_k p_{kj} \ , \ j = 1,2,\ldots,K \ .$$

Therefore, the expected volume of work placed into station n can be calculated as

$$W_n = \sum_{j \in \mathcal{J}(n)} Q_j v_j \ , \tag{18}$$

and the total expected service time over all visits of all jobs at station n can be calculated as

$$T_n(\mathbf{X}) = \frac{W_n}{\eta_n(\mathbf{X})} \ . \tag{19}$$

Because the network time-to-empty cannot be less than any given station's time-to-empty, we deduce the following bound for the expected network time-to-empty:

$$\frac{1}{\theta} \le \min_{1 \le n \le N} \frac{1}{W_n} \sum_{m=1}^{M} \pi_{nm} x_{nm} \ . \tag{20}$$

Therefore, we can formulate the Time-to-empty Minimization problem for a Clopen network (TMC) as follows:

(TMC)

maximize

$$\Lambda(\mathbf{X}) = \min_{1 \le n \le N} \frac{1}{W_n} \sum_{m=1}^{M} \pi_{nm} x_{nm} \tag{21}$$

subject to:

$$\sum_{n=1}^{N} x_{nm} \le b_m \ , \qquad m = 1,2,\ldots,M \ , \tag{22}$$

$$\sum_{n \in S_i} \sum_{m=1}^{M} x_{nm} \le B_i \ , \quad i = 1,2,\ldots,K \ , \tag{23}$$

$$x_{nm} \in \mathbb{N} \ , \ n = 1,2,\ldots,N \ , \ m = 1,2,\ldots,M \ . \tag{24}$$

Note that the formulations of the server allocation problem for open and clopen networks are similar. However, the implied meaning of the corresponding objective functions is different.

4 Upper bounds for the maximum throughput

The general solution of integer programming problems presents a difficult task [6], as does the solution of the particular problem of throughput maximization in a network with heterogeneous flexible servers [3]. However, the maximum throughput of a network can be easily estimated. In this section, we give an upper bound for the throughput of open networks. A lower bound for the time-to-empty of a clopen network can be similarly derived.

4.1 The case $\pi_{nm} = \pi_n$ for all *n* and *m*

Assume that server productivities are independent of the server type, that $\pi_{nm} = \pi_n > 0$ for all *n* and *m,* and that $\mathbf{X}^*$ is a solution of the TMO. Then, for every station *n,* we have that

$$\lambda(\mathbf{X}^*)\frac{w_n}{\pi_n} \le \sum_{m=1}^{M} x^*_{nm} ,$$

and it follows from (23) that for each set $S_i \in \mathcal{G}$, the following inequality holds:

$$\lambda(\mathbf{X}^*)\sum_{n\in S_i}\frac{w_n}{\pi_n} \le \sum_{n\in S_i}\sum_{m=1}^{M} x^*_{nm} \le B_i .$$

Therefore, the maximum throughput of the network has the following upper bound:

$$\lambda(\mathbf{X}^*) \le \min_{1\le i\le K} \frac{B_i}{\sum_{n\in S_i}\frac{w_n}{\pi_n}}. \tag{25}$$

Note that the denominator in (25) is the total expected service time required for processing a job over all its visits to the set of stations S_i. This upper bound for network throughput in (25) generalizes a similar bound derived in [3].

4.2 General case

In the general case, a relaxation of the constraints in (24) can be used to obtain an upper bound for the maximum throughput. Consider the following Relaxed TMO (RTMO):

(RTMO)

Maximize

$$\lambda(\mathbf{X}) = \min_{1\le n\le N}\left(\frac{1}{w_n}\sum_{m=1}^{M}\pi_{nm}x_{nm}\right) \tag{26}$$

subject to:

$$\sum_{n=1}^{N} x_{nm} \le b_m , \qquad m = 1,2,\ldots,M , \tag{27}$$

$$\sum_{n\in S_i}\sum_{m=1}^{M} x_{nm} \le B_i, \qquad i=1,2,\ldots,K, \tag{28}$$

$$x_{nm} \ge 0, x_{nm} \in \mathbb{R}, \qquad n=1,2,\ldots,N,\ m=1,2,\ldots,M. \tag{29}$$

The only difference between RTMO and TMO is that here, the unknown variables x_{nm} may take on any nonnegative real values. Because the feasible space in RTMO is larger than in TMO, the value of the objective function $\lambda(\widetilde{\mathbf{X}})$ for a solution $\widetilde{\mathbf{X}}$ to RTMO yields an upper bound for the maximum network throughput $\lambda(\mathbf{X}^*)$ provided by the solution $\mathbf{X}^*$ from TMO.

