

AN INTEGER-LINEAR ALGORITHM FOR OPTIMIZING ENERGY EFFICIENCY IN DATA CENTERS

Amin Ghodousian¹ and Maryam Raeisian Parvari²

¹ Department of Engineering Science, University of Tehran, Tehran, Iran

² Department of Algorithms and computation, University of Tehran, Tehran, Iran

ABSTRACT

Nowadays, to meet the enormous computational requests, energy consumption, the largest part which is related to idle resources, is strictly increased as a great part of a data center's budget. So, minimizing energy consumption is one of the most important issues in the field of green computing. In this paper, we present a mathematical model formed as integer-linear programming which minimizes energy consumption and maximizes user's satisfaction, simultaneously. However, migration variables, as principal decision variables of the model, can be relaxed to continuous activities in some practical problems. This constraint relaxation helps a decision maker to find faster solutions that are usually good approximations for optimum. Near feasible solutions (infeasible solutions that are desirably close to the feasible region) have been investigated as another relaxation considering the kind of solutions. For this purpose, we initially present a measure to evaluate the amount of infeasibility of solutions and then let the model consider an extended region including solutions with remissible infeasibility, if necessary.

KEYWORDS

Data center, Idle resources, Energy consumption, Migration, linear Optimization, Integer Programming.

1. INTRODUCTION

Nowadays, a wide range of demands and data have led to strictly increasing the number of data centers all around the world. Therefore, researchers's concern over the amount of energy consumption in a data center is quite understandable, since not only does it have a prominent impact on the budget of data centers [22], [3] but also leads to more and more emitted carbon dioxide (CO₂) outputs and other greenhouse gases (GHG). Reports show that these gases can be really hazardous for global warming [20]. As a result, energy saving has been one of the most important topics in recent years. To this end, the latest remarkable design for data centers is the vision of *energy proportionality* i.e., matching energy to the load [4], [8]. In this context, *Processor speed scaling* [1], [21] and *powering down nodes* [2] are two notable presented methods. The former is established dynamically to change the speed of processors to the workload, whereas the latter reduces energy consumption by managing the number of available nodes. In this paper, we pursue the second strategy and try to manage simultaneous energy consumption and also user's satisfaction as another economical parameter.

Energy consumption has been studied by many researches with emphasis on finding and deleting idle resources. According to previously obtained results [6], these resources consume energy as much as their active counterparts. Arguably, an effective use of active resources and powering

down unused ones can reduce costs significantly. Shutting down *idle resources* as one of the energy inefficiency factors in data centers [10] can reduce PUE (Power Usage Effectiveness) close to the standard value. As reported by Barroso and Holzle [4], Chen et al. [6], Fan et al. [7], Lefugery et al. [13] a great amount of energy, more than 60% of peak power, wastes when processors are idle. Therefore, eliminating these servers can be helpful to reduce base power [17]. However, there are two challenges here. The first is how to determine idle periods of resources and how to evaluate if it is worth paying switching cost or not. And the second is how to control servers; because, the server which is put to sleep may soon need to wake again [15]. Besides, it is also critical to respond to requests without sacrificing the service level agreements (SLAs), e.g., throughput, deadline and etc.

Meisner et al. [16] proposed the PowerNap approach in which there is a shift between high performance active state and low powernap state. They had a scrupulous look at optimizing energy efficiency during nap and minimizing transition time between states. Another method was surveyed by Le'on and Navarro [14]. They investigated the limits of energy savings and tried to specify the optimal set of active resources based on the Stackelberg game concept [9]. Their optimization problem is divided into two levels; infrastructure operator as a leader to specify which servers should be on or off and users as followers to select among active servers.

Virtualization is another impressive technique applied to data centers to reduce energy consumption and increase server efficiency [22], [18], [19]. This method enables multiple servers to be co-located and run concurrently on a single physical machine. Actually, it reduces capital costs by turning off extra servers and increases efficiency by enabling more work from less hardware (about 20% of energy consumption in a data center [22]).

The approach of this paper is based on an optimization model formed as integer-linear programming. By this formulation, the model finds the best migrations to optimize the distribution of requests satisfying some structural restrictions such as: the capacity bounds of servers and the least number of servers that should be on. The objective of the problem is twofold: the maximization of user satisfaction and the minimization of energy consumption consists of three factors: setup fee, the amount of responding energy consumed by servers and migration costs.

Although some difficulties arise computationally from integer variables, the linear structure of the model enables us to apply well-known methods in the field of integer programming such as interior point algorithms or powerful combined approaches such as simplex implementation together with a special cutting plane technique [5].

In many situations, irrational values for migrations are either theoretically meaningful (for example, by considering them with a continuous measure such as time, energy and so on instead of their number) or practically adjustable (like cases in which a number of discontinuous objects such as humans are reported in terms of percentage). Hence as another approach, one can relax the main variables of the model, migration variables, to continuous ones and obtain approximating solutions. These approximations are usually good solutions attained such faster than the case in which all variables are integer. Moreover, all or some variables of such solutions can often easily be rounded to desirable integer ones based on the judgment and the experience of a decision maker, if necessary.

Making a few modifications, we present a generalized model to include near feasible solutions as well. Actually, a solution with remissible infeasibility and adequately favorable objective function value (even probably better than feasible optimum) may play an impressive role in making policy decisions, especially when there are no feasible solutions for a problem. For this

purpose, we introduce a measure to evaluate the amount of solutions infeasibilities and attempt to find the best solution with minimum constraints in violation.

In section (2), we present the definition of our problem and some notations which are used in the entire paper. Section (3) includes the main formulation of the problem. The formulation of the user's satisfaction factor and its linearization have been studied in sections (4) and (5), respectively. In section (6), we introduce a measure to evaluate the infeasibility of solutions and investigate near feasible solutions. Finally, numerical examples are provided in section (7) for further clarification.

2. PROBLEM DEFINITION

As we know, servers/resources in a data center/system are assembled in racks and racks are placed in chassis.

Without loss of generality, suppose a system that consists of n users sending requests to m resources (on or off). By this assumption, we introduce some parameters and variables that will be used throughout the paper.

a_{ij} : The amount of requests from the i^{th} user to the j^{th} resource. So, clearly $Y_k^0 = \sum_{i=1}^n a_{ik}$ denotes total requests sent to the k^{th} resource.

M_i & m_i : The upper and the lower bound on the capacity of i^{th} resource, respectively.

In many situations, when received requests to some resources are lower than a certain amount, it is not advantageous to turn it on (or to keep it on). Here such economical and somewhat structural reasons in a system necessitate that the lower bounds m_i ($i = 1, 2, \dots, m$) as well as upper bounds are also considered in a problem.

q_i : A binary variable that represents whether the i^{th} resource is on or off ($q_i = 0$ when server is off and $q_i = 1$ when it is on).

ρ_i : Setup fee (the cost of turning on the i^{th} resource).

