Economides, A.A.: Hierarchical resource sharing: NCP & VIP formulation.
Information & Decision Technologies vol. 19, pp. 379-392, 1994

Hierarchical resource sharing

NCP and VIP formulation

Anastasios A. Economides

University of Macedonia, Egnatia 156, Thessaloniki 540 06, Greece

Received April 1991
Revised August 1993
Communicated by M.G. Singh

Previous studies on problems in distributed systems and computer networks assume that different users (or classes of jobs) have equal rights in using the system resources. In this paper I consider systems with two hierarchical classes of jobs. Jobs of the most powerful class (e.g. higher priority) use the system resources for their own benefit, ignoring the inconvenience that they cause to jobs of the less powerful class. I consider the two-hierarchical class joint load sharing, routing and congestion control problem and formulate it as a Stackelberg game. The most powerful class is the leader, while the less powerful class is the follower. I derive the equilibrium conditions formulating the problem both as a non-linear complementarity problem (NCP) and as a variational inequality problem (VIP).

1. Introduction

Jobs arrive at the source nodes of a distributed system requiring processing and communication. However, not all of the jobs are admitted into the system, because this may cause saturation of some system resources. The problem of controlling these externally arriving jobs is the *congestion control* problem. The processing of jobs can be done at the source nodes or at some other destination nodes. The problem of selecting the node for processing a job is the *load-sharing* problem. After selecting the destination node for processing, the job should be transferred there. Also, jobs arrive at the source nodes simply requiring transfer to a specific destination node. A job that requires transfer to a destination node may arrive there through one of several paths. The problem of selecting the best path for transferring them to their destination is the *routing* problem. All these externally arriving jobs are assigned to the distributed system resources in an efficient way, such that some cost function is minimized. While previous research has concentrated on each problem in isolation, or at most on two problems combined, it is important to analyze all problems simultaneously, since the decisions for each problem affects those of another. In this paper I consider the multi-objective *joint* load-sharing, routing and congestion control problem.

Previous research on each of the above problems assumes a single class of jobs and the optimization of a single objective function. However, in a distributed system there are multiple classes of jobs, each with possibly different objectives. It is quite common to require differentiated service among different classes by assigning different priorities to different classes, for example interactive jobs have higher priority than batch jobs. A high priority class may acquire most of the resources that it needs, while a

North-Holland
Information and Decision Technologies 19 (1994) 379–392

SSDI 0923-0408(94)00550-Q

low priority class should wait for the high priority class to complete service. Since the reason for having priorities is to give preferential treatment to the high priority jobs, it is *not* meaningful to define a *single* multi-objective function (e.g. a convex combination of the objective functions of the different priority classes) for global optimization *across all the priority classes simultaneously*. However, we can still optimize the behavior of jobs within each priority class. Therefore a different approach should be taken for performance optimization of multi-priority systems. Also, it is desirable that the system administrator has more power than the system users and frequently he/she wants to impose his/her decisions on the users. So, the administrator (leader) directs the users (followers) according to his/her objectives. In this paper, I consider two-hierarchical classes and define this *hierarchical resource-sharing problem* as a *Stackelberg game*. Furthermore, I provide two alternative formulations for solving this problem. In the *non-linear complementarity problem* (NCP) formulation, I express the Stackelberg equilibrium conditions of the problem as a NCP. In the *variational inequality problem* (VIP) formulation, I express the Stackelberg equilibrium conditions of the problem as a VIP. These formulations open the theory and algorithms of NCP and VIP [17, 30, 32] for the study of particular applications.

2. Previous studies

Previous studies on each one of the load-sharing, routing and congestion control problems formulate the problem as a non-linear programming problem.

For the load-sharing problem: Buzen and Chen [13] and Ni and Hwang [41, 42] study the load-sharing problem for a multiprocessor, where there is no communication network. de Souza e Silva and Gerla [19] model the system as a product form queueing network with fixed closed chain routing. They minimize a measure of the average delay with respect to the open chain flows. Tantawi and Towsley [52] model each computer site as well as the network as single nodes. They minimize the expected job response time. Kurose and Simha [36–38] use first and second derivative algorithms, as well as [50] stochastic approximation algorithms for load sharing. Lin et al. [39] consider the joint load-sharing and routing problem and minimize a linear combination of the expected job response time at the computer sites and the expected message delay in the network.

For the routing problem: Wardrop [53] calculates the traffic patterns according to the user optimization criterion and the system optimization criterion, for a network consisting of two nodes connected by n independent paths. Dafermos and Sparrow [18] develop the optimality conditions for system and user optimum. Fratta et al. [26] use a steepest descent method, called flow deviation, to find the flow assignment. Cantor and Gerla [14] propose a method, based on decomposition techniques, for finding the flow assignment. Schwartz and Cheung [47] propose the gradient projection algorithm to find the flow assignment, for networks with a relative small number of commodities, since it needs less computation time than the flow deviation method. Stern [51] uses relaxation methods to find the flow assignment. Gallager [27, 28] proposes a distributed loop-free algorithm for finding the flow assignment. Bertsekas [2–4] proposes a gradient projection algorithm. Bertsekas et al. [5, 6] also propose second derivative algorithms, which may be viewed as approximations to a constrained version of Newton's method. Kobayashi and Gerla [35] consider the single-objective multiple class routing problem in closed queueing networks. Each closed chain corresponds to a different class of customers. They maximize the throughput such that the end-to-end expected delay is bounded. However, the average delay is not convex for closed chains routing. Economides and Silvester [23] formulate and solve the routing problem in networks with variable quality links. Some links have a high error rate and therefore packets fail and must be retransmitted. They also [21, 22, 25] formulate and solve the multiple class routing problem as a team optimization problem when the classes of packets cooperate, and as a Nash game when the classes of packets compete among themselves.

For the congestion control problem: Gallager and Golestaani [29] use a penalty function approach to model and solve the joint flow control and routing problem. They find the rates that achieve a tradeoff between user cost functions and network congestion cost. Then a dynamic flow control algorithm admits or rejects individual packets. Hahne and Gallager [31] investigate the round robin scheduling for fair flow control. They maximize the minimum packet rate of a virtual circuit. Jaffe [33] suggests that each virtual circuit adjusts its throughput rate to achieve an ideal delay–throughput tradeoff. Bharath-Kumar [7] maximizes the power by selecting the rate at which messages are allowed to enter the message path and the average total number in the system. Bharath-Kumar and Jaffe [8] investigate flow control algorithms to maximize the power. They control the rate of message entry to every virtual circuit.

In all previous studies, the different users (or classes of jobs) have equal rights in using the system resources. So, an optimization problem was formulated and solved. In this paper, I consider that different users (or classes of jobs) have different rights, so I attack the problem in a different way. I consider distributed systems with two-hierarchical classes of jobs. Jobs of the most powerful class use the system resources for their own benefit, ignoring the inconvenience that they cause to jobs of the less powerful class. For problems in such systems, I develop a methodology based on the Stackelberg game theory. The first paper to formulate and solve a problem in distributed systems as a Stackelberg game was by Economides and Silvester [24]. There, we formulate and solve the multi-objective load-sharing problem for two preemptive priority classes of jobs as a Stackelberg game. Here, I extend the results of that paper. I consider the joint load-sharing, routing and congestion control problem for two-hierarchical classes of jobs in an arbitrary distributed system. Furthermore, I formulate the problem as a non-linear complementarity problem (NCP) and as a variational inequality problem (VIP) for Stackelberg equilibrium.

