

BRAESS-LIKE PARADOXES FOR NON-COOPERATIVE DYNAMIC LOAD BALANCING IN DISTRIBUTED COMPUTER SYSTEMS

Said Fathy El-Zoghdy

Department of Computer Science, Faculty of Science, Menoufia University, Shebin El-Koom, Egypt.

Email: elzoghdy@yahoo.com

Abstract

We consider a distributed computer system in Wardrop equilibrium, i.e., situations where no user can reduce its own response time by unilaterally choosing another path, if all the other users retain their present paths. The Braess paradox is a famous example of paradoxical cases where adding capacity to a network degrades the performance of all users. This study examines numerically some examples around the Braess-like paradox in a distributed computer system. We found that Braess's paradox can occur, namely in equilibrium the mean job response time in the network after adding a communication line for the sharing of jobs between nodes, for some system parameter setting, can be greater than the mean job response time in the network before adding the communication line. Indeed, two different types of paradox called weak and strong paradox have been characterized. In the range of parameter values examined, the worst case ratio of performance degradation obtained in the examined network model is about 75% and 65% for the cases of weak and strong paradox respectively.

Keywords - Braess Paradox, Wardrop Equilibrium, Distributed Computer Systems, Load Balancing, Performance Evaluation, Non-cooperative Networks.

1. Introduction

The exponential growth of computer networking, in terms of number of users and components, traffic volume and diversity of service, demands massive upgrades of capacity in existing networks. Traditionally, capacity design methodologies have been developed with a single-class networking paradigm in mind. This approach overlooks the non-cooperative structure of modern (high speed and large-scale) networks and entails, as will be explained in the sequel, the danger of degraded performance when resources are added to a network. Indeed, load balancing decisions in large-scale computer and communication networks (e.g., Computing GRID, Internet) are often made by each user independently, according to its own individual performance objectives. Such networks are henceforth called *non-cooperative*, and game theory [7] provides the systematic framework to study and understand their behavior. Under the non-cooperative paradigm, the network is considered as a site of competition for the network resources among selfish users [13]–[21]. The most common example of a non-cooperative network is the Internet. In the current TCP (Transmission Control Protocol) flow control mechanism, each user adjusts its transmission window the maximum number of unacknowledged packets that the user can have circulating in the network independently, based on some feedback information about the level of congestion in the network (detected as packet loss). Moreover, the Internet Protocol (both IPv4 and the current IPv6 specifications) provides the option of source routing, that enables the user to determine the path(s) its flow follows from source to destination [13]–[18].

An important problem in current high-speed and large-scale computer and communication networks is to provide all users with satisfactory network performance. Intuitively, we can think that the total processing capacity of a system will increase when the capacity of a part of the system increases and so we expect improvements in performance objectives accordingly in that case. The famous Braess paradox tells us that this is not always the case; i.e., adding capacity to the system

may sometimes lead to the degradation in the benefits of all users in a Wardrop equilibrium [3, 4, 5, 8]. A Wardrop equilibrium is attained in the situation where each user chooses a path of the minimum cost, the choice of a single user has only a negligible impact on the cost of each path and the equilibrium cost of each used path is identical, which is not greater than the costs of unused paths [1].

The Braess paradox has been the subject of considerable research, see for example [5, 9, 10, 13, 15, 16] and the survey in [8]. It attracted the attention of researchers in many fields such as Arora and Sen [2] in the field of Software Multi-Agent Systems, Roughgarden and Tardos [18] in the Theory of Computing, Cohen and Kelly [5], and Cohen and Jeffries [4] in queuing networks, and El-Zoghdy *et al* [9, 10, 11] in distributed computational systems.

Cohen and Kelly [5] reported the first example of Braess's paradox in a mathematical model of a queuing networks. They investigated Braess's paradox in the setting where the users have knowledge only of mean queue lengths of the network servers that is, they used a static load balancing policy. Then they raised the question of whether the paradox also occurs in networks where users have information about the instantaneous queue lengths, not just mean queue lengths (i.e., each user upon arrival can see the number of waiting users in every server in the network). In a later paper, Kelly [6] gives an example showing that providing users with some additional information about the current system state may lead to system performance degradation. On the same model of Cohen and Kelly [5], Calvert, Solomon and Ziedins [3] considered the situation where users have full knowledge of all the instantaneous queue lengths of the network servers and they are able to make their load balancing decisions based on that knowledge (i.e., the load balancing policy is a dynamic one). They found that Braess's paradox can also occur in this setting as well.