E_i : The amount of energy consumed by the i^{th} resource (after being turned on) to respond to each request.

Sometimes, the initial distribution of requests is impossible to be responded by resources (i.e., the amount of sent requests exceed the lower or/and the upper bounds of some resources). On the other hand, a feasible initial distribution is usually nonoptimal. Precisely, the operation of responses to requests, in their initial distribution, is often very costly and/or energy consuming. Total energy consumed by a data center is principally related to the differences of energy consumption rates in resources and also the number of resources that are inessentially or uneconomically turned on. Furthermore, a feasible strategy minimizing the total energy consumption of a data center may not be a desirable policy in view of user's satisfaction. Therefore, in order to find a feasible optimal distribution (a distribution satisfying capacity bounds of servers with minimum cost and maximum satisfaction for users) targeted in this paper, we need, almost always, to refer to some or all of the initial requests of a resource to other resources. This reassignment of requests is said to be **migration**.

The reasons of migration can be divided into three categories:

- I. $Y_i^0 > M$, i.e., the amount of initial requests sent to the i^{th} resource exceeds the capacity of the upper bound. In this case, either some sent requests to the i^{th} resource should be migrated to others or the i^{th} resource itself should be turned off. Similarly, when $Y_i^0 < m_i$, either some sent requests to other resources are referred to resource i or resource i is turned off. In the two cases above, a turned off resource can be equivalently interpreted as a resource all of whose requests are migrated.
- II. $Y_i^0 > 0, q_i = 0$, i.e. there exist some initial requests to a resource, which is not awake. In this case, either we turn this resource on (with paying setup fee, ρ_i or refer all its received requests, Y_i^0 , to other resources.
- III. $q_i = 1$ and $m_i \leq Y_i^0 \leq M_i$, but E_i is high. In this case, some or all of initial requests are migrated to other resources.

In order to calculate the costs paid by translating requests between servers, let x_{ij} be total migrations from resource i to resource j (clearly, $x_{ij} \geq 0$) and c_{ij} , be the cost per migration among them. So, $c_{ij}x_{ij}$ shows the total cost of migrations from resource i to j . In every data center, migration cost between two resources in the same rack is less than this cost between resources in different racks (see Figure 1).

Figure 1 shows a data center consisting of two racks A and B and five resources. So as noted, we have $c_{23} \leq c_{14}$, for instance.

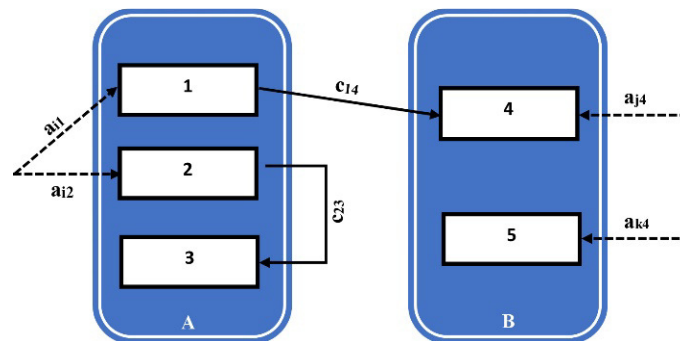


Figure 1. A data center with 2 racks and 5 servers. Migration cost c_{23} between servers 2 and 3 in common rack A is less than or equal to c_{14} between servers 1 and 4 placed in different racks.

Regarding the mentioned topics, our problem is formed to find an optimal migration in which:

- A1. The total setup cost of resources is minimized.
- A2. The total amount of energy consumed by resources is minimized.
- A3. The total cost of migrations is minimized.
- A4. Users' satisfaction is maximized.
- A5. If it has been decided to turn/keep the i^{th} resource off, then it must have no request (all initial sent requests to this resource must be migrated to others and there must exist no migrated requests from others to this resource).
- A6. If it has been decided to turn/keep the i^{th} resource on, then the amount of requests of this resource (after migrations) must belong to the interval $[m_i, M_i]$.

- A7. At least r resources are on (Systems are always prevented from violating the service level agreements by keeping some prespecified number of servers awake).

3. PROBLEM FORMULATION

In this section, we give a formulation for the problem above that satisfies conditions A1-A3 and A5-A7. We postpone the mathematical interpretation for user's satisfaction until section 4. Later, in section 5, the formulation of condition A4 is presented as linear constraints.

As already mentioned, the amount of initial requests to the i^{th} resource, Y_i^0 may not be feasible (cases I and II) or probably not optimal (case III). Therefore, the operation of migration adjusts the initial requests by decreasing or increasing the value of Y_i^0 's. To evaluate the amount of these reassigned requests that should be responded to later, we define new variables Y_i for each server i ($i = 1, 2, \dots, m$) which are as follows:

$$Y_i = Y_i^0 - \sum_{\substack{j=1 \\ j \neq i}}^m x_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} \quad (1)$$

Based on the previous notations, Y_i^0 and $\sum_{\substack{j=1 \\ j \neq i}}^m x_{ij}$ show the total amount of initial requests sent to the i^{th} resource and the total migrated requests from the i^{th} resource to the others, respectively. Similarly, $\sum_{\substack{j=1 \\ j \neq i}}^m x_{ji}$ is the total referred requests to the i^{th} resource. Thus, Y_i indicates the remaining requests in the i^{th} resource after all migrations. Variables Y_i have useful interpretations, especially in the next section where we utilize them to define a measure for user's satisfaction. Actually, we can consider variable Y_j as the active capacity of the i^{th} resource in contrast to its admissible capacity. Admissible capacity is the real or the structural capacity of a server that is initially known and bounded by m_i and M_i ($i = 1, \dots, m$). Hence, this type of capacity must not be necessarily used exhaustively, and is clearly independent of the amount of requests. On the contrary, active capacities are not constant, but they are variables whose values are subsequently determined by migration operation. By (1), the active capacity of a server is equal to the sum of requests reassigned by migrations to that server and shows the used amount of the server's capacity. Clearly, we have $m_i \leq Y_i \leq M_i, i = 1, \dots, m$.

Objective function:

Based on definition Y_i in (1), i^{th} resource consumes Y_i units of energy to respond to Y_i requests. Then, the total consumed energy in resources is equal to $\sum_{i=1}^m E_i Y_i$. Since $q_i \in \{0, 1\}$ ($\forall i \in \{1, \dots, m\}$), the total cost of setup fee is attained by $\sum_{i=1}^m \rho_i q_i$. Moreover, the total cost of migrations is also calculated as $\sum_{i \neq j} c_{ij} x_{ij}$.