My approach is based on the theory of Stackelberg games. For the reader interested in this theory, I briefly survey research on static Stackelberg games. These problems have been named Stackelberg games after the pioneering work of von Stackelberg (1938) on two-level optimization problems.

Bracken and McGill [12] formulate mathematical programs with optimization problems in the constraints, i.e. Stackelberg games. Polak and Mayne [46] present an algorithm for minimizing a function subject to functional inequality constraints. Blankenship and Falk [11] describe a generalized cutting-plane algorithm for constrained optimization and minimax problems.

Simaan [49] considers Stackelberg games for two-level optimization problems. Castanon and Sandel [15] illustrate the non-convexity of Stackelberg games and suggest that the leader's cost function must be selected carefully.

Papavassilopoulos [43, 44] describes algorithms for leader–follower games. He notices that difficulties arise due to the non-convex character of the follower's reaction set. He also [45] solves the linear quadratic Gaussian static Nash and Stackelberg games. He presents necessary and sufficient conditions for existence and uniqueness of the solution as well as procedures for finding all the solutions.

Bialas and Karwan [9, 10] present techniques for Stackelberg games. Shimizu and Aiyoshi [48] apply the penalty method to solve the Stackelberg game. Chang and Luh [16] and Luh et al. [40] derive necessary and sufficient conditions for Stackelberg games via the inducible region concept.

In the next section I introduce the analytical model that incorporates the control variables for the load-sharing, routing and congestion control decisions. The form of the joint load-sharing, routing and congestion control problem remains the same across almost all systems. Consequently, my basic model is versatile enough to model a broad range of applications.

3. Network model

In this section, I introduce a simplified model for an arbitrary distributed system. The purpose of the paper is not to solve a particular application, but to develop the framework for solving a large class of applications. Therefore, I introduce only the control variables that are necessary in setting up this framework. For the same reason, it is not necessary to be specific about the objective functions of the two-hierarchical classes, because this will not add to the main ideas of the paper. However, for the reader interested in specific objective functions, I refer to [20,24].

First, I develop the analytical model that integrates the load-sharing, routing and congestion control problems. I consider a distributed system as seen by jobs of class c, as a set of source nodes, $\boldsymbol{S}^c$, destination nodes for a given source node $[s.]$, $\boldsymbol{D}^c_{[s.]}$, source–destination pairs, $\boldsymbol{SD}^c$, and paths between a given source–destination pair $[sd]$, $\boldsymbol{\Pi}^c_{[sd]}$.

The formulation of the joint load-sharing, routing and congestion control problem can be done either on the link flow space or on the path flow space. In future high speed computer communication networks the transmission delay will be extremely small and we will not want to spend extra time in network management decisions inside the network. Therefore, the computationally intensive processes, such as the network management decisions, will be transferred outside of the network either to the source or to the destination node. With this in mind, I formulate the joint load-sharing, routing and congestion control problem on the path flow space, which means that the routing decisions will be done at the source nodes. In this way, I also avoid loops, since the packets will follow a previously determined loop-free path.

Class c jobs that require processing arrive at the source node $[s.]$. A load-sharing decision is made as to where these jobs will be processed. Let the fraction of jobs sent to node $[.d]$ for processing be $\psi^c_{[sd]}$. Since only one destination node is selected, the sum of the load-sharing fractions from node $[s.]$ to all destination nodes $[.d]$ is equal to one. So, let us define the constraint set for the class c load sharing decision variables to be

$$\boldsymbol{LS}^c = \Bigg\{ \psi^c_{[sd]} \quad \forall [.d] \in \boldsymbol{D}^c_{[s.]} \quad \forall [s.] \in \boldsymbol{S}^c / \text{such that}$$

$$\sum_{[.d] \in \boldsymbol{D}^c_{[s.]}} \psi^c_{[sd]} = 1 \quad \forall [s.] \in \boldsymbol{S}^c, \quad \text{and}$$

$$\psi^c_{[sd]} \geq 0 \quad \forall [.d] \in \boldsymbol{D}^c_{[s.]},\ [s.] \in \boldsymbol{S}^c \Bigg\}$$

and for all classes $\boldsymbol{LS} = \{\ldots \boldsymbol{LS}^c \ldots\}$.

For a more compact presentation we write the load-sharing fractions from all source nodes to all destination nodes for class c as the vector $\boldsymbol{\Psi}^c = [\ldots \psi^c_{[sd]} \ldots]$ and for all classes as the vector $\boldsymbol{\Psi} = [\ldots \boldsymbol{\Psi}^c \ldots]$.

After it is decided which fraction of jobs will be processed by destination node $[.d]$, they should be transferred there – this is the routing problems. For routing, we must specify which path between source–destination pair $[sd]$ will be selected. In addition, some jobs may be rejected outside of the network, for congestion control reasons. So, let the fraction of rejected jobs for the source–destination pair $[sd]$ be $\phi^c_{o[sd]}$, and the fraction of jobs routed through path $\pi[sd]$ be $\phi^c_{\pi[sd]}$. Since a job for the source–destination pair may only be rejected or routed through a single path, all these fractions must sum to one. So, let us define the constraint set of the class c routing and congestion control decision variables as

$$\boldsymbol{RC}^c = \Bigg\{ \phi^c_{o[sd]}, \phi^c_{\pi[sd]} \quad \forall \pi[sd] \in \boldsymbol{\Pi}^c_{[sd]} \quad \forall [sd] \in \boldsymbol{SD}^c / \text{such that}$$

$$\phi^c_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^c_{[sd]}} \phi^c_{\pi[sd]} = 1 \quad \forall [sd] \in \boldsymbol{SD}^c$$

$$\phi^c_{o[sd]}, \phi^c_{\pi[sd]} \geq 0 \quad \forall \pi[sd] \in \boldsymbol{\Pi}^c_{[sd]},\ [sd] \in \boldsymbol{SD}^c \Bigg\}$$

and for all classes $\boldsymbol{RC} = \{\ldots \boldsymbol{RC}^c \ldots\}$.

For a more compact presentation I write the routing and congestion control fractions of all paths between all source–destination pairs for class c as the vector $\boldsymbol{\Phi}^c = [\ldots \phi^c_{o[sd]} \ldots \phi^c_{\pi[sd]} \ldots]$, and for all classes as the vector $\boldsymbol{\Phi} = [\ldots \boldsymbol{\Phi}^c \ldots]$.

Finally, let the cost function for class c, $J^c(\boldsymbol{\Phi}, \boldsymbol{\Psi})$, be non-negative, continuously differentiable and convex with respect to $(\boldsymbol{\Phi}, \boldsymbol{\Psi}) \in (\boldsymbol{RC}, \boldsymbol{LS})$. Also, the feasibility set $(\boldsymbol{RC}, \boldsymbol{LS})$ is considered to be non-empty.

4. Hierarchical resource sharing

In this section I define the hierarchical resource-sharing problem using the Stackelberg game theory.