For continuity with the work of Calvert, Solomon and Ziedins [3], this paper investigates the phenomenon of Braess's paradox under a dynamic individually optimal load balancing policy on the same computer network model. The measure of system performance degradation is the ratio of the difference between the mean response times of the network after (*augmented network*) and before (*initial network*) adding a communication line for the sharing of jobs between nodes over that of the initial network (i.e., if the measure is greater than zero, this mean that there is paradox in that case).

Our simulation results show that Braess's paradox can occur, namely in equilibrium the mean job response time in the augmented network, for some system parameter setting, can be greater than the mean job response time in the initial network. Indeed, two different types of paradox have been characterized. We called them *weak* and *strong* paradox (see section II). Based on the definitions of the weak and the strong paradox, we can say that what Calvert, Solomon and Ziedins [3] found is a weak paradox. In this paper, we present some examples of weak and strong paradox, and estimate the worst case ratio of the system performance degradation in the range of parameter values examined.

From the course of numerical experimentation, we found that the worst case ratio of performance degradation obtained in the examined computer network model is about 75(65)% for the case of weak (strong) paradox. Although our study was originally motivated by design problems in the field of computer networking, the results may be applied to other types of networks such as transportation networks, large-scale computer and communication networks (e.g., Internet, Computing GRID [22]).

The outline of the paper is as follows. Section 2 presents the description and the assumptions of the model studied in this paper. Section 3 presents the dynamic individually optimal load balancing policy. Section 4 describes the results of numerical examination. Finally, section 5 summarizes this paper.

2. System Model and Assumptions

As mentioned in section I, the model considered in this paper is studied by Calvert, Solomon and Ziedins [3]. First, we consider their initial network as illustrated in Fig. 1. This network consists of six nodes numbered 0,1,...,5. Node 0 is the entrance node, node 1 and node 4 are single server queues with first-come-first-served (FCFS) service discipline, node 2 and node 3 are infinite servers and node 5 is the exit node. Before adding capacity (a link), the network has two paths, $0 - 1 - 2 - 5$ (P1) and $0 - 3 - 4 - 5$ (P2) from entrance to exit. Each user individually chooses a path to minimize his mean response time from entrance to exit, given the choices of other users. Equilibrium is defined to occur when no user can lower his total mean response time by a change of path, if all the other users retain their present paths. Thus users in the network may be viewed as playing a non-cooperative game, each seeking to minimize its own mean response time from source to exit.

We assume that jobs arrive at the system according to a time-invariant Poisson process, i.e. inter-arrival times of jobs are independent, identically and exponentially distributed with mean $1/\lambda$. Also, we assume that users have service times that are exponentially distributed with mean $1/\mu_1$, $1/\mu_2$, $1/\mu_3$ and $1/\mu_4$ at nodes i , $i = 1, \dots, 4$ respectively. We further assume that service times are independent of each other and of arrival time. A job arriving at the initial network will choose at the decision point $pd1$ either to join node 1 (i.e., go through P1) or node 3 (i.e., go through P2) knowing the service rates μ_1, \dots, μ_4 and the system load information, i.e., n_1, n_2, n_3 and n_4 , where n_1, n_2, n_3 and n_4 are the numbers of jobs in the queues of the first, second, third and fourth servers respectively.

We compare the performance of initial network with that of the augmented network illustrated in Fig. 2. The augmented network differs from the initial one by the addition of a communication line between the two FCFS servers (node 1 and node 4). So, it has one more path from entrance to exit than the initial network. This path is $0 - 1 - 4 - 5$ (P3). In the augmented network, jobs can make load balancing decisions at two decision points namely $dp1$ and $dp2$. Upon arrival to the network, a job has to decide at $dp1$ either to join node 1 (i.e., go through P1) or node 3 (i.e., go through P2), and then if it chooses to join node 1, at the time it leaves node 1 it has to decide at $dp2$ either to join node 2 or node 4 (i.e., to continue using P1 or to use P3).