Therefore, the objective function satisfying conditions A1-A3 can be written in the following way:

$$\min \quad \sum_{i=1}^m \rho_i q_i + \sum_{i=1}^m E_i Y_i + \sum_{i \neq j} \sum c_{ij} x_{ij}$$

Constraints:

Consider the following inequalities:

$$m_i q_i \leq Y_i \leq M_i q_i \quad , i = 1, \dots, m \quad (2)$$

$$0 \leq \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} \leq \left(\sum_{i=1}^m Y_i^0 \right) \cdot q_i \quad , i = 1, \dots, m \quad (3)$$

$$\sum_{i=1}^m q_i \geq r \quad (4)$$

We will show that relations (1), (2), (3) and (4) satisfy conditions A5, A6 and A7. At first, we present a lemma.

Lemma 3.1.

Suppose that the i^{th} resource is off ($q_i = 0$). Then, condition A5 is satisfied iff equations (5) and (6) are satisfied.

$$\sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} = 0 \quad (5)$$

$$Y_i^0 = \sum_{\substack{j=1 \\ j \neq i}}^m x_{ij} \quad (6)$$

Proof: Equation (5) means that there exists no migrated request from other resources to i and equation (6) shows that the total initial requests to i^{th} resource, Y_i^0 , is equal to the total migrated requests received. In the other words, all initial requests to this resource migrate to others. Hence, if $q_i = 0$, then equations (5) and (6) are satisfied iff the i^{th} resource has no requests iff A5 is satisfied. □

Theorem 3.2. Relations (2)-(4), satisfy conditions A5-A7.

Proof. Let $q_i = 0$. Then, we have by (2) and (3), respectively.

$$Y_i = 0 \quad (1^*)$$

and

$$\sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} = 0 \quad (2^*)$$

From (1*) and (1)

$$Y_i^0 - \sum_{\substack{j=1 \\ j \neq i}}^m x_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} = 0 \quad (3^*)$$

Then, it is attained by (2*) and (3*),

$$Y_i^0 = \sum_{\substack{j=1 \\ j \neq i}}^m x_{ij} \quad (4^*)$$

Relations (2*), (4*) and lemma 3.1 imply condition A5. In the other case, let $q_i = 1$. Then, relation (3) is converted to obvious inequalities, $0 \leq \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} \leq \sum_{i=1}^m Y_i^0$ (clearly, the amount of requests referred to i^{th} resource, $0 \leq \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji}$, is non-negative and less than the total initial requests, $\sum_{i=1}^m Y_i^0$). In this case, from (2), we have $m_i \leq Y_i \leq M_i$ that implies condition A6. However, inequality (4) is clearly equivalent to condition A7. \square Finally, the problem is then formulated in the following way:

$$\begin{aligned} \min \quad & \sum_{i=1}^m \rho_i q_i + \sum_{i=1}^m E_i Y_i + \sum_{i \neq j} \sum c_{ij} x_{ij} \\ & Y_i = Y_i^0 - \sum_{\substack{j=1 \\ j \neq i}}^m x_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} \\ & m_i q_i \leq Y_i \leq M_i q_i \quad , \quad i = 1, \dots, m \\ & 0 \leq \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} \leq \left(\sum_{k=1}^m Y_k^0 \right) \cdot q_i \quad , \quad i = 1, \dots, m \\ & \sum_{i=1}^m q_i \geq r \\ & x_{ij} \in \mathbb{N} \cup \{0\} \quad , \quad i, j = 1, \dots, m \quad , \quad i \neq j \\ & q_i \in \{0,1\} \quad , \quad i = 1, \dots, m \end{aligned} \quad (7)$$

Where r and $\sum_{k=1}^m Y_k^0 = \sum_{k=1}^m \sum_{i=1}^n a_{ik}$ are scalars and \mathbb{N} is the set of natural numbers.

Definition 3.3.

For brevity, let X be the feasible set of problem (7). Equivalently, X is the set of feasible

points (q, Y, x) in (7) where $q = (q_1, \dots, q_m)$, $Y = (Y_1, \dots, Y_m)$ and $x = (x_{11}, \dots, x_{ij}, \dots, x_{m(m-1)})$ in which $i \neq j$. Also, let $f(q, Y, x) = \sum_{i=1}^m \rho_i q_i + \sum_{i=1}^m E_i Y_i + \sum_{i \neq j} \sum c_{ij} x_{ij}$ be the value of objective function at (q, Y, x) .

4. A FORMULATION FOR USER'S SATISFACTION

According to the mentioned notations, Y_i is the active capacity of the j^{th} resource (the amount of requests remaining in resource j after all migrations) and $a_{1j}, a_{2j}, \dots, a_{nj}$ represent the amount of the initial requests from n users to this resource. After the migrations, each server divides its active capacity among n users. Suppose that the contribution of the i^{th} user to the active capacity of the j^{th} resource is equal to Y_{ij} . We can also consider Y_{ij} as bandwidth dedicated by resource j to user i .

From this new notation, we have two cases:

- If $a_{ij} > Y_{ij}$, then resource j can only respond to Y_{ij} request of the i^{th} user ($a_{ij} - Y_{ij}$ amount of initial request from user i remains unanswered by resource j).
- If $a_{ij} \leq Y_{ij}$, then all initial requests from user i will be responded by resource j . Specially, if $a_{ij} \leq Y_{ij}$, then $Y_{ij} - a_{ij}$ amount of the active capacity of resource j is used for other users.

By considering the two above-mentioned cases, the amount of i^{th} user's requests that is responded by resource j is equal to $\min\{a_{ij}, Y_{ij}\}$. Thus, $\sum_{j=1}^m \min\{a_{ij}, Y_{ij}\}$ denotes the total amount of i^{th} user's requests responded by all resources. Clearly, the greater the term $\sum_{j=1}^m \min\{a_{ij}, Y_{ij}\}$ is, the more the amount of the i^{th} user's requests is responded to and then the greater response rate to the i^{th} user's satisfaction is increased. Hence, in order to maximize user's satisfaction, we maximize the following summation

$$\sum_{i=1}^n \sum_{j=1}^m \min\{a_{ij}, Y_{ij}\} \quad (8)$$

Furthermore, the sum of the contributions of users to the active capacity of resource j , $\sum_{i=1}^n Y_{ij}$, must obviously not exceed this capacity. In other words, we have

$$\begin{aligned} \sum_{i=1}^n Y_{ij} &\leq Y_j, & j = 1, 2, \dots, m \\ Y_{ij} &\geq 0, & j = 1, 2, \dots, m, \quad i = 1, 2, \dots, n \end{aligned} \quad (9)$$

By multiplying (8) by -1 and adding it to the objective function of problem (7), the new objective function would be:

$$\begin{aligned} g(q, Y, y, x) &= \sum_{i=1}^m \rho_i q_i + \sum_{i=1}^m E_i Y_i + \sum_{i \neq j} \sum c_{ij} x_{ij} \\ &- \sum_{i=1}^n \sum_{j=1}^m \min\{a_{ij}, Y_{ij}\} = f(q, Y, x) - \sum_{i=1}^n \sum_{j=1}^m \min\{a_{ij}, Y_{ij}\} \end{aligned} \quad (10)$$

By replacing the objective function of problem (7) with (10) and adding (9) to the constraints set of the problem, we find the improved problem (11) in which the satisfaction of users is also included.

$$\min \quad g(q, Y, y, x) = f(q, Y, x) - \sum_{i=1}^n \sum_{j=1}^m \min\{a_{ij}, Y_{ij}\} \quad (11)$$

Subject to : X
 $\sum_{i=1}^n Y_{ij} \leq Y_j, \quad j = 1, \dots, m$
 $Y_{ij} \geq 0, \quad i = 1, \dots, n, \quad j = 1, \dots, m$

Where X is as described in definition 3.3.