Next, I give some definitions for my two-hierarchical class problem similar to those in [1] for Stackelberg games:

Definition 1. In a two-hierarchical class joint load-sharing, routing and congestion control problem, with the most powerful (e.g. high priority) class α as the leader and the less powerful (e.g. low priority) class β as the follower, the set $\mathcal{R}^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha)$, defined for the class α strategy $(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha) \in (\boldsymbol{RC}^\alpha, \boldsymbol{LS}^\alpha)$, by

$$\begin{aligned}\mathcal{R}^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha) = \{(\boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta) \in (\boldsymbol{RC}^\beta, \boldsymbol{\Psi}^\beta) \text{ such that } \\ J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta) \leq J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \hat{\boldsymbol{\Phi}}^\beta, \hat{\boldsymbol{\Psi}}^\beta), \\ \forall(\hat{\boldsymbol{\Phi}}^\beta, \hat{\boldsymbol{\Psi}}^\beta), \text{ such that } (\hat{\boldsymbol{\Phi}}^\beta, \hat{\boldsymbol{\Psi}}^\beta) \in (\boldsymbol{RC}^\beta, \boldsymbol{LS}^\beta)\}\end{aligned}$$

is the *optimal response* (rational reaction) set of the less powerful class β to the strategy of the most powerful class α.

The above definition says that the less powerful class β chooses its decision vector $(\boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)$, which minimizes its cost function $J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)$, for given strategy $(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha)$ of the most powerful class α.

Definition 2. In a two-hierarchical class joint load-sharing, routing and congestion control problem with the most powerful (e.g. high priority) class α as the leader, a strategy $(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *}) \in (\boldsymbol{RC}^\alpha, \boldsymbol{LS}^\alpha)$ is called a *Stackelberg equilibrium* strategy for the most powerful class α if and only if

$$\begin{aligned}&\inf_{(\boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta) \in \mathcal{R}^\beta(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *})} J^\alpha(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *}, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta) \\ &\leq \inf_{(\boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta) \in \mathcal{R}^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha)} J^\alpha(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta) \quad \forall(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha) \in (\boldsymbol{RC}^\alpha, \boldsymbol{LS}^\alpha).\end{aligned}$$

This means that the most powerful class α chooses its strategy $(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *})$, which minimizes its cost function $J^\alpha(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)$, given the optimal response set $\mathcal{R}^\beta(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *})$ of the less powerful class β to its strategy $(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *})$.

Definition 3. Let $(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *}) \in (\boldsymbol{RC}^\alpha, \boldsymbol{LS}^\alpha)$ be a Stackelberg equilibrium strategy for the most powerful (e.g. high priority) class α. Then any element $(\boldsymbol{\Phi}^{\beta *}, \boldsymbol{\Psi}^{\beta *}) \in \mathcal{R}^\beta(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *})$ is an optimal strategy for the less powerful (e.g. low priority) class β that is in equilibrium with $(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *})$. The strategy $(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *}, \boldsymbol{\Phi}^{\beta *}, \boldsymbol{\Psi}^{\beta *})$ is a *Stackelberg equilibrium solution* for the game with the most powerful class α as the leader and the cost pair $J^\alpha(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *}, \boldsymbol{\Phi}^{\beta *}, \boldsymbol{\Psi}^{\beta *})$, $J^\beta(\boldsymbol{\Phi}^{\alpha *}, \boldsymbol{\Psi}^{\alpha *}, \boldsymbol{\Phi}^{\beta *}, \boldsymbol{\Psi}^{\beta *})$ is the corresponding *Stackelberg equilibrium outcome*.

Having defined the two-hierarchical class problem as a Stackelberg game, I express it as the following *hierarchical resource-sharing problem*:

$$\begin{aligned}
&\text{minimize} && J^{\alpha}(\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha}, \boldsymbol{\Phi}^{\beta}, \boldsymbol{\Psi}^{\beta})\\
&\text{with respect to} && (\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha}, \boldsymbol{\Phi}^{\beta}, \boldsymbol{\Psi}^{\beta})\\
&\text{such that} && (\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha}) \in (\boldsymbol{RC}^{\alpha}, \boldsymbol{LS}^{\alpha}),\ (\boldsymbol{\Phi}^{\beta}, \boldsymbol{\Psi}^{\beta}) \in (\boldsymbol{RC}^{\beta}, \boldsymbol{LS}^{\beta})\\
& && J^{\beta}(\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha}, \boldsymbol{\Phi}^{\beta}, \boldsymbol{\Psi}^{\beta}) = \min_{(\hat{\boldsymbol{\Phi}}^{\beta},\, \hat{\boldsymbol{\Psi}}^{\beta}) \in (\boldsymbol{RC}^{\beta}, \boldsymbol{LS}^{\beta})} J^{\beta}(\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha}, \hat{\boldsymbol{\Phi}}^{\beta}, \hat{\boldsymbol{\Psi}}^{\beta}) .
\end{aligned}$$

The leader α chooses $(\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha})$ (his routing, congestion control and load-sharing fractions) to minimize $J^{\alpha}(\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha}, \boldsymbol{\Phi}^{\beta}, \boldsymbol{\Psi}^{\beta})$ (his cost function) by taking into account the rational reactions of the follower β. He announces his decision first and enforces it on the follower. Then the follower β chooses $(\boldsymbol{\Phi}^{\beta}, \boldsymbol{\Psi}^{\beta})$ (his routing, congestion control and load-sharing fractions) to minimize $J^{\beta}(\boldsymbol{\Phi}^{\alpha}, \boldsymbol{\Psi}^{\alpha}, \boldsymbol{\Phi}^{\beta}, \boldsymbol{\Psi}^{\beta})$ (his cost function) subject to the imposed decision of the leader.

In [24] we explicitly solve a simple case of the above problem. We consider a load-sharing problem in a system with two preemptive resume priority classes of jobs, where each class minimizes its average job delay. In the next section I proceed to formulate the above general problem as a NCP and a VIP.

5. NCP and VIP formulations

In this section I formulate the hierarchical resource–sharing problem as a non-linear complementarity problem (NCP) and as a variational inequality problem (VIP).

First, for each class $c \in \{\alpha, \beta\}$, I define

$l_{o[sd]}^{c,\text{Stack}}$: the length for the rejected flow $[sd]$,

$l_{\pi[sd]}^{c,\text{Stack}}$: the length for the path $\pi[sd]$, and

$l_{[sd]}^{c,\text{Stack}}$: the length for the source–destination pair $[sd]$.