For any solution $\mathbf{Y}$ to the RTMO, there exists a balanced solution $\mathbf{X}$ providing the same throughput as $\mathbf{Y}$ and satisfying the following equations:

$$\mu_n(\mathbf{X}) = \lambda(\mathbf{X}),\ n=1,2,\ldots,N. \tag{30}$$

For example, the matrix $\mathbf{X}$ defined by

$$x_{nm} = y_{nm}\frac{\lambda(\mathbf{Y})}{\mu_n(\mathbf{Y})}$$

satisfies conditions (27)-(30).

5 Solution of the relaxed throughput maximization problem

Starting with some initial server allocation, we can consistently increase network throughput by moving servers from non-bottleneck stations to bottleneck stations. Figure 3 illustrates a simplified throughput maximization procedure for RTMO in which the constraints from (28) have been omitted. This procedure is iterative and converges to a solution with required accuracy ε, where $0 < \varepsilon << 1$, while undertaking the following steps:

1. For each station *n* and server type *m*, compute the initial server quantities x_{nm} using load-proportional server allocation.
2. For each station *n*, compute the saturation rate $\mu_n(\mathbf{X})$ and network throughput $\lambda(\mathbf{X})$.
3. If the server allocation is balanced, then stop the server allocation procedure.
4. Select a node j^* for which $\mu_{j^*}(\mathbf{X}) = \lambda(\mathbf{X})$.
5. For each non-bottleneck station i and each server type k, compute the number of type k servers that can be relocated from station i to bottleneck station j^* in addition to the throughput gain achieved after relocation.

   5.1. If possible, compute the balancing number of type *k* servers that are required to be relocated from station i to station j^* to equate the saturation rates of stations *i* and j^* as

$$y_{i,k} = \begin{cases} \dfrac{\mu_i(\mathbf{X}) - \mu_{j^*}(\mathbf{X})}{\dfrac{\pi_{ik}}{w_i} + \dfrac{\pi_{j^*k}}{w_{j^*}}}, & \pi_{j^*k} > 0, \\ 0, & \text{otherwise.} \end{cases} \tag{31}$$

5.2. Compute the number of type k servers that can be relocated from station i to station j^* as

$$\Delta_{ik} = \min\{x_{ik}, y_{ik}\}. \tag{32}$$

5.3. Compute the throughput gain achieved after relocating Δ_{ik} servers of type k from station i to station j^* as

$$g_{ik} = \Delta_{ik} \frac{\pi_{j^*k}}{w_{j^*}}. \tag{33}$$

6. Select a station $i = i^*$ and a server type $k = k^*$ for which the throughput gain g_{ik} attains its maximum value.
7. Adjust the allocated quantities of servers by removing $\Delta_{i^*k^*}$ servers of type k^* from station i^* and adding $\Delta_{i^*k^*}$ servers of type k^* to station j^*; i.e., set

$$x_{i^*k^*} := x_{i^*k^*} - \Delta_{i^*k^*},\ x_{j^*k^*} := x_{j^*k^*} + \Delta_{i^*k^*}. \tag{34}$$

8. If $\Delta_{i^*k^*} > \varepsilon\lambda(\mathbf{X})$, then return to step 2. Otherwise, stop the server allocation procedure.

Before each iteration of this method, there is a group of bottleneck stations having the same saturation rate. One by one, the saturation rate of a bottleneck station in the group increases, and after increasing the saturation rate of the last bottleneck station in the group, a new group of bottleneck stations arises. The saturation rate in the new group of bottleneck stations is higher than that of the previous group of bottleneck stations. Therefore, the sequence of network throughputs calculated in each iteration of the method forms a non-decreasing, bounded and convergent sequence. When convergence is reached, $\lambda(\mathbf{X})$ gives the highest throughput that can be achieved with available servers.