However, term $\sum_{i=1}^n \sum_{j=1}^m \min\{a_{ij}, Y_{ij}\}$ in the objective makes the problem non-linear.

Definition 4.1.

Let X_g be the feasible set of problem (11). Equivalently, X_g is the set of feasible points (q, Y, y, x) in (11) where $y = (Y_{11}, Y_{12}, \dots, Y_{nm})$ and q, Y and x are as described in definition 3.3.

Remark 4.2.

Since X_g is the union of X and constraints (9), it is clear that if $(q, Y, y, x) \in X_g$ then $(q, Y, x) \in X$. On the other hand, X is formed as the intersection of constraints all of which are independent of variables $Y_{ij}, i = 1, \dots, n, j = 1, \dots, m$.

Hence, if $(q, Y, x) \in X$ then, we also have $(q, Y, y, x) \in X$ for each $y = (Y_{11}, \dots, Y_{nm})$.

5. LINEARIZATION OF PROBLEM (11)

In order to linearize objective (10), firstly we set:

$$t_{ij} \leq a_{ij}, Y_{ij} \quad , \quad i = 1, \dots, n \quad , \quad j = 1, \dots, m \tag{12}$$

By adding (12) as new constraints to problem (11) and replacing $\sum_{i=1}^n \sum_{j=1}^m \min\{a_{ij}, Y_{ij}\}$ with $\sum_{i=1}^n \sum_{j=1}^m t_{ij}$ in the objective, we have the following integer-linear problem

$$\begin{aligned} \min \quad & f(q, Y, x) - \sum_{i=1}^n \sum_{j=1}^m t_{ij} \\ \text{Subject to :} \quad & X \\ & \sum_{i=1}^n Y_{ij} \leq Y_j \quad , \quad j = 1, \dots, m \\ & Y_{ij} \geq 0 \quad , \quad i = 1, \dots, n \quad , \quad j = 1, \dots, m \\ & t_{ij} \leq a_{ij}, Y_{ij} \quad , \quad i = 1, \dots, n \quad , \quad j = 1, \dots, m \end{aligned} \tag{13}$$

Where X is as described in definition 3.3.

We will slightly later (in theorem 5.5) prove that the optimum of problem (13) is correspondent to an optimum for problem (11) with the same objective function values.

Definition 5.1.

Let X_h be the feasible set of problem (13). Similar to definitions 3.3 and 4.1, X_h is the set of feasible points (q, Y, y, x, t) in (13) where $t = (t_{11}, t_{12}, \dots, t_{nm})$ and q, Y, y and x are as in definitions 3.3 and 4.1. Also, let $h(q, Y, y, x, t) = f(q, Y, x) - \sum_{i=1}^n \sum_{j=1}^m t_{ij}$ be the value of objective function in problem (13) at (q, Y, y, x, t) .

Remark 5.2.

Similar to remark 4.2, since X_h is the union of X_g and constraints (12), $(q, Y, y, x, t) \in X_h$ implies $(q, Y, y, x) \in X_g$. Also, since X_g is resulted from the intersection of constraints all of which are independent of variables $t_{ij}, i = 1, \dots, n, j = 1, \dots, m$, then $(q, Y, y, x) \in X_g$ implies $(q, Y, y, x, t) \in X_g$ for each $t = (t_{11}, \dots, t_{nm})$.

Lemma 5.3 presents conditions by which each feasible solution in problem (11) or (13) has a correspondent feasible solution in another problem with the same objective function values.

Lemma 5.3.

(a) Suppose that $(q, Y, y, x, t) \in X_h$ such that $t_{ij} = \min\{a_{ij}, Y_{ij}\}$ for $i = 1, \dots, n$ and $j = 1, \dots, m$. Then, $(q, Y, y, x) \in X_g$ and $h(q, Y, y, x, t) = g(q, Y, y, x)$.

(b) Suppose that $(q, Y, y, x) \in X_g$. If we define $t_{ij} = \min\{a_{ij}, Y_{ij}\}$ for $i = 1, \dots, n, j = 1, \dots, m$, then $(q, Y, y, x, t) \in X_h$ and $h(q, Y, y, x, t) = g(q, Y, y, x)$

Proof. (a) Let $(q, Y, y, x, t) \in X_h$. From remark 5.2, $(q, Y, y, x) \in X_g$ also, since $t_{ij} = \min\{a_{ij}, Y_{ij}\}$ for $i = 1, \dots, n, j = 1, \dots, m$, we have

$$h(q, Y, y, x, t) = f(q, Y, x) - \sum_{i=1}^n \sum_{j=1}^m t_{ij} = f(q, Y, x) - \sum_{i=1}^n \sum_{j=1}^m \min\{a_{ij}, Y_{ij}\} = g(q, Y, y, x)$$

(b) Let $(q, Y, y, x) \in X_g$. Remark 5.2 results in, $(q, Y, y, x, t) \in X_g$ for each variable t . Hence, (q, Y, y, x, t) satisfies (9). By combining remarks 4.2 and 5.2, we have $(q, Y, y, x, t) \in X$ for each variable y and t . Therefore, (q, Y, y, x, t) belongs to X and satisfies (9). Furthermore, assuming, $t_{ij} = \min\{a_{ij}, Y_{ij}\}, i = 1, \dots, n, j = 1, \dots, m$, then, (q, Y, y, x, t) also satisfies (12). Thus we have, $(q, Y, y, x, t) \in X_h$.

The equivalence of the objective values is attained similar to the proof of part (a).□

Lemma 5.4.

Suppose that $(q^*, Y^*, y^*, x^*, t^*) \in X_h$ is an optimal solution of (13). Then, $t_{ij}^* = \min\{a_{ij}, Y_{ij}^*\}$ for each $i = 1, \dots, n$ and $j = 1, \dots, m$.

Proof.

On the contrary, suppose that $t_{rs}^* < \min\{a_{rs}, Y_{rs}^*\}$ for some $r = 1, \dots, n$ and $s = 1, \dots, m$.