$$\begin{aligned}
l_{o[sd]}^{\alpha,\text{Stack}} \triangleq{}& \frac{\partial J^{\alpha}}{\partial \phi_{o[sd]}^{\alpha}} + \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} Q_{o[\hat{s}\hat{d}]}^{\alpha\beta} * \frac{\partial^2 J^{\beta}}{\partial \phi_{o[sd]}^{\alpha}\, \partial \phi_{o[\hat{s}\hat{d}]}^{\beta}} * \phi_{o[\hat{s}\hat{d}]}^{\beta}\\
&+ \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \sum_{p[\hat{s}\hat{d}] \in \boldsymbol{\Pi}_{[\hat{s}\hat{d}]}^{\beta}} Q_{p[\hat{s}\hat{d}]}^{\alpha\beta} * \frac{\partial^2 J^{\beta}}{\partial \phi_{o[sd]}^{\alpha}\, \partial \phi_{p[\hat{s}\hat{d}]}^{\beta}} * \phi_{p[\hat{s}\hat{d}]}^{\beta}\\
&+ \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} Q_{[\hat{s}\hat{d}]}^{\alpha\beta} * \frac{\partial^2 J^{\beta}}{\partial \phi_{o[sd]}^{\alpha}\, \partial \psi_{[\hat{s}\hat{d}]}^{\beta}} * \psi_{[\hat{s}\hat{d}]}^{\beta} + \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \bar{Q}_{o[\hat{s}\hat{d}]}^{\alpha\beta} * \left(- \frac{\partial^2 J^{\beta}}{\partial \phi_{o[sd]}^{\alpha}\, \partial \phi_{o[\hat{s}\hat{d}]}^{\beta}} \right)\\
&+ \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \sum_{p[\hat{s}\hat{d}] \in \boldsymbol{\Pi}_{[\hat{s}\hat{d}]}^{\beta}} \bar{Q}_{p[\hat{s}\hat{d}]}^{\alpha\beta} * \left(- \frac{\partial^2 J^{\beta}}{\partial \phi_{o[sd]}^{\alpha}\, \partial \phi_{p[\hat{s}\hat{d}]}^{\beta}} \right) + \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \bar{Q}_{[\hat{s}\hat{d}]}^{\alpha\beta} * \left(- \frac{\partial^2 J^{\beta}}{\partial \phi_{o[sd]}^{\alpha}\, \partial \psi_{[\hat{s}\hat{d}]}^{\beta}} \right);
\end{aligned}$$

$$\begin{aligned}
l_{\pi[sd]}^{\alpha,\text{Stack}} \triangleq{}& \frac{\partial J^{\alpha}}{\partial \phi_{\pi[sd]}^{\alpha}} + \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} Q_{o[\hat{s}\hat{d}]}^{\alpha\beta} * \frac{\partial^2 J^{\beta}}{\partial \phi_{\pi[sd]}^{\alpha}\, \partial \phi_{o[\hat{s}\hat{d}]}^{\beta}} * \phi_{o[\hat{s}\hat{d}]}^{\beta}\\
&+ \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \sum_{p[\hat{s}\hat{d}] \in \boldsymbol{\Pi}_{[\hat{s}\hat{d}]}^{\beta}} Q_{p[\hat{s}\hat{d}]}^{\alpha\beta} * \frac{\partial^2 J^{\beta}}{\partial \phi_{\pi[sd]}^{\alpha}\, \partial \phi_{p[\hat{s}\hat{d}]}^{\beta}} * \phi_{p[\hat{s}\hat{d}]}^{\beta}\\
&+ \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} Q_{[\hat{s}\hat{d}]}^{\alpha\beta} * \frac{\partial^2 J^{\beta}}{\partial \phi_{\pi[sd]}^{\alpha}\, \partial \psi_{[\hat{s}\hat{d}]}^{\beta}} * \psi_{[\hat{s}\hat{d}]}^{\beta} + \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \bar{Q}_{o[\hat{s}\hat{d}]}^{\alpha\beta} * \left(- \frac{\partial^2 J^{\beta}}{\partial \phi_{\pi[sd]}^{\alpha}\, \partial \phi_{o[\hat{s}\hat{d}]}^{\beta}} \right)\\
&+ \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \sum_{p[\hat{s}\hat{d}] \in \boldsymbol{\Pi}_{[\hat{s}\hat{d}]}^{\beta}} \bar{Q}_{p[\hat{s}\hat{d}]}^{\alpha\beta} * \left(- \frac{\partial^2 J^{\beta}}{\partial \phi_{\pi[sd]}^{\alpha}\, \partial \phi_{p[\hat{s}\hat{d}]}^{\beta}} \right) + \sum_{[\hat{s}\hat{d}] \in \boldsymbol{SD}^{\beta}} \bar{Q}_{[\hat{s}\hat{d}]}^{\alpha\beta} * \left(- \frac{\partial^2 J^{\beta}}{\partial \phi_{\pi[sd]}^{\alpha}\, \partial \psi_{[\hat{s}\hat{d}]}^{\beta}} \right);
\end{aligned}$$

$$
\begin{aligned}
l^{\alpha,\mathrm{Stack}}_{[sd]} \triangleq{}& \frac{\partial J^{\alpha}}{\partial \psi^{\alpha}_{[sd]}} + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \psi^{\alpha}_{[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}} * \phi^{\beta}_{o[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} Q^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \psi^{\alpha}_{[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}} * \phi^{\beta}_{p[\hat{s}\hat{d}]} + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \psi^{\alpha}_{[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}} * \psi^{\beta}_{[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \psi^{\alpha}_{[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}}\right) + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} \bar{Q}^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \psi^{\alpha}_{[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}}\right) \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \psi^{\alpha}_{[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}}\right);
\end{aligned}
$$

$$
\begin{aligned}
l^{\beta,\mathrm{Stack}}_{o[sd]} \triangleq{}& \frac{\partial J^{\alpha}}{\partial \phi^{\beta}_{o[sd]}} + Q^{\alpha\beta}_{o[sd]} * \left[\frac{\partial J^{\beta}}{\partial \phi^{\beta}_{o[sd]}} - q^{\beta}_{[sd]}\right] + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{o[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}} * \phi^{\beta}_{o[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} Q^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{o[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}} * \phi^{\beta}_{p[\hat{s}\hat{d}]} + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{o[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}} * \psi^{\beta}_{[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{o[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}}\right) + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} \bar{Q}^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{o[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}}\right) \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{o[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}}\right);
\end{aligned}
$$

$$
\begin{aligned}
l^{\beta,\mathrm{Stack}}_{\pi[sd]} \triangleq{}& \frac{\partial J^{\alpha}}{\partial \phi^{\beta}_{\pi[sd]}} + Q^{\alpha\beta}_{\pi[sd]} * \left[\frac{\partial J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}} - q^{\beta}_{[sd]}\right] + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}} * \phi^{\beta}_{o[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} Q^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}} * \phi^{\beta}_{p[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}} * \psi^{\beta}_{[\hat{s}\hat{d}]} + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}}\right) \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} \bar{Q}^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}}\right) + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}}\right);
\end{aligned}
$$

$$
\begin{aligned}
l^{\beta,\mathrm{Stack}}_{[sd]} \triangleq{}& \frac{\partial J^{\alpha}}{\partial \psi^{\beta}_{[sd]}} + Q^{\alpha\beta}_{[sd]} * \left[\frac{\partial J^{\beta}}{\partial \psi^{\beta}_{[sd]}} - q^{\beta}_{[s.]}\right] + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \psi^{\beta}_{[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}} * \phi^{\beta}_{o[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} Q^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \psi^{\beta}_{[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}} * \phi^{\beta}_{p[\hat{s}\hat{d}]} + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} Q^{\alpha\beta}_{[\hat{s}\hat{d}]} * \frac{\partial^2 J^{\beta}}{\partial \psi^{\beta}_{[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}} * \psi^{\beta}_{[\hat{s}\hat{d}]} \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{o[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \psi^{\beta}_{[sd]}\,\partial \phi^{\beta}_{o[\hat{s}\hat{d}]}}\right) + \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \sum_{p[\hat{s}\hat{d}]\in \Pi^{\beta}_{[\hat{s}\hat{d}]}} \bar{Q}^{\alpha\beta}_{p[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \psi^{\beta}_{[sd]}\,\partial \phi^{\beta}_{p[\hat{s}\hat{d}]}}\right) \\
&+ \sum_{[\hat{s}\hat{d}]\in SD^{\beta}} \bar{Q}^{\alpha\beta}_{[\hat{s}\hat{d}]} * \left(-\frac{\partial^2 J^{\beta}}{\partial \psi^{\beta}_{[sd]}\,\partial \psi^{\beta}_{[\hat{s}\hat{d}]}}\right);
\end{aligned}
$$

where $Q^{\alpha\beta}_{o[sd]}$, $Q^{\alpha\beta}_{\pi[sd]}$, $Q^{\alpha\beta}_{[sd]}$, $\bar{Q}^{\alpha\beta}_{o[sd]}$, $\bar{Q}^{\alpha\beta}_{\pi[sd]}$, $\bar{Q}^{\alpha\beta}_{[sd]}$, $q^{\beta}_{[sd]}$, and $q^{\beta}_{[s.]}$ are Lagrange multipliers (see the Appendix).