As mentioned earlier, the aim of this paper is to investigate the phenomenon of Braess's paradox on the considered computer network model in Wardrop equilibrium under a dynamic individually optimal load balancing policy. We focus our attention on the system performance degradation that may occur as a result of adding capacity (a communication line) for the sharing of jobs between nodes. To this aim we differentiate between two types of paradox, we call them *weak and strong* paradox which can be defined as follows:

Definition 1 (Paradox) We say that a Braess-like paradox occurs if condition 1 is satisfied. That is, if the overall mean response time of the augmented network is higher than that of the initial network for the same system parameter setting.

Definition 2 (Strong Paradox) We say that a strong-paradox occurs if conditions 1 and 2 are satisfied for the same system parameter setting. That is, plus condition 1, the minimum of mean response times offered by the three paths (P1, P2 and P3), from entrance to exit, in augmented network is greater than the maximum of the mean response times offered by the two paths (P1 and P2), from entrance to exit, in the initial network for the same system parameter setting. Which means that all the users of the network suffer from performance degradation as a result of adding a communication line for the sharing of jobs between nodes.

Definition 3 (Weak Paradox) We say that a weak-paradox occurs if conditions 1 and 3 are satisfied for the same system parameter setting. That is, plus condition 1, the minimum of mean response times offered by the three paths (P1, P2 and P3), from entrance to exit, in augmented network is less than or equal to the maximum of the mean response times offered by the two paths

(P1 and P2), from entrance to exit, in the initial network for the same system parameter setting. Which means that even though the overall response time of the augmented network is greater than that of the initial network, there exist some users of the augmented network whom do not suffer from degradation i.e., their mean response time in the augmented network is less than or equal to that of the initial network:

$$\frac{(T_a - T_i)}{T_i} > 0 \tag{1}$$

$$\frac{\min_{j=1,2,3} (TaP_j)}{\max_{i=1,2} (TiPi)} > 1 \tag{2}$$

$$\frac{\min_{j=1,2,3} (TaP_j)}{\max_{i=1,2} (TiPi)} \leq 1 \tag{3}$$

where,

- T_i : is the overall mean response of the initial network.
- T_a : is the overall mean response of the augmented network.
- $T_i P_i$: is the mean response of jobs that take the path number i , ($i = 1, 2$) in the initial network.
- $T_a P_j$: is the mean response of jobs that take the path number j , ($j = 1, 2, 3$) in the augmented network.

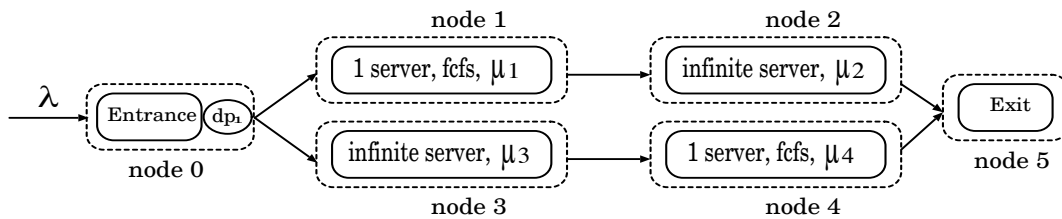


Fig 1. the initial network

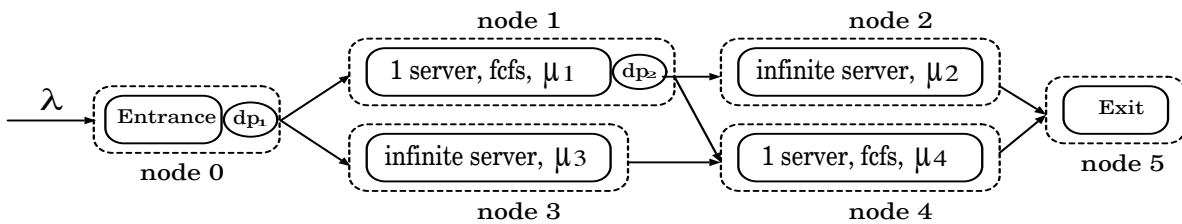


Fig. 2 the augmented network

3. Dynamic Individually Optimal Load Balancing Policy

As mentioned earlier, the load balancing policy used for the considered models is a dynamic individually optimal load balancing policy where every job strives to optimize (minimize) its mean response time independently (non-cooperatively) of the other jobs. According to this policy, jobs are scheduled so that every job may feel that its own expected response time is minimum if it knows the expected node delay at each node. In other words, when the individually optimal policy is realized, the expected response time of a job cannot be improved further when the load balancing