Set $t' = (t'_{11}, \dots, t'_{nm})$ where

$$t'_{ij} = \begin{cases} \min\{a_{rs}, Y_{rs}^*\} & i = r \text{ and } j = s \\ t_{ij} & \text{elsewhere} \end{cases}$$

Thus, we have $t_{ij} \leq t'_{ij}$ for each $i = 1, \dots, n$ and $j = 1, \dots, m$ and specially, $t_{rs}^* < t'_{rs}$. Therefore, considering definition 5.1, we have $(q^*, Y^*, y^*, x^*, t') \in X_h$ and $h(q^*, Y^*, y^*, x^*, t') < h(q^*, Y^*, y^*, x^*, t^*)$ that contradicts the optimality of $(q^*, Y^*, y^*, x^*, t^*)$.□

Theorem 5.5.

Let $(q^*, Y^*, y^*, x^*, t^*)$ be an optimal solution in (13). Then, (q^*, Y^*, y^*, x^*) is the optimum of problem (11) with the same objective function values.

Proof.

Let the assumption of the theorem be true. By lemma 5.4, $t_{ij}^* = \min\{a_{ij}, Y_{ij}^*\}$ for $i = 1, \dots, n, j = 1, \dots, m$. This fact together with lemma 5.3 (part (a)) implies $(q^*, Y^*, y^*, x^*) \in X_g$ and $h(q^*, Y^*, y^*, x^*, t^*) = g(q^*, Y^*, y^*, x^*)$. It remains to prove the optimality of (q^*, Y^*, y^*, x^*) in (11). By contradiction, suppose that (q^*, Y^*, y^*, x^*) is not

optimum. Thus, there exists $(\hat{q}, \hat{Y}, \hat{y}, \hat{x}) \in X_g$ such that $g(\hat{q}, \hat{Y}, \hat{y}, \hat{x}) < g(q^*, Y^*, y^*, x^*)$. Also, from lemma 5.3 (part (b)), let $(\hat{q}, \hat{Y}, \hat{y}, \hat{x}, \hat{t}) \in X_h$ be the correspondent solution to $(\hat{q}, \hat{Y}, \hat{y}, \hat{x})$ such that $h(\hat{q}, \hat{Y}, \hat{y}, \hat{x}, \hat{t}) = g(\hat{q}, \hat{Y}, \hat{y}, \hat{x})$. Therefore,
 $h(\hat{q}, \hat{Y}, \hat{y}, \hat{x}, \hat{t}) = g(\hat{q}, \hat{Y}, \hat{y}, \hat{x}) < g(q^*, Y^*, y^*, x^*) = h(q^*, Y^*, y^*, x^*, t^*)$
 that contradicts the optimality of $(q^*, Y^*, y^*, x^*, t^*)$. \square

6. INFEASIBILITY ANALYSIS

Obviously, when the sum of requests is greater than that of the capacities of resources, the feasible region of the problem is empty. Usually, the infeasibility of a problem hides some useful information playing often an impressive role in the selection of a strategy. For example:

- When problem (13) has no feasible solutions, we have no additional information but the infeasibility of our system, while it is often necessary to know the total number of requests (although not sufficient) a system can respond to.
- Critical situations e.g., Bankruptcy situations, in practice are usually correspondent to the case of infeasibility in a model. And, in these situations, it is more important to have a certain strategy more than a simple answer "the problem is infeasible".
- There may exist an infeasible solution with a remissible infeasibility (an infeasible solution being adequately near the feasible region) and a considerably desirable objective function value. A decision maker may condone these infeasibilities and find a good near feasible solution instead of stopping with no solutions (or optimal feasible ones with less desirability in objective function). Near feasible solutions have been frequently investigated by models permitting the constraints to be slightly perturbed or dislocated [11]. Interestingly, these near feasible points are practically more reasonable than feasible ones resulted from crisp constraints and their objective values may be even better than that of a feasible optimum [11].

In this section, we introduce some variables to measure the amount of the infeasibility of solutions. This measure provides us with information we need to analyze good near solutions of the problem. At first, we add a new resource as an artificial server to the problem. Assuming m resources in the system, all parameters and variables related to the artificial server are indexed by $m + 1$. q_{m+1} and $x_{i(m+1)}$ ($i = 1, 2, \dots, m$) denotes the status of the augmented resource (on or off) and the amount of migrated jobs from the i^{th} original resource to this resource, respectively. By considering this new server, relation (1) is modified for $i = 1, \dots, m$ in the following way:

$$Y_i = Y_i^0 - \sum_{\substack{j=1 \\ j \neq i}}^m x_{ij} + \sum_{\substack{j=1 \\ j \neq i}}^m x_{ji} - x_{i(m+1)} \quad (14)$$

For a fixed i , equality (14) represents, as was previously mentioned, total requests over the i^{th} resource after all migrations (probably, migrations to the artificial server). The value of $x_{i(m+1)}$ is interpreted as the amount of requests not being responded by resource i . We define,

$$Y_{m+1} = \sum_{i=1}^m x_{i(m+1)} \quad (15)$$

i.e., the total amount of requests that are not responded in our system. We apply variable Y_{m+1} to measure the amount of the infeasibility of solutions. Since, it is meaningless to consider any migration from the artificial server to others, we write only incoming requests to this server (artificial migration) in equation (15). Justifiably, no costs are associated to such artificial

migrations (because they are not real migrations). Moreover, we add a new constraint to problem (13) which is as follows:

$$0 \leq \sum_{i=1}^m x_{i(m+1)} \leq q_{m+1} \cdot \left(\sum_{k=1}^m Y_k^0 \right) \quad (16)$$

Equality $q_{m+1} = 0$ indicates that the artificial server is off and equivalently all requests are responded by original servers. In this case, (16) implies $\sum_{i=1}^m x_{i(m+1)} = 0$. The latter equality means no migrations are referred from the original servers to the artificial server that is, in turn, equivalent to $q_{m+1} = 0$. If $q_{m+1} = 1$, then (16) is considered as trivial inequalities

$$0 \leq \sum_{i=1}^m x_{i(m+1)} \leq \sum_{k=1}^m Y_k^0 \quad (17)$$

(the amount of requests not being responded to by the original servers is less than or equal to the total requests). In this case, these inequalities do not obviously impose any restriction on the feasible region of the problem.

Finally, in order to minimize the amount of non-responded requests in the system, we add term $P \cdot [q_{m+1} + \sum_{i=1}^m x_{i(m+1)}]$ to the objective function of problem (13) in which P is an adequately large penalty coefficient. These statements have been formally presented in lemma 6.1 and theorems 6.3 and 6.4.

Lemma 6.1.

Let $P = m \cdot \rho_{max} + \left(\sum_{k=1}^m Y_k^0 \right) \cdot (m \cdot E_{max} + m^2 \cdot c_{max} + mn)$ where $\rho_{max} = \max\{\rho_i : i = 1, \dots, m\}$, $E_{max} = \max\{E_i : i = 1, \dots, m\}$ and $c_{max} = \max\{c_{ij} : i, j = 1, \dots, m, i \neq j\}$. Then, $h(q, Y, y, x, t) \leq P$ for each $(q, Y, y, x, t) \in X_h$.