Class c jobs arriving at a source node $[s.]$ will be rejected if the length $l^{c,\text{Stack}}_{o[sd]}$ is minimum, otherwise they will be routed along the minimum length path $l^{c,\text{Stack}}_{\pi[sd]}$, destined for the minimum length destination $l^{c,\text{Stack}}_{[sd]}$.

Next, let us define the vector of the congestion control, routing and load-sharing fractions as well as Lagrange multipliers:

$$
\begin{aligned}
\mathbf{Z} \stackrel{\Delta}{=} [&\dots \phi^{\alpha}_{o[sd]} \dots \phi^{\alpha}_{\pi[sd]} \cdots \psi^{\alpha}_{[sd]} \dots \\
&\dots \phi^{\beta}_{o[sd]} \dots \phi^{\beta}_{\pi[sd]} \dots \psi^{\beta}_{[sd]} \dots \\
&\dots q^{\beta}_{[sd]} \dots \\
&\dots q^{\beta}_{[s.]} \dots \\
&\dots Q^{\alpha}_{[sd]} \dots Q^{\alpha}_{[s.]} \dots \\
&\dots Q^{\beta}_{[sd]} \dots Q^{\beta}_{[s.]} \dots \\
&\dots \phi^{\beta}_{o[sd]} \dots \phi^{\beta}_{\pi[sd]} \dots \psi^{\beta}_{[sd]} \dots \\
&\dots \bar{Q}^{\alpha\beta}_{o[sd]} \dots \bar{Q}^{\alpha\beta}_{\pi[sd]} \dots \bar{Q}^{\alpha\beta}_{[sd]} \dots]^{\mathrm{T}},
\end{aligned}
$$

where $Q^{\alpha}_{[sd]}$, $Q^{\alpha}_{[s.]}$, $Q^{\beta}_{[sd]}$, and $Q^{\beta}_{[s.]}$ are Lagrange multipliers (see the Appendix).

Finally, define the vector

$$
\begin{aligned}
\nabla L(\mathbf{Z}) \stackrel{\Delta}{=} \Big[&\dots (l^{\alpha,\text{Stack}}_{o[sd]} - Q^{\alpha}_{[sd]}) \dots (l^{\alpha,\text{Stack}}_{\pi[sd]} - Q^{\alpha}_{[sd]}) \dots (l^{\alpha,\text{Stack}}_{[sd]} - Q^{\alpha}_{[s.]}) \dots \\
&\dots (l^{\beta,\text{Stack}}_{o[sd]} - Q^{\beta}_{[sd]}) \dots (l^{\beta,\text{Stack}}_{\pi[sd]} - Q^{\beta}_{[sd]}) \dots (l^{\beta,\text{Stack}}_{[sd]} - Q^{\beta}_{[s.]}) \dots \\
&\dots \Big(Q^{\alpha\beta}_{o[sd]} * \phi^{\beta}_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}} Q^{\alpha\beta}_{\pi[sd]} * \phi^{\beta}_{\pi[sd]} \Big) - \Big(\bar{Q}^{\alpha\beta}_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}} \bar{Q}^{\alpha\beta}_{\pi[sd]} \Big) \dots \\
&\dots \sum_{[.d] \in \boldsymbol{D}^{\beta}_{[s.]}} (Q^{\alpha\beta}_{[sd]} * \psi^{\beta}_{[sd]} - \bar{Q}^{\alpha\beta}_{[sd]}) \dots \\
&\dots \Big(1 - \phi^{\alpha}_{o[sd]} - \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\alpha}_{[sd]}} \phi^{\alpha}_{\pi[sd]} \Big) \dots \Big(1 - \sum_{[.d] \in \boldsymbol{D}^{\alpha}_{[s.]}} \psi^{\alpha}_{[sd]} \Big) \dots \\
&\dots \Big(1 - \phi^{\beta}_{o[sd]} - \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}} \phi^{\beta}_{\pi[sd]} \Big) \dots \Big(1 - \sum_{[.d] \in \boldsymbol{D}^{\beta}_{[s.]}} \psi^{\beta}_{[sd]} \Big) \dots \\
&\dots \Big(\frac{\partial J^{\beta}}{\partial \phi^{\beta}_{o[sd]}} - q^{\beta}_{[sd]} \Big) \dots \Big(\frac{\partial J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}} - q^{\beta}_{[sd]} \Big) \dots \Big(\frac{\partial J^{\beta}}{\partial \psi^{\beta}_{[sd]}} - q^{\beta}_{[s.]} \Big) \dots \\
&\dots \Big(\frac{\partial J^{\beta}}{\partial \phi^{\beta}_{o[sd]}} - q^{\beta}_{[sd]} \Big) \dots \Big(\frac{\partial J^{\beta}}{\partial \phi^{\beta}_{\pi[sd]}} - q^{\beta}_{[sd]} \Big) \dots \Big(\frac{\partial J^{\beta}}{\partial \psi^{\beta}_{[sd]}} - q^{\beta}_{[s.]} \Big) \dots \Big].
\end{aligned}
$$

First, I formulate the hierarchical resource-sharing problem as a non-linear complementarity problem (NCP).

Theorem NCP. $(\boldsymbol{\Phi}^*, \boldsymbol{\Psi}^*) \in (\boldsymbol{RC}, \boldsymbol{LS})$ *is a Stackelberg equilibrium solution for the two-hierarchical class joint load-sharing, routing and congestion control problem if and only if it solves the following Non-linear Complementarity Problem (NCP):*

$$\nabla L(\mathbf{Z}^*) * \mathbf{Z}^* = 0\,,$$
$$\nabla L(\mathbf{Z}^*) \geqslant 0\,,$$
$$\mathbf{Z}^* \geqslant \mathbf{0}\,.$$

Proof. See the Appendix. □

Next, I formulate the joint problem as a variational inequality problem (VIP).

Theorem VIP. $(\boldsymbol{\Phi}^*, \boldsymbol{\Psi}^*) \in (\boldsymbol{RC}, \boldsymbol{LS})$ *is a Stackelberg equilibrium solution for the two-hierarchical class joint load-sharing, routing and congestion control problem if and only if it solves the following Variational Inequality Problem (VIP):*

$$\nabla L(\mathbf{Z}^*) * (\mathbf{Z} - \mathbf{Z}^*) \geqslant 0\,, \quad \forall \mathbf{Z} > \mathbf{0}\,.$$

Proof. Karamardian [34] shows that the NCP: $f(x^*) * x^* = 0$; $f(x^*) \geqslant 0$; $x^* > 0$ and the VIP: find x^* such that $f(x^*) * (x - x^*) \geqslant 0$, $\forall x > 0$, are equivalent. □

In this section I have expressed the hierarchical resource-sharing problem as a NCP and a VIP.