6 Examples

The examples below illustrate the use of our proposed method for the analysis of an optimal company structure. We consider a company for which each arriving job requires sequential processing of one unit of work for each of three operations. All servers (workers) are split into five types, and Figure 3 depicts the number of servers of each type. For each server type, Figure 4 depicts the service rates for each operation. In the examples below, each station is dedicated to exactly one job class.

In Model 1, the company has two separate offices as depicted in Figure 5. Operation 1 is performed at stations 1 and 4, operation 2 is performed at stations 2 and 5 and operation 3 is performed at stations 3 and 6. Six servers from categories 1-3 are working in the first office, while seven servers from categories 4 and 5 are working in the second office. The corresponding productivity matrix for the company is depicted in Figure 6. Each job arrives into one of the two offices with equal probability. Therefore, the

expected number of visits to each station is equal to 0.5, as depicted in Figure 7. Figure 14 presents the results of load-proportional server allocation (5) and the corresponding saturation flows for Model 1. Figure 15 presents the results of calculating server allocation for Model 1 using our proposed throughput maximization procedure, as well as the corresponding saturation flows.

In Model 2, the company also has two separate offices, but station 3 and station 6 have been merged and operation 3 is only performed at station 3 in the first office, as depicted in Figure 8. The corresponding productivity matrix is shown in Figure 9, and the expected number of visits to each station is shown in Figure 10. Figure 16 presents the results of load-proportional server allocation (5) and the corresponding saturation flows for Model 2. Figure 17 presents the results of calculating server allocation for Model 2 using our proposed throughput maximization procedure, as well as the corresponding saturation flows.

In Model 3, the company only has one office as depicted in Figure 11. The corresponding productivity matrix is rendered in Figure 12. In this case, each job visits each station; therefore, the expected number of visits to each station is equal to 1, as depicted in Figure 13. Figure 18 presents the results of load-proportional server allocation (5) and the corresponding saturation flows for Model 3. Figure 19 presents the results of calculating server allocation for Model 3 using our proposed throughput maximization procedure, as well as the corresponding saturation flows.

For each model, Figure 20 depicts the throughputs achieved with load-proportional server allocation and with server allocation proposed by our throughput maximization procedure. Figure 21 depicts server utilization for a job arrival rate of $a = 1000$, achieved with our proposed method of server allocation.

7 Conclusions

In this paper, we have studied the performance of multiclass networks having several types of flexible servers. We have formulated the problem of optimal static server allocation in an open network as the integer programming problem of throughput maximization. We show that the problem of minimizing the average time-to-empty of a clopen network can be formulated similarly. Finding the solution to such problems for networks containing heterogeneous servers is difficult, and so we propose a method for calculating an upper bound for the maximum network throughput of open networks in addition to a lower bound for the minimum time-to-empty for clopen networks. Our results generalize some of the results found in [2, 3].

References

[1] Greenberg, H.J., Lundgren, J.R. and Maybee, J.S. Graph theoretic methods for the qualitative analysis of rectangular matrices. *SIAM J. Algebraic and Discrete Methods*, Vol. 2 (3), pp. 227–239, 1981.

[2] Andradottir, S., Ayhan, H. and D.G. Down, Dynamic server allocation for queueing networks with flexible servers, *Operations Research*, Vol. 51 (6), pp. 952-968, 2003.

[3] Down, D.G. and Karakostas, G.. Maximizung throughput in queueing networks with limited flexibility, *European J. of Operational Research*, Vol. 187(1), pp. 98-112, 2008.

[4] Hopp, W.J. and Van Oyen, M.P. Agile workforce evaluation: a framework for cross-training and coordination. *IIE Transactions*, Vol. 36 (10), pp. 919–940, 2004.

[5] Gurumurthi, S. and S. Benjaafar, Modeling and Analysis of Flexible Queueing Systems, *Naval Research Logistics*, Vol. 51, pp. 755-782, 2004.

[6] Yves Pochet and Laurence A. Wolsey. *Production Planning by Mixed Integer Programming*, Springer, 2006.

[7] O.Z. Aksin et.al. A review of workforce cross-training in call centers from an operations management perspective, in *Workforce Cross Training Handbook* (D. Nembhard ed.), CRC Press, pp. 211-240, 2007.

[8] Issam Al-Azzoni and D. G. Down, Linear programming based affinity scheduling for heterogeneous computing systems. IEEE Transactions on Parallel and Distributed Systems, Vol. 19(2), pp. 1671-1682, 2008.