decisions for other jobs are fixed, and the system reaches an equilibrium [12]. It appears that this policy is closely related to a completely decentralized scheme in that each job itself determines on the basis of the system load information which node should process it. For the studied models, the load balancing decisions in this policy are based on a general decision procedure which can be formalized as follows:

For both initial and augmented networks, given that the number of jobs currently in the first and fourth servers are n_1 and n_4 respectively, an arriving job to the system (both initial and augmented networks) has to take a load balancing decision at dp_1 to choose between either joining node 1 (i.e., go through P1) or node 3 (i.e., go through P2), the expected mean response time via P1 is $((n_1+1)/\mu_1+1/\mu_2)$ and that via P2 is $(1/\mu_3+(n_4+1)/\mu_4)$, regardless of the decision procedure at dp_2 . In the event of a tie, we let the users to choose P1, thus users choose P1 iff $((n_1+1)/\mu_1+1/\mu_2) \leq (1/\mu_3+(n_4+1)/\mu_4)$.

In the augmented network there is a second load balancing decision to be made at dp_2 for the users going through P1 to choose between node 2 and node 4 (i.e., to continue using P1 or to use P3). Given that the number of jobs currently in the fourth server is n_4 , the expected mean response time to the exit from that point via P1 is $1/\mu_2$ and that via P3 is $(n_4+1)/\mu_4$, regardless of the decision procedure at dp_1 . In the event of a tie, we let the users to choose P1, thus users choose P1 iff $1/\mu_2 \leq (n_4+1)/\mu_4$.

As it could be seen from the previous explanation, the decision procedure used to make load balancing decisions is a simple and a general one in contrast to that presented by Calvert, Solomon and Ziedins [3]. As mentioned in section I, there exist some significant differences in between the load balancing decision procedure used in our simulator and that used by Calvert, Solomon and Ziedins [3]. These differences can be summarized as follows:

1. In their decision procedure, they supposed that the arrival stream of jobs at the entrance is finite but in ours, we consider it infinite, and
2. They also supposed that every arriving job at the entrance is aware of the number of users behind it but in ours, we ignored that.

For more information about the load balancing decision procedure used by Calvert, Solomon and Ziedins, the reader is referred to *theorem 2.1* in [3].

4. Results and Discussion

To investigate the phenomenon of Braess's paradox on the considered model in Wardrop equilibrium, a course of numerical experimentation has been done using the OMNet++ (*Objective Modular Network Tested*) discrete event simulation system. Through the course of numerical experimentation, we find some examples around the Braess's paradox on Cohen-Kelly computer network model in Wardrop equilibrium under a dynamic individually optimal load balancing policy. The mean response times T_i , T_a , T_{iP1} , T_{iP2} , T_{aP1} , T_{aP2} , and T_{aP3} of the initial, augmented, the two paths (P1 and P2) from entrance to exit in the initial and the three paths (P1, P2 and P3) from entrance to exit in the augmented network respectively have been estimated for various combinations of job arrival rate to the system λ , job processing rates μ_1 , μ_2 , μ_3 and μ_4 , at the first, second, third, and fourth servers respectively. In our simulator, a 95% confidence interval is used. As mentioned in section I, the measure of the system performance degradation is the ratio of the difference between the mean response times of the augmented and initial networks over that of the initial network, i.e., $(T_a - T_i)/T_i$. So, if condition 1 is satisfied this means that the system is

suffering from performance degradation (Braess-like paradox). In addition to condition 1, if condition 2(3) is satisfied, we say that a *strong (weak)* paradox occurs.