6. Conclusions

The usual approach to the solution of resource-sharing problems in distributed systems and computer networks is the optimization of a single objective function as seen by the system administrator.

However, different classes of jobs usually not only have different objectives but also different access rights in using the system resources. A class of jobs may have higher priority in using the system resources than another class of jobs. So, it is not meaningful to optimize a single global objective function for all classes of jobs.

In this paper, assuming such hierarchical classes of jobs, I propose a new methodology for the joint load-sharing, routing and congestion control problem. I define a hierarchical resource-sharing problem based on Stackelberg game theory. Furthermore, I provide two alternative formulations (non-linear complementarity problem and variational inequality problem formulations) for solving this hierarchical resource-sharing problem. The proposed methodology may be extended to multiple hierarchical levels, where at each level there will be multiple cooperative or competitive classes.

Appendix

Proof. Using the Karush–Kuhn–Tucker conditions for the less powerful class β minimization problem, we can express the hierarchical resource sharing problem as follows:

minimize $J^\alpha(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)$

with respect to $(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta, \boldsymbol{q}^\beta)$

such that
$$\left[\frac{\partial J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)}{\partial \phi^\beta_{o[sd]}} - q^\beta_{[sd]}\right] * \phi^\beta_{o[sd]} = 0, \quad \forall [sd] \in \boldsymbol{SD}^\beta, \tag{1}$$

$$\left[\frac{\partial J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)}{\partial \phi^\beta_{\pi[sd]}} - q^\beta_{[sd]}\right] * \phi^\beta_{\pi[sd]} = 0, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^\beta_{[sd]}, [sd] \in \boldsymbol{SD}^\beta, \tag{2}$$

$$\left[\frac{\partial J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)}{\partial \psi^\beta_{[sd]}} - q^\beta_{[s.]}\right] * \psi^\beta_{[sd]} = 0, \quad \forall [.d] \in \boldsymbol{D}^\beta_{[s.]}, [s.] \in \boldsymbol{S}^\beta, \tag{3}$$

$$\frac{\partial J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)}{\partial \phi^\beta_{o[sd]}} - q^\beta_{[sd]} \geqslant 0, \quad \forall [sd] \in \boldsymbol{SD}^\beta, \tag{4}$$

$$\frac{\partial J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)}{\partial \phi^\beta_{\pi[sd]}} - q^\beta_{[sd]} \geqslant 0, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^\beta_{[sd]}, [sd] \in \boldsymbol{SD}^\beta, \tag{5}$$

$$\frac{\partial J^\beta(\boldsymbol{\Phi}^\alpha, \boldsymbol{\Psi}^\alpha, \boldsymbol{\Phi}^\beta, \boldsymbol{\Psi}^\beta)}{\partial \psi^\beta_{[sd]}} - q^\beta_{[s.]} \geqslant 0, \quad \forall [.d] \in \boldsymbol{D}^\beta_{[s.]}, [s.] \in \boldsymbol{S}^\beta, \tag{6}$$

$$\phi^\alpha_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^\alpha_{[sd]}} \phi^\alpha_{\pi[sd]} = 1, \quad \forall [sd] \in \boldsymbol{SD}^\alpha \tag{7}$$

$$\sum_{[.d] \in \boldsymbol{D}^\alpha_{[s.]}} \psi^\alpha_{[sd]} = 1, \quad \forall [s.] \in \boldsymbol{S}^\alpha, \tag{8}$$

$$\phi^\alpha_{o[sd]}, \phi^\alpha_{\pi[sd]} \geqslant 0, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^\alpha_{[sd]}, [sd] \in \boldsymbol{SD}^\alpha,$$

$$\psi^\alpha_{[sd]} \geqslant 0, \quad \forall [.d] \in \boldsymbol{D}^\alpha_{[s.]}, [s.] \in \boldsymbol{S}^\alpha,$$

$$\phi^\beta_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^\beta_{[sd]}} \phi^\beta_{\pi[sd]} = 1, \quad \forall [sd] \in \boldsymbol{SD}^\beta, \tag{9}$$

$$\sum_{[.d] \in \boldsymbol{D}^\beta_{[s.]}} \psi^\beta_{[sd]} = 1, \quad \forall [s.] \in \boldsymbol{S}^\beta, \tag{10}$$

$$\phi^\beta_{o[sd]}, \phi^\beta_{\pi[sd]} \geqslant 0, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^\beta_{[sd]}, [sd] \in \boldsymbol{SD}^\beta,$$

$$\psi^\beta_{[sd]} \geqslant 0, \quad \forall [.d] \in \boldsymbol{D}^\beta_{[s.]}, [s.] \in \boldsymbol{S}^\beta,$$

where $\boldsymbol{q}^\beta = [\ldots q^\beta_{[sd]} \ldots q^\beta_{[s.]} \ldots]$ are the Lagrange multipliers associated with the class β constraint sets $\boldsymbol{RC}^\beta$ and $\boldsymbol{LS}^\beta$.

In the sequence, we associate the Lagrange multipliers $Q^{\alpha\beta}_{o[sd]}$, $Q^{\alpha\beta}_{\pi[sd]}$, and $Q^{\alpha\beta}_{[sd]}$, with the equality constraints (1), (2), and (3); the $\bar{Q}^{\alpha\beta}_{o[sd]}$, $\bar{Q}^{\alpha\beta}_{\pi[sd]}$, and $\bar{Q}^{\alpha\beta}_{[sd]}$ with the inequality constraints (4), (5), and (6); the $Q^\alpha_{[sd]}$, and $Q^\alpha_{[s.]}$ with the class α constraints (7) and (8); and the $Q^\beta_{[sd]}$ and $Q^\beta_{[s.]}$ with the class β constraints (9) and (10).

Evaluating the Karush–Kuhn–Tucker conditions for the most powerful class α minimization problem, we have the following conditions:

$$[l^{\alpha*,\text{Stack}}_{o[sd]} - Q^{\alpha}_{[sd]}] * \phi^{\alpha*}_{o[sd]} = 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$[l^{\alpha*,\text{Stack}}_{\pi[sd]} - Q^{\alpha}_{[sd]}] * \phi^{\alpha*}_{\pi[sd]} = 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\alpha}_{[sd]}\,, \quad \forall [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$[l^{\alpha*,\text{Stack}}_{[sd]} - Q^{\alpha}_{[s.]}] * \psi^{\alpha*}_{[sd]} = 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$[l^{\beta*,\text{Stack}}_{o[sd]} - Q^{\beta}_{[sd]}] * \phi^{\beta*}_{o[sd]} = 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$[l^{\beta*,\text{Stack}}_{\pi[sd]} - Q^{\beta}_{[sd]}] * \phi^{\beta*}_{\pi[sd]} = 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$[l^{\beta*,\text{Stack}}_{[sd]} - Q^{\beta}_{[s.]}] * \psi^{\beta*}_{[sd]} = 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$[l^{\alpha*,\text{Stack}}_{o[sd]} - Q^{\alpha}_{[sd]} \geqslant 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$[l^{\alpha*,\text{Stack}}_{\pi[sd]} - Q^{\alpha}_{[sd]} \geqslant 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\alpha}_{[sd]}, \quad \forall [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$[l^{\alpha*,\text{Stack}}_{[sd]} - Q^{\alpha}_{[s.]} \geqslant 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$[l^{\beta*,\text{Stack}}_{o[sd]} - Q^{\beta}_{[sd]} \geqslant 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$[l^{\beta*,\text{Stack}}_{\pi[sd]} - Q^{\beta}_{[sd]} \geqslant 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$[l^{\beta*,\text{Stack}}_{[sd]} - Q^{\beta}_{[s.]} \geqslant 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$Q^{\alpha\beta}_{o[sd]} * \phi^{\beta*}_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}} Q^{\alpha\beta}_{\pi[sd]} * \phi^{\beta*}_{\pi[sd]} = \bar{Q}^{\alpha\beta}_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}} \bar{Q}^{\alpha\beta}_{\pi[sd]}\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\sum_{[.d] \in \boldsymbol{D}^{\beta}_{[s.]}} Q^{\alpha\beta}_{[sd]} * \psi^{\beta*}_{[sd]} = \sum_{[.d] \in \boldsymbol{D}^{\beta}_{[s.]}} \bar{Q}^{\alpha\beta}_{[sd]}\,, \quad \forall [s.] \in \boldsymbol{S}^{\beta}\,,$$