[9] Denning, P.J. Throughput. *Wiley Encyclopedia of Computer Science and Engineering*, B.W. Wah (Ed.), Wiley-Interscience, 2008.

[10] Richard J. Boucherie and Nico M. van Dijk. *Queueing Networks: A Fundamental Approach*, Springer, 2011.

[11] Micha Pióro and Deepankar Medhi. *Routing, Flow, and Capacity Design in Communication and Computer Networks*, Elsevier, 2004.

[12] George Shanthikumar, David D. Yao and W. H. M. Zijm. *Stochastic Modeling and Optimization of Manufacturing Systems and Supply Chains*, Springer, 2003.

[13] François Sainfort, John Blake, Diwakar Gupta and Ronald L. Rardin. *Operations Research for Health Care Delivery Systems*, WTEC, 2005.

[14] O. Martikainen and M. Lahti. Performance analysis of OSI TP4/CLNP on FDDI, In *Proc. 2nd MultiG Workshop*, Stockholm, 17.6.1991, 1 – 14, 1991.

[15] Maury Bramson. Stability of queuing networks. *Probability Surveys*, vol. 15, pp.169-345, 2008.

Figure 1 Fundamental digrapgh of the productivity matrix

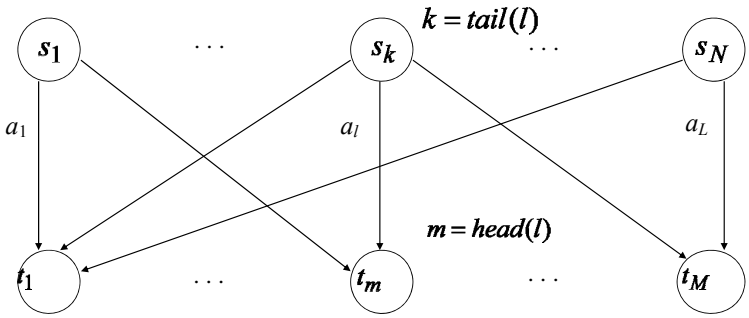

Figure 2 Throughput maximization procedure

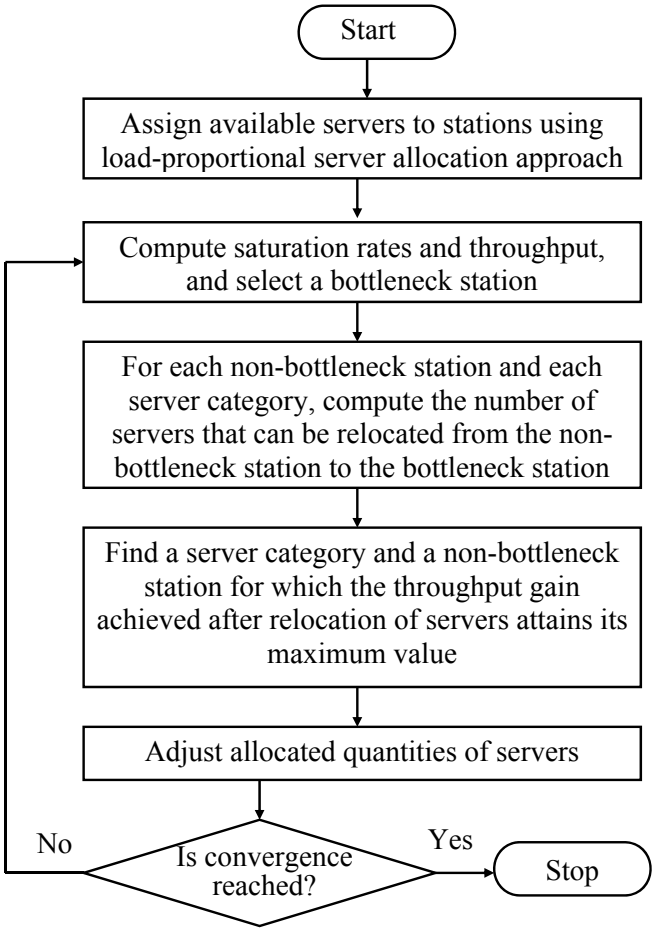

Figure 3 Quantities of servers

server type	1	2	3	4	5
server quantity	1	3	2	4	3

Figure 4 Service rates

operation	server type				
	1	2	3	4	5
1	0	300	300	330	330
2	2300	0	2300	0	2800
3	0	220	220	240	240