We started our numerical experimentation by doing the same experiment as that of Calvert, Solomon and Ziedins [3]. In this experiment, the service rate parameters are $\mu_1=\mu_4=2.5$, $\mu_2=\mu_3=0.5$ and the arrival rate λ varies from 0.05 to 5 (*note that an arrival rate of 5.0 is the upper bound of the capacity of the system in this setting*). Although there exist some significant differences in between the load balancing decision procedure used by Calvert, Solomon and Ziedins [3] and ours, we almost got the same result (from the overall point of view). From that experiment, we can conclude that the expected response times, given a finite arrival stream, approximate the expected response times with an infinite arrival stream. Fig 3 shows the overall system performance degradation ratio in that case. From this figure, we note that, for low arrival rates, the augmented network performs better than the initial network (*note that when $T_a < T_i$, we put the degradation ratio to zero which means no paradox is realized*), while the reverse is true for high arrival rates. Braess's paradox appears for $\lambda > 2.5$ and the worst case ratio of performance degradation is achieved for $\lambda = 4.25$ and is about 28.99%.

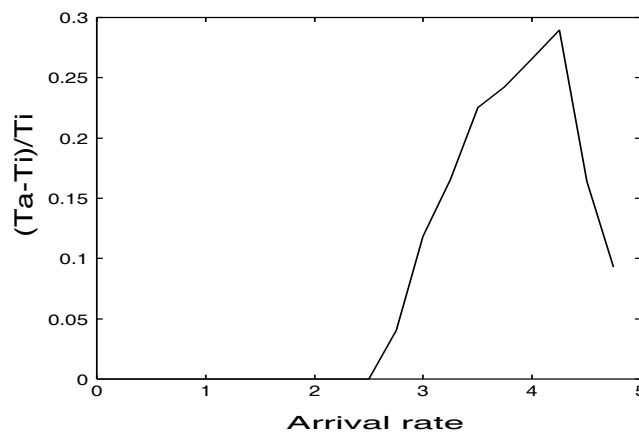


Fig. 3 the overall system performance degradation ratio for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=0.5$.

To check the type of this paradox, in Fig. 4, we computed the mean response time of every path individually, from entrance to exit, in both initial and augmented networks. From Fig. 4, we notice that, as anticipated the mean response times T_{iP1} and T_{iP2} of the two paths (P1 and P2), from entrance to exist, in the initial network are almost the same. But for the augmented network the mean response times T_{aP1} , T_{aP2} and T_{aP3} of the three paths (P1, P2 and P3), from entrance to exist, are significantly different from each other and T_{aP2} is always greater than or equal to T_{aP1} and T_{aP3} . We think that this occurs because the users who firstly decided to join the third server (P2) at the decision point $dp1$ have based their decision on the number of users in the fourth server upon their arrival and they did not anticipated that this number may be increased (by users who decided to join the fourth server (P3) at the decision point $dp2$) while they are being processed by the third server. Also from that figure, we can notice that $\min_{j=1,2,3} (T_{aPj}) \leq \max_{i=1,2} (T_{iPi})$ for all the values of λ which means that, in equilibrium, always there exist some users of the augmented network whom their response times improved as a result of adding the communication line. In other words, not all users of the augmented network are suffering from performance degradation even though the overall response time of the augmented network is greater than that of the initial network for $\lambda > 2.5$. This type of paradox we call it *weak paradox*, so what Calvert, Solomon and Ziedins [3] have

reported is a *weak paradox*.

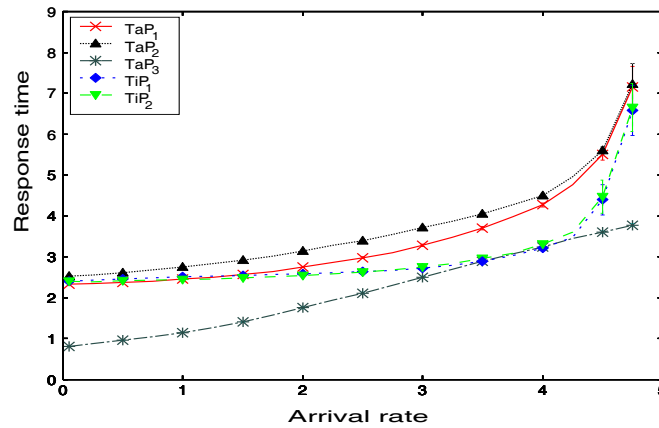


Fig. 4 mean response times of all the paths from entrance to exit in the initial and augmented networks for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=0.5$.