Proof. Since $q_i \in \{0, 1\}, i = 1, \dots, m$, then

$$\sum_{i=1}^m \rho_i q_i \leq m \cdot \rho_{max} \quad (1^*)$$

From $Y_i \leq \sum_{k=1}^m Y_k^0$ ($i = 1, \dots, m$), we have

$$\sum_{i=1}^m E_i Y_i \leq m \cdot E_{max} \cdot \sum_{k=1}^m Y_k^0 \quad (2^*)$$

Requests can be migrated from each original server to other $m - 1$ original servers and/or one artificial server. Also, there is no migration from the artificial server to others. Therefore, there exist, at most, m^2 migrations in a system. This fact together with $x_{ij} \leq \sum_{k=1}^m Y_k^0$ ($i, j = 1, \dots, m, i \neq j$), implies:

$$\sum_{i \neq j} \sum c_{ij} x_{ij} \leq m^2 \cdot c_{max} \cdot \sum_{k=1}^m Y_k^0 \quad (3^*)$$

At last, since $Y_{ij} \leq Y_j$ (from relation (9)) and $a_{ij}, Y_{ij} \leq \sum_{k=1}^m Y_k^0$, then

$$t_{ij} \leq a_{ij}, Y_{ij} \leq \sum_{k=1}^m Y_k^0 \quad (i = 1, \dots, n, j = 1, \dots, m)$$

Thus,

$$\sum_{i=1}^n \sum_{j=1}^m t_{ij} \leq nm \cdot \sum_{k=1}^m Y_k^0 \quad (4^*)$$

The result follows by summing (1*), (2*), (3*) and (4*). \square

Definition 6.2.

Let \bar{X} be a new feasible set attained from X_h by adding constraints (15) and (16) and replacing equalities (1) with modified version (14). Similar to definition 5.1, \bar{X} consists of feasible points (q, Y, y, x, t, x_{m+1}) where $q = (q_1, \dots, q_m, q_{m+1})$ and $x_{m+1} = (x_{1(m)}, \dots, x_{m(m+1)})$ and Y, y, x and t are the same as the ones presented in definition 5.1. Also, let P be defined as in lemma 6.1 and $\bar{f}(q, Y, y, x, t, x_{m+1}) = h(q, Y, y, x, t) + P \cdot [q_{m+1} + \sum_{i=1}^m x_{i(m+1)}]$ obtained by adding penalty term $P \cdot [q_{m+1} + \sum_{i=1}^m x_{i(m+1)}]$ to the objective of problem (13).

Theorem 6.3.

Suppose that $(q, Y, y, x, t, x_{m+1}) \in \bar{X}$ in which there is no migration to the artificial server and $(q', Y', y', x', t', x'_{m+1}) \in \bar{X}$ has at least one artificial migration. Then,

$$\bar{f}(q, Y, y, x, t, x_{m+1}) = \bar{f}(q', Y', y', x', t', x'_{m+1})$$

Proof.

Since (q, Y, y, x, t, x_{m+1}) has no artificial migration, then $x_{i(m+1)} = 0$ for each $i = 1, \dots, m$. Thus, $\bar{f}(q, Y, y, x, t, x_{m+1}) = h(q, Y, y, x, t) + P \cdot q_{m+1}$. On the other hand, in $(q', Y', y', x', t', x'_{m+1})$ there exists at least one migration to the artificial resource that implies $x'_{k(m+1)} > 0$ for some $k \in \{1, \dots, m\}$. Thus, $q'_{m+1} = 1$, from (16) and then

$$q'_{m+1} + \sum_{i=1}^m x'_{i(m+1)} \geq 2 \quad (*)$$

Hence, we have

$$\begin{aligned} \bar{f}(q, Y, y, x, t, x_{m+1}) &= h(q, Y, y, x, t) + P \cdot q_{m+1} && \text{(from lemma 6.1)} \\ &\leq P + P \cdot q_{m+1} && \text{(from (*))} \\ &\leq P \cdot [q'_{m+1} + \sum_{i=1}^m x'_{i(m+1)}] \\ &\leq P \cdot [q'_{m+1} + \sum_{i=1}^m x'_{i(m+1)}] + f(q', Y', x', t') \\ &= \bar{f}(q', Y', x', t', x'_{m+1}) \end{aligned}$$

In theorem 6.3, the penalty term enforces the model to find a feasible solution as optimum (if the feasible region is non-empty). Theorem 6.4, in the following part, shows that the inequality proved by theorem 6.3 is strict for an optimal feasible solution.

Theorem 6.4.

Suppose that optimal solution $(q^*, Y^*, y^*, x^*, t^*, x_{m+1}^*) \in \bar{X}$ with no artificial migration. If $(q', Y', y', x', t', x'_{m+1}) \in \bar{X}$ has at least one artificial migration, then

$$\bar{f}(q^*, Y^*, x^*, t^*, x_{m+1}^*) < \bar{f}(q', Y', y', x', t', x'_{m+1})$$

Proof.

Similar to the proof of theorem 6.3, we have $x_{i(m+1)}^* = 0, (i = 1, \dots, m)$ and

$$q'_{m+1} + \sum_{i=1}^m x'_{i(m+1)} \geq 2 \tag{*}$$

If $q_{m+1}^* = 1$, then we set $\hat{q} = (\hat{q}_1, \dots, \hat{q}_{m+1})$ in which

$$\hat{q}_i = \begin{cases} 0 & i = m + 1 \\ q_i^* & i \neq m + 1 \end{cases}$$

Thus, point $(\hat{q}, Y^*, y^*, x^*, t^*, x_{m+1}^*) \in \bar{X}$ and

$\bar{f}(\hat{q}, Y^*, y^*, x^*, t^*, x_{m+1}^*) < \bar{f}(q^*, Y^*, y^*, x^*, t^*, x_{m+1}^*)$ that contradicts the optimality of $(q^*, Y^*, y^*, x^*, t^*, x_{m+1}^*)$ Therefore, $q_{m+1}^* = 0$. Now, we have

$$\bar{f}(q^*, Y^*, y^*, x^*, t^*, x_{m+1}^*) = h(q^*, Y^*, y^*, x^*, t^*) \tag{from lemma 6.1}$$

$$\leq P \tag{from (*)}$$

$$\begin{aligned} &< P \cdot [q'_{m+1} + \sum_{i=1}^m x'_{i(m+1)}] \\ &\leq P \cdot [q'_{m+1} + \sum_{i=1}^m x'_{i(m+1)}] + h(q', Y', y', x', t') \\ &= \bar{f}(q', Y', y', x', t', x'_{m+1}) \quad \square \end{aligned}$$

Remark 6.5.