Figure 5 Model 1

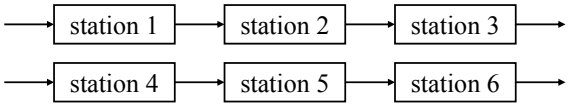

Figure 6 Productivity matrix for Model 1

station	server type				
	1	2	3	4	5
1	0	300	300	0	0
2	2300	0	2300	0	0
3	0	220	220	0	0
4	0	0	0	330	330
5	0	0	0	0	2800
6	0	0	0	240	240

Figure 7 Expected number of visits for Model 1

station	1	2	3	4	5	6
expected number of visits	0.5	0.5	0.5	0.5	0.5	0.5

Figure 8 Model 2

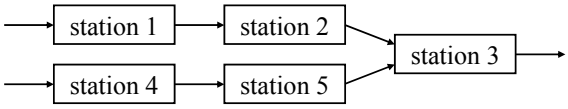

Figure 9 Productivity matrix for Model 2

station	server type				
	1	2	3	4	5
1	0	300	300	0	0
2	2300	0	2300	0	0
3	0	220	220	240	240
4	0	0	0	330	330
5	0	0	0	0	2800

Figure 10 Expected number of visits for Model 2

station	1	2	3	4	5
expected number of visits	0.5	0.5	1.0	0.5	0.5

Figure 11 Model 3

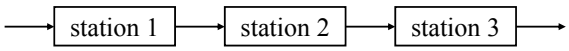

Figure 12 Productivity matrix for Model 3

station	server type				
	1	2	3	4	5
1	0	300	300	330	330
2	2300	0	2300	0	2800
3	0	220	220	240	240

Figure 13 Expected number of visits for Model 3

station	1	2	3
expected number of visits	1.0	1.0	1.0

Figure 14 Load-proportional server allocation and saturation flows for Model 1

station	saturation rate	server type				
		1	2	3	4	5
1	1242.680	0	1.269	0.801	0	0
2	5081.141	1	0	0.105	0	0
3	1242.680	0	1.731	1.094	0	0
4	1905.848	0	0	0	1.684	1.203
5	794.269	0	0	0	0	0.142
6	1905.848	0	0	0	2.316	1.655

Figure 15 Optimal server allocation and saturation flows for Model 1

station	saturation rate	server type				
		1	2	3	4	5
1	1269.231	0	1.269	0.846	0	0
2	4600.000	1	0	0	0	0
3	1269.231	0	1.731	1.154	0	0
4	1905.848	0	0	0	1.684	1.203
5	1818.092	0	0	0	0	0.325
6	1818.092	0	0	0	2.316	1.472

Figure 16 Load-proportional server allocation and saturation flows for Model 2

station	saturation rate	server type				
		1	2	3	4	5
1	793.992	0	0.805	0.518	0	0
2	4911.065	1	0	0.068	0	0
3	2009.904	0	2.195	1.414	2.933	2.133
4	1215.911	0	0	0	1.067	0.776
5	511.911	0	0	0	0	0.091

Figure 17 Optimal server allocation and saturation flows for Model 2

station	saturation rate	server type				
		1	2	3	4	5
1	1642.353	0	0.805	1.932	0	0
2	4911.065	1	0	0.068	0	0
3	1537.879	0	2.195	0	2.933	1.462
4	1537.879	0	0	0	1.067	1.263
5	1537.879	0	0	0	0	0.275

Figure 18 Load-proportional server allocation and saturation flows for Model 3

station	saturation rate	server type				
		1	2	3	4	5
1	1574.264	0	1.269	0.801	1.684	1.203
2	2937.705	1	0	0.105	0	0.142
3	1574.264	0	1.731	1.094	2.316	1.655

Figure 19 Optimal server allocation and saturation flows for Model 3

station	saturation rate	server type				
		1	2	3	4	5
1	1607.183	0	1.269	0.906	1.684	1.208
2	2300.000	1	0	0	0	0
3	1607.183	0	1.731	1.094	2.316	1.792