$$\phi^{\alpha*}_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\alpha}_{[sd]}} \phi^{\alpha*}_{\pi[sd]} = 1\,, \quad \forall [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$\sum_{[.d] \in \boldsymbol{D}^{c}_{[s.]}} \psi^{\alpha*}_{[sd]} = 1\,, \quad \forall [s.] \in \boldsymbol{S}^{\alpha}\,,$$

$$\phi^{\alpha*}_{o[sd]}, \phi^{\alpha*}_{\pi[sd]} \geqslant 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\alpha}_{[sd]},\ [sd] \in \boldsymbol{SD}^{\alpha}\,,$$

$$\psi^{\alpha*}_{[sd]} \geqslant 0, \quad \forall [.d] \in \boldsymbol{D}^{\alpha}_{[s.]},\ [s.] \in \boldsymbol{S}^{\alpha}\,,$$

$$\phi^{\beta*}_{o[sd]} + \sum_{\pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}} \phi^{\beta*}_{\pi[sd]} = 1\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\sum_{[.d] \in \boldsymbol{D}^{c}_{[s.]}} \psi^{\beta*}_{[sd]} = 1\,, \quad \forall [s.] \in \boldsymbol{S}^{\beta}\,,$$

$$\phi^{\beta*}_{o[sd]}, \phi^{\beta*}_{\pi[sd]} \geqslant 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]},\ [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\psi^{\beta*}_{[sd]} \geqslant 0, \quad \forall [.d] \in \boldsymbol{D}^{\beta}_{[s.]},\ [s.] \in \boldsymbol{S}^{\beta}\,,$$

$$\left[\frac{\partial J^{\beta*}}{\partial \phi^{\beta}_{o[sd]}} - q^{\beta}_{[sd]}\right] * \phi^{\beta*}_{o[sd]} = 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\left[\frac{\partial J^{\beta*}}{\partial \phi^{\beta}_{\pi[sd]}} - q^{\beta}_{[sd]}\right] * \phi^{\beta*}_{\pi[sd]} = 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}, [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\left[\frac{\partial J^{\beta *}}{\partial \psi^{\beta}_{[sd]}} - q^{\beta}_{[s.]}\right] * \psi^{\beta *}_{[sd]} = 0\,, \quad \forall [.d] \in \boldsymbol{D}^{\beta}_{[s.]}, [s.] \in \boldsymbol{S}^{\beta}\,,$$

$$\frac{\partial J^{\beta *}}{\partial \phi^{\beta}_{o[sd]}} - q^{\beta}_{[sd]} \geqslant 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\frac{\partial J^{\beta *}}{\partial \phi^{\beta}_{\pi[sd]}} - q^{\beta}_{[sd]} \geqslant 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}, [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\frac{\partial J^{\beta *}}{\partial \psi^{\beta}_{[sd]}} - q^{\beta}_{[s.]} \geqslant 0\,, \quad \forall [.d] \in \boldsymbol{D}^{\beta}_{[s.]}, [s.] \in \boldsymbol{S}^{\beta}$$

$$\bar{Q}^{\alpha\beta}_{o[sd]} * \left[q^{\beta}_{[sd]} - \frac{\partial J^{\beta *}}{\partial \phi^{\beta}_{o[sd]}}\right] = 0\,, \quad \bar{Q}^{\alpha\beta}_{o[sd]} \geqslant 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\bar{Q}^{\alpha\beta}_{\pi[sd]} * \left[q^{\beta}_{[sd]} - \frac{\partial J^{\beta *}}{\partial \phi^{\beta}_{\pi[sd]}}\right] = 0\,, \quad \bar{Q}^{\alpha\beta}_{\pi[sd]} \geqslant 0\,, \quad \forall \pi[sd] \in \boldsymbol{\Pi}^{\beta}_{[sd]}, [sd] \in \boldsymbol{SD}^{\beta}\,,$$

$$\bar{Q}^{\alpha\beta}_{[sd]} * \left[q^{\beta}_{[s.]} - \frac{\partial J^{\beta *}}{\partial \psi^{\beta}_{[sd]}}\right] = 0\,, \quad \bar{Q}^{\alpha\beta}_{[sd]} \geqslant 0\,, \quad \forall [sd] \in \boldsymbol{SD}^{\beta}\,.$$

Expressing the above conditions in vector form, we have the result. □

Acknowledgments

The author wishes to express his sincere thanks to the anonymous reviewers for their constructive suggestions.

References

[1] T. Basar and G.J. Olsder, Dynamic Noncooperative Game Theory (Academic Press, 1982).

[2] D.P. Bertsekas, Algorithms for nonlinear multicommodity network flow problems, Proc. Int. Symposium on Systems Analysis and Optimization (Springer-Verlag, 1979).

[3] D.P. Bertsekas, A class of optimal routing algorithms for communication networks, Proc. 5th Int. Conference on Computer Communications (1980) pp. 71–75.

[4] D.P. Bertsekas, Optimal routing and flow control methods for communication networks, Proc. 5th International Conference on Analysis and Optimization of Systems (1982) 615–643.

[5] D.P. Bertsekas and E.M. Gafni, Projected Newton methods and optimization of multicommodity flows, IEEE Trans. Automat. Control AC-28 (12) (1983) 1090–1096.

[6] D.P. Bertsekas, E.M. Gafni and R.G. Gallager, Second derivative algorithms for minimum delay distributed routing in networks, IEEE Trans. Commun. COM-32 (8) (1984) 911–919.

[7] K. Bharath-Kumar, Optimum end-to-end flow control in networks, Proc. Int. Communications Conference (IEEE, 1980) pp. 23.3.1–23.3.6.

[8] K. Bharath-Kumar and J.M. Jaffe, A new approach to performance-oriented flow control, IEEE Trans. Commun. COM-29 (4) (1981) 427–435.

[9] W.F. Bialas and M.H. Karwan, Multilevel optimization: a mathematical programming perspective, Proc. Conference on Decision and Control (IEEE, 1980), pp. 761–765.

[10] W.F. Bialas and M.H. Karwan, On two-level optimization, IEEE Trans. Automat. Control AC-27 (1) (1982) 211–214.