As was previously mentioned, by adding the penalty term to the objective function, the model insists to find a feasible solution as optimum. However, such optimums may consume a high amount of energy. In order to prevent these situations and equivalently find good near feasible solutions, we can consider an upper bound on the amount of total energy consumed by our system which is as follows:

$$\sum_{i=1}^m E_i Y_i \leq M$$

where M is a prespecified scalar. This restriction allows the model to neglect some requests and consequently to save the energy of the system.

7. NUMERICAL EXAMPLES

Figure 2 illustrates a system (data center) with two racks A and B consisting of 3 and 4 servers, respectively. The initial distribution of requests from 12 users to these 7 servers have been given by matrix $a = (a_{ij})_{12 \times 7}$ below

$$a = \begin{pmatrix} 7 & 0 & 3 & 0 & 30 & 12 & 30 \\ 14 & 11 & 0 & 21 & 0 & 11 & 0 \\ 0 & 0 & 1 & 3 & 12 & 0 & 0 \\ 9 & 0 & 0 & 7 & 8 & 2 & 25 \\ 2 & 12 & 0 & 10 & 12 & 1 & 9 \\ 60 & 0 & 0 & 0 & 0 & 0 & 32 \\ 10 & 11 & 3 & 20 & 0 & 4 & 0 \\ 0 & 26 & 0 & 21 & 4 & 26 & 10 \\ 9 & 0 & 0 & 0 & 7 & 0 & 0 \\ 0 & 0 & 0 & 18 & 0 & 20 & 8 \\ 0 & 0 & 0 & 0 & 5 & 12 & 10 \\ 1 & 12 & 1 & 10 & 0 & 0 & 0 \end{pmatrix}$$

Where a_{ij} represents the amount of requests from the i^{th} user to the j^{th} server.

For a fixed column j in a , the sum of components a_{ij} ($i = 1, \dots, 12$), $\sum_{i=1}^{12} a_{ij}$, says that all requests are sent to the j^{th} server. Total initial requests, $\sum_{k=1}^7 Y_k^0 = \sum_{k=1}^7 \sum_{i=1}^{12} a_{ik}$, are equal to 592 and the minimum admissible turned on servers, $\sum_{i=1}^7 q_i$, has been taken greater than or equal to 4. Capacity bounds m_i 's and M_i 's, migrations costs c_{ij} 's, energy costs E_i 's and setup fee ρ_i 's are assumed in the following way:

$$M_i = 100, \quad m_i = 10, \quad i = 1, 2, \dots, 7$$

$$c_{ij} = \begin{cases} 1 & \text{if servers } i \text{ and } j \in A \text{ or, } i \text{ and } j \in B \\ 2 & \text{elsewhere} \end{cases}$$

$$E_1 = E_3 = E_4 = E_6 = E_7 = 8, \quad E_2 = 5, \quad E_5 = 1$$

$$\rho_1 = \rho_2 = \rho_6 = \rho_7 = 2, \quad \rho_3 = \rho_4 = 5, \quad \rho_5 = 10$$

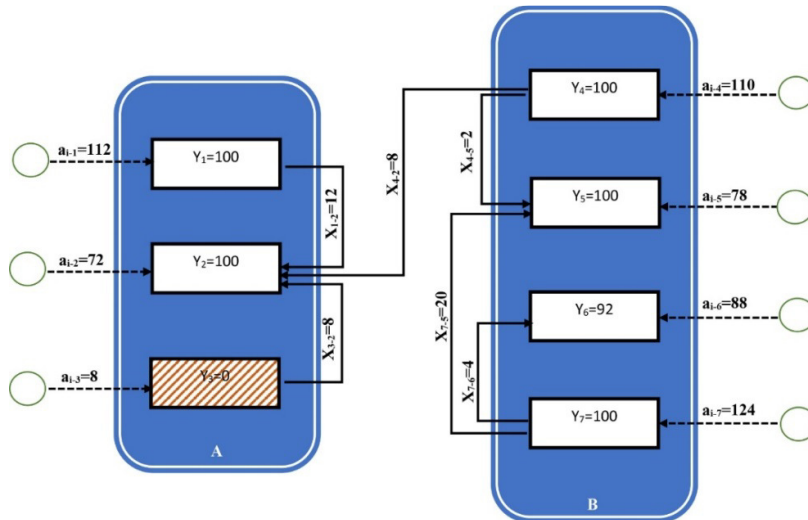


Figure 2. A data center with racks A and B and 7 resources. Shaded server 3 is off. Initial requests and migrations have been depicted by dotted and solid lines, respectively.

The minimum objective function value is 3283. The values of the optimal migrations, x_{ij} 's, have been written next to solid lines representing these migrations between the servers. The sum of requests sent to each server has been shown by dotted lines in Figure 2. For example, the initial request to server 4 is $Y_4^0 = \sum_{i=1}^{12} a_{i4} = a_{14} + a_{24} + a_{34} + a_{44} + a_{54} + a_{64} + a_{74} + a_{84} + a_{94} + a_{104} + a_{114} + a_{124} = 110.8$ and 2 units of these requests have been migrated to server 2 and 5, respectively. As is shown in the figure, the optimal request responded by this server is $Y_4 = 100$ (displayed in the center of the server). Also, of the optimal solution of problem (13), we have

$$t_{11} = 7 \quad t_{12} = 0 \quad t_{13} = 0 \quad t_{14} = 0 \quad t_{15} = 30 \quad t_{16} = 12 \quad t_{17} = 30$$

The maximum satisfaction of the first user is attained by:

$$\sum_{j=1}^7 t_{1j} = 79$$

Therefore, $\frac{\sum_{j=1}^7 t_{1j}}{\sum_{j=1}^7 a_{1j}} = 96.3\%$ of the first user's requests have been fulfilled.

By section 2, the system prefers expectedly to migrate jobs between servers in the same racks. According to the initial distribution of requests, resources 3 and 7 received initial requests out of the admissible range of their capacities. In the optimal solution of figure 2, all requests to server 3 are migrated and then, $q_3 = 0$ (server 3 becomes off). Also, some migrations have been referred to from server 7 to decrease the amount of its received requests to less than or equal to the upper bound of its capacity $100 = Y_7 \leq M_7 = 100$.

Finally, although server 5 has the highest setup fee, its low energy consumption causes the model to keep this server in the system.