Figure 20 Network throughputs

model	load-proportional server allocation	optimal server allocation
1	794.269	1269.231
2	511.911	1537.879
3	1574.264	1607.183

Figure 21 Server utilizations for a job arrival rate of $a = 1000$

model	server type				
	1	2	3	4	5
1	0.217	0.788	0.788	0.539	0.540
2	0.204	0.639	0.595	0.650	0.650
3	0.435	0.622	0.622	0.622	0.622

Aikaisemmin ilmestynyt ETLAn Keskusteluaiheita-sarjassa

Previously published in the ETLA Discussion Papers Series

No 1246 *Heli Koski – Mika Pajarinen*, The Role of Business Subsidies in Job Creation of Start-ups, Gazelles and Incumbents. 07.04.2011. 21 p.

No 1247 *Antti Kauhanen*, The Perils of Altering Incentive Plans. A Case Study. 08.04.2011. 22 p.

No 1248 *Rita Asplund – Sami Napari*, Intangible Capital and Wages. An Analysis of Wage Gaps Across Occupations and Genders in Czech Republic, Finland and Norway. 11.04.2011. 18 p.

No 1249 *Mari Kangasniemi – Antti Kauhanen*, Performance-related Pay and Gender Wage Differences. 21.04.2011. 19 p.

No 1250 *Ye Zhang*, Wireless Acquisition of Process Data. 24.05.2011. 52 p.

No 1251 *Rita Asplund – Erling Barth – Per Lundborg – Kjersti Misje Nilsen*, Challenges of Nordic Labour Markets: A Polarization of Working Life? 08.06.2011. 21 p.

No 1252 *Jari Hyvärinen*, Innovaatiotoiminta: Näkemyksiä ympäristö- ja energia-alaan. 1.6.2011. 39 s.

No 1253 *Ari Hyytinen – Mika Maliranta*, Firm Lifecycles and External Restructuring. 17.06.2011. 34 p.

No 1254 *Timo Seppälä – Olli Martikainen*, Europe Lagging Behind in ICT Evolution: Patenting Trends of Leading ICT Companies. 22.06.2011. 18 p.

No 1255 *Paavo Suni – Pekka Ylä-Anttila*, Kilpailukyky ja globaalin toimintaympäristön muutos. Suomen koneteollisuus maailmantaloudessa. 19.08.2011. 39 s.

No 1256 *Jari Hyvärinen*, Innovaatiotoiminta: Näkemyksiä hyvinvointialaan ja työelämän kehittämiseen. 31.8.2011. 28 s.

No 1257 *Terttu Luukkonen – Matthias Deschryvere – Fabio Bertoni – Tuomo Nikulainen*, Importance of the Non-financial Value Added of Government and Independent Venture Capitalists. 2.9.2011. 28 p.

No 1258 *Ari Hyytinen – Mika Pajarinen – Pekka Ylä-Anttila*, Finpron vaikuttavuus – Finpron palveluiden käytön vaikutukset yritysten kansainvälistymiseen ja menestymiseen. 15.9.2011. 32 p.

No 1259 *Kari E.O. Alho*, How to Restore Sustainability of the Euro? 19.9.2011. 27 p.

No 1260 *Heli Koski*, Does Marginal Cost Pricing of Public Sector Information Spur Firm Growth? 28.9.2011. 15 p.

Elinkeinoelämän Tutkimuslaitoksen julkaisemat "Keskusteluaiheita" ovat raportteja alustavista tutkimustuloksista ja väliraportteja tekeillä olevista tutkimuksista. Tässä sarjassa julkaistuja monisteita on mahdollista ostaa Taloustieto Oy:stä kopiointi- ja toimituskuluja vastaavaan hintaan.

Papers in this series are reports on preliminary research results and on studies in progress. They are sold by Taloustieto Oy for a nominal fee covering copying and postage costs.

Julkaisut ovat ladattavissa pdf-muodossa osoitteessa: www.etla.fi/julkaisuhaku.php
Publications in pdf can be downloaded at www.etla.fi/eng/julkaisuhaku.php

ETLA
Elinkeinoelämän Tutkimuslaitos
The Research Institute of the Finnish Economy
Lönnrotinkatu 4 B
00120 Helsinki

Puh. 09-609 900
Fax 09-601 753
www.etla.fi
etunimi.sukunimi@etla.fi

ISSN 0781–6847