[11] J.W. Blankenship and J.E. Falk, Infinitely constrained optimization problems, J. Optimization Theory Appl. 19 (2) (1976) 261–281.
[12] J. Bracken and J.T. McGill, Mathematical programs with optimization problems in the constraints, Op. Res. 21 (1) (1973) 37–44.
[13] J.P. Buzen and P.P.S. Chen, Optimal load balancing in memory hierarchies, Information Processing 74 (North-Holland, 1974) pp. 271–275.
[14] D.G. Cantor and M. Gerla, Optimal routing in a packet-switched computer network, IEEE Trans. Computers C-23 (10) (1974) 1062–1069.
[15] D.A. Castanon and N.R. Sandel, Jr., On hierarchical Stackelberg optimization problems, Proc. Conference on Decision and Control (IEEE, 1977) 104–107.
[16] T.-S. Chang and P.B. Luh, Derivation of necessary and sufficient conditions for single-stage Stackelberg games via the inducible region concept, IEEE Trans. Automat. Control AC-29 (1) (1984) 63–66.
[17] R.W. Cottle, F. Giannesi and J.-L. Lions, Variational Inequalities and Complementarity Problems: Theory and Applications (Wiley, 1980).
[18] S.C. Dafermos and F.T. Sparrow, The traffic assignment problem for a general network, J. Res. National Bureau of Standards B, Math. Sci. 73B (2) (1969) 91–118.
[19] E. de Souza e Silva and M. Gerla, Load balancing in distributed systems with multiple classes and site constraints, in: E. Gelenbe (Ed.), Performance '84 (North Holland, 1984) pp. 17–33.
[20] A.A. Economides, A unified game-theoretic methodology for the joint load sharing, routing and congestion control problem, Ph.D. Dissertation, University of Southern California (1990).
[21] A.A. Economides and J.A. Silvester, A game theory approach to cooperative and non-cooperative routing problems, Proc. IEEE International Telecommunications Symposium, Brazil (1990) pp. 597–601.
[22] A.A. Economides and J.A. Silvester, Multi-objective routing in integrated services networks: a game theory approach, Proc. of IEEE Infocom 91 Conference (IEEE, 1991) pp. 1220–1227.
[23] A.A. Economides and J.A. Silvester, Optimal routing in a network with unreliable links, Proc. of IEEE Computer Networking Symposium (IEEE, 1988) 288–297.
[24] A.A. Economides and J.A. Silvester, Priority load sharing: an approach using Stackelberg games, 28th Annual Allerton Conference on Communications, Control, and Computing, 1990, also as USC Technical Report CENG 89-39 (1989).
[25] A.A. Economides and J.A. Silvester, Routing games, USC Technical Report CENG 89-38 (1989).
[26] L. Fratta, M. Gerla and L. Kleinrock, The flow deviation method: An approach to store-and-forward communication network design, Networks 3 (1973) 97–133.
[27] R.G. Gallager, Distributed network optimization algorithms, Proc. International Communications Conference (IEEE, 1979) pp. 43.2.1–43.2.2.
[28] R.G. Gallager, A minimum delay routing using distributed computation, IEEE Trans. Commun. COM-25 (1) (1977).
[29] R.G. Gallager and S.J. Golestaani, Flow control and routing algorithms for data networks, Proc. International Computer Communications Conference (1980) pp. 779–784.
[30] R. Glowinski, J.T. Lions and R. Tremolieres, Numerical Analysis of Variational Inequalities (North-Holland, 1981).
[31] E.L. Hahne and R.G. Gallager, Round robin scheduling for fair flow control in data communication networks, Proc. International Communications Conference (IEEE, 1986) pp. 103–107.
[32] I. Hlavacek, J. Haslinger, J. Necas and J. Lovisek, Solution of Variational Inequalities in Mechanics (Springer-Verlag, 1988).
[33] J.M. Jaffe, A decentralized 'optimal', multiple-use, flow control algorithm, Proc. International Computer Communications Conference (1980) pp. 839–843.
[34] S. Karamardian, Generalized complementarity problem, J. Optimization Theory Appl. 8 (3) (1971) 161–168.
[35] H. Kobayashi and M. Gerla, Optimal routing in closed queueing networks, ACM Trans. Computer Systems 1 (4) (1983) 294–310.
[36] J.F. Kurose and R. Simha, A microeconomic approach to optimal resource allocation in distributed computer systems, IEEE Trans. on Computers 38 (5) (1989) 705–717.
[37] J.F. Kurose and R. Simha, Second derivative algorithms for optimal resource allocation in distributed computer systems, Proc. Int. Conference on Distributed Computing Systems (IEEE, 1987) pp. 56–63.
[38] J.F. Kurose and S. Simha, A distributed algorithm for optimum static load balancing in distributed computer systems, Proc. Infocom 86 Conference (IEEE, 1986) pp. 458–467.
[39] H.-C. Lin, J.R. Yee and C.S. Raghavendra, Optimal joint load balancing and routing in message switched computer networks, Proc. Infocom 88 Conference (IEEE, 1988) pp. 274–281.
[40] P.B. Luh, S.-C. Chang and T.-S. Chang, Solutions and properties of multi-stage Stackelberg games, Automatica 20 (2) (1984) 251–256.
[41] L.M. Ni and K. Hwang, Optimal load balancing in a multiple processor system with many job classes, IEEE Trans. Software Eng. SE-11 (5) (1985) 491–496.
[42] L.M. Ni and K. Hwang, Optimal load balancing strategies for a multiple processor system, Proc. Int. Conference on Parallel Processing (IEEE, 1981) pp. 352–357.

[43] G.P. Papavassilopoulos, Algorithms for leader–follower games, Proc. 18th Annual Allerton Conference on Communication, Control, and Computing (1980) pp. 851–859.

[44] G.P. Papavassilopoulos, Algorithms for static Stackelberg games with linear costs and polyhedra constraints, Proc. Conference on Decisions and Control (IEEE, 1982) pp. 647–652.

[45] G.P. Papavassilopoulos, Solution of some stochastic quadratic Nash and leader–follower games, SIAM J. Control and Optimization 19 (4) Sept. (1981) 651–666.

[46] E. Polak and D.Q. Mayne, An algorithm for optimization problems with functional inequality constraints, IEEE Trans. Automat. Control AC-21 (2) April (1976) 184–193.

[47] M. Schwartz and C.K. Cheung, The gradient projection algorithm for multiple routing in message-switched networks, IEEE Trans. Commun. COM-26 (1976) 449–456.

[48] K. Shimizu and E. Aiyoshi, A new computational method for Stackelberg and min-max problems by use of a penalty method, IEEE Trans. Automat. Control AC-26 (2) April (1981) 460–466.

[49] M. Simaan, Stackelberg optimization of two-level systems, IEEE Trans. Systems, Man, and Cybernetics, July (1977) 554–557.

[50] R. Simha and J.F. Kurose, Stochastic approximation schemes for a load balancing problem, 27th Annual Allerton Conference on Communication, Control, and Computing (1989) 839–848.

[51] T.E. Stern, A class of decentralized routing algorithms using relaxation, Proc. National Telecommunications Conference (IEEE, 1976) pp. 42.1.1–42.1.5.

[52] A.N. Tantawi and D. Towsley, Optimal static load balancing in distributed computer systems, J. ACM 32 (2), April (1985), 445–465.

[53] J.G. Wardrop, Some theoretical aspects of road traffic research, Proc. Inst. Civil Eng., Part III, 1 (1952) 325–378.