We also found the same type of paradox for $\mu_1=\mu_4=1.25$, $\mu_2=\mu_3=0.5$ ($\mu_1=\mu_4=2.5$, $\mu_2=\mu_3=1$) and the arrival rate λ varies from 0.05 to 2.5 (0.05 to 5) as shown in Fig. 6 (8). In these cases, we found that the worst case ratio of performance degradation is achieved for $\lambda = 2.25(4)$ and is about 20.72(20.51)% as shown in Fig 5(7).

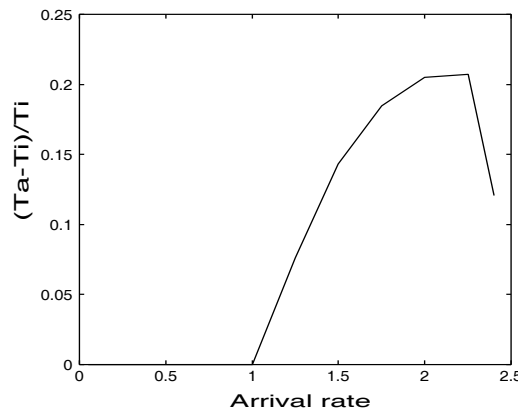


Fig. 5 the overall system performance degradation ratio for various values of λ while keeping $\mu_1=\mu_4=1.25$ and $\mu_2=\mu_3=0.5$.

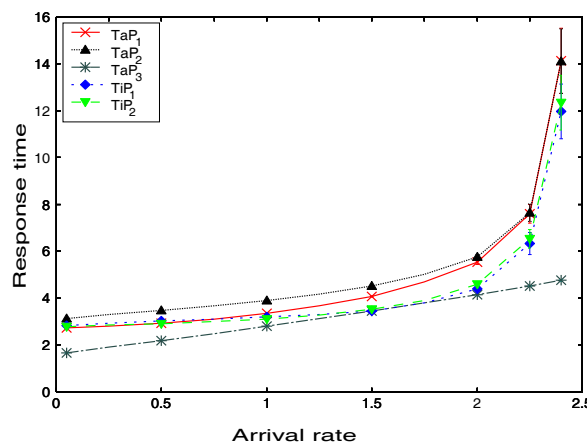


Fig. 6 mean response times of all the paths from entrance to exit in the initial and augmented networks for various values of λ while keeping $\mu_1=\mu_4=1.25$ and $\mu_2=\mu_3=0.5$.

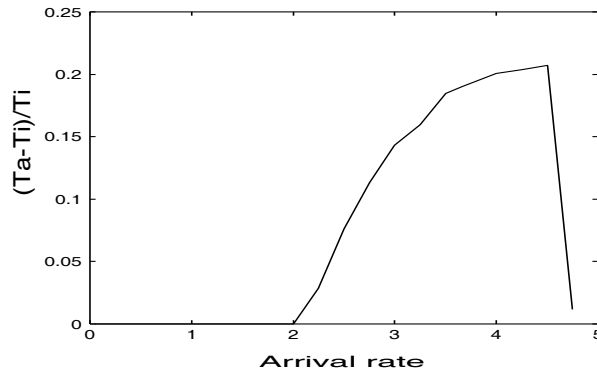


Fig 7. the overall system performance degradation ratio for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=1$.

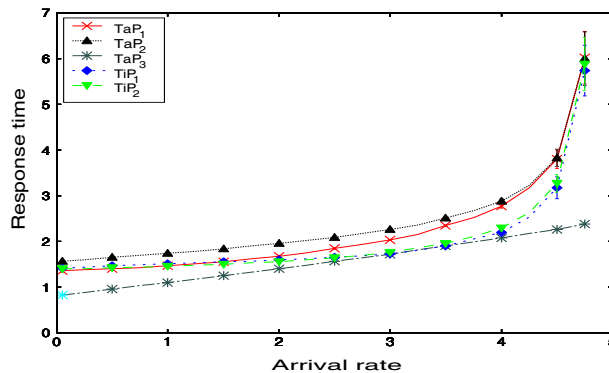


Fig. 8 mean response times of all the paths from entrance to exit in the initial and augmented networks for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=1$.