Figure 3 illustrates a system with infeasible solutions set and total initial requests $\sum_{j=1}^7 Y_j^0 = 660$.

$$a = \begin{pmatrix} 5 & 20 & 20 & 0 & 11 & 12 & 30 \\ 10 & 0 & 0 & 21 & 0 & 11 & 11 \\ 0 & 2 & 2 & 31 & 10 & 26 & 0 \\ 20 & 0 & 0 & 7 & 8 & 0 & 25 \\ 12 & 0 & 0 & 0 & 12 & 1 & 9 \\ 0 & 5 & 5 & 8 & 0 & 0 & 8 \\ 5 & 3 & 3 & 0 & 15 & 4 & 7 \\ 26 & 10 & 1 & 0 & 4 & 26 & 12 \\ 0 & 2 & 2 & 21 & 0 & 0 & 2 \\ 12 & 0 & 0 & 13 & 5 & 10 & 0 \\ 0 & 13 & 1 & 0 & 0 & 12 & 10 \\ 12 & 21 & 21 & 3 & 0 & 0 & 12 \end{pmatrix}$$

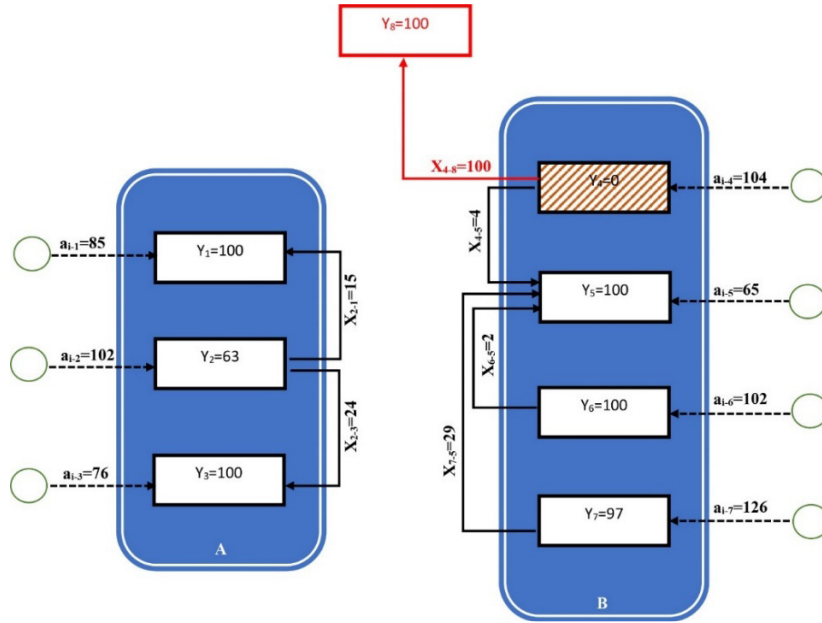


Figure 3. A data center with racks *A* and *B* and 7 resources. Shaded server 4 is off. Initial requests and migrations have been depicted by dotted and solid lines, respectively. Server 8 is the only artificial server in the system.

However, equality $Y_8 = 100$ (related to the artificial server) states that the system can still respond to $\sum_{k=1}^7 Y_k^0 - 100 = 560$ amount of requests. This amount of infeasibility (100 units) maybe excusable (compared with the total initial requests = 660). Therefore, the current solution can be considered as a good near feasible solution (note that in this example the feasible region is empty) with some amounts of satisfactions. For example, the satisfaction of the first user is obtained as $\sum_{j=1}^7 t_{1j} = 88$ where $t_{1j} (j = 1, \dots, 7)$ are resulted from the modified model presented by definition 6.2.

REFERENCES

- [1] M. Andrews, A.Fernandez Anta, L.Zhang and W. Zhao, (2013), Routing and scheduling for energy and delay minimization in the power down model, *Networks* 61(3), 226-237.
- [2] M. Andrews, A.Fernandez, L.Zhang and W. Zhao, (2010), Routing for energy minimization in the speed scaling model, *INFOCOM, Proceedings IEEE*, 1-9.
- [3] L.A. Barroso, (2005), The Price of Performance: An Economic Case for Chip Multiprocessing, *ACM Queue* 3 (7), 48- 53.
- [4] L.A. Barroso and U.Holzle, (2007), The case for energy-proportional computing, *IEEE* 40(12), 33-37.
- [5] J.M. S. Bazzara, J. J. Jarvis and H. D. Sherali, (2011), *Linear programming and network flows*.
- [6] G.chen, W.He, J.Liu, S.Nath, L.Rigas, L.Xiao and F.Zhao, (2008), Energy-aware server provisioning and load dispatching for connection-intensive internet services, *NSDI'08: 5th USENIX Symposium on Networked Systems Design and Implementation*, 337-350.
- [7] X. Fan, W.-D. Weber and L.A.Barroso, (2007), Power provisioning for a warehouse-sized computer, *In Proceedings of the 34th annual international symposium on Computer architecture: ACM* 35(2), 13-23.
- [8] M. Feldman, K. Lai and L. Zhang, (2009), The proportional share allocation market for computational resources, *IEEE transactions on parallel and Distributed System* 20(8), 1075-1088.
- [9] D. Fudenberg and J. Tirole, (1991), *Game Theory*, Cambridge, MA: MIT Press.
- [10] L. Ganesh, (2012), *Data center energy management*, Dissertation, Cornell University.

- [11] A.Ghodousian and E.khorram, (2008), Fuzzy linear optimization in the presence of the fuzzy relation inequality constraints with max-min composition, *Information Sciences* 178(2), 501-519.
- [12] Y.Jin, Y.Wen, Q.Chen and Z.Zhu, (2013), An Empirical Investigation of the Impact of Server Virtualization on Energy Efficiency for Green Data Center, *Computer Journal* 56(8), 977.
- [13] C.Lefugery, X. Wang and M. Ware, (2007), Server-level power control, *ICAC '07: Autonomic Computing*, 4.
- [14] X.Leon and L.Navarro, (2013), A Stackelberg game to drive the limits of energy savings for the allocation of data center resources, *Future Generation Computer Systems* 29(1), 74-83.
- [15] M.lin, A.wierman, Lachen.L.H.Andrew and E.Thereska, (2013), Dynamic rightsizing for power-proportional data centers, *IEEE/ACM Transactions on Networking* 21(5), 1378-1391.
- [16] D.Meisner, B.T.Gold and T.F.Wenisch, (2009), PowerNap: Eliminating Server Idle Power, *ACM SIGPLAN Notices - ASPLOS* 44(3), 205-216.
- [17] E.Pinherio, R.Bianchini, E.V.Carrera and T.Heath, (2003), compilers and Operating systems for low power, Springer, ch. Dynamic Cluster Reconfiguration for Power and Performance, 75-93.
- [18] R. Talaber, T. Brey and L. Lamers, (2009), Using virtualization to improve data center efficiency, The Green Grid, Tech. rep, White Paper.
- [19] M.Uddin, A.Rahman, (2011), Virtualization Implementation Model for Cost Effective and Efficient Data Centers, *International journal of Advanced Computer Science and Applications* 2(1).
- [20] Webb M, (2008), SMART 2020: Enabling the low carbon economy in the information age. Tech. rep, Climate Group on behalf of the Global Sustainability Initiative (GeSI).
- [21] A. Weirman, L. L. H. Andrew and A. Tang, (2009), Power-aware speed scaling in processor sharing systems, *INFOCOM 2009*, 2007-2015.
- [22] J.Hamilton, Cost of power in large-scale data centers, <http://perspectives.mvdirona.com/> (2008).