In the following examples, we found what we call *strong paradox*. Fig. 9 shows the overall system performance degradation ratio for $\mu_1=\mu_4=2.5$, $\mu_2=\mu_3=0.25$ and the arrival rate λ varies from 0.05 to 5. From this figure, we note that, for low arrival rates, the augmented network performs better than the initial network (note that, when $T_a < T_i$, we put the degradation ratio to zero which means no paradox is realized), while the reverse is true for high arrival rates. Braess's paradox appears for $\lambda > 2.5$ and the worst case ratio of performance degradation is achieved for $\lambda = 4$ and is about 49.99%.

Fig. 10 shows the mean response time of every path individually, from entrance to exit, in both initial and augmented networks. From that figure, we can notice that $\min_{j=1,2,3}(TaP_j) > \max_{i=1,2}(TiP_i)$ for the values of $2.75 < \lambda < 4.625$ which means that all the users of the augmented network in equilibrium suffer from performance degradation as a result of adding the communication line.

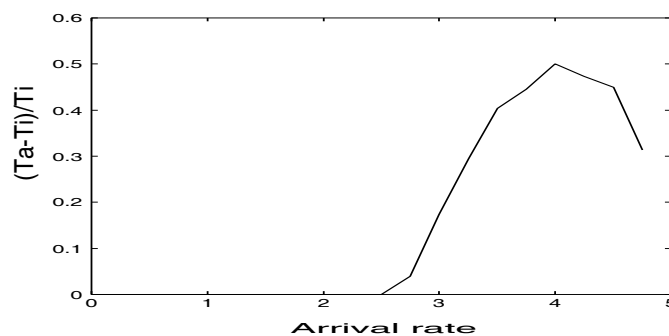


Fig. 9 the overall system performance degradation ratio for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=0.25$.

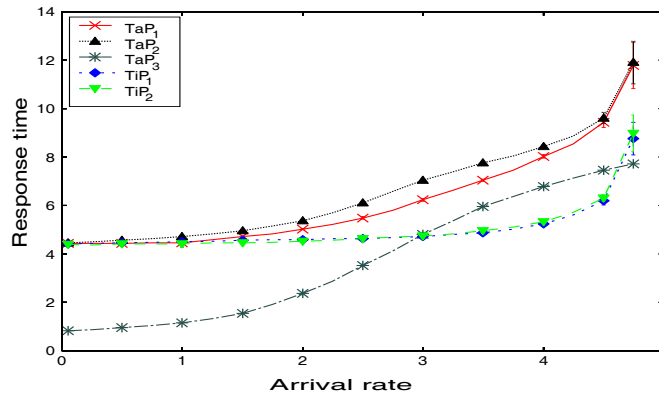


Fig. 10 mean response times of all the paths from entrance to exit in the initial and augmented networks for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=0.25$.

To estimate how much is the performance degradation, we compute $(\min_{j=1,2,3}(TaP_j) - \max_{i=1,2}(TiP_i)) / \max_{i=1,2}(TiP_i)$ as shown in Fig.11 to be the measure of performance degradation in that case. As shown in Fig. 11, all the users of the augmented network suffer from performance degradation for the values of $2.75 < \lambda < 4.625$ and the worst case ratio of performance degradation is achieved for $\lambda = 4$ and is about 27.44%.

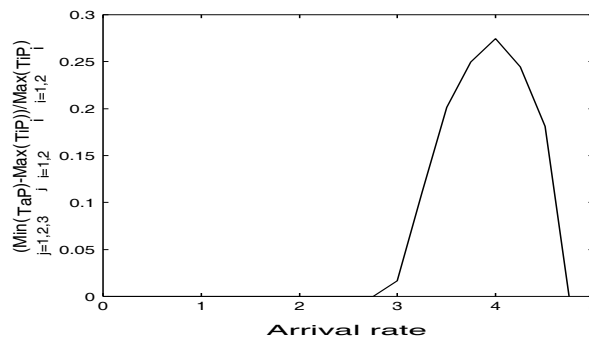


Fig. 11 the system performance degradation ration in a strong paradox case $((\min_{j=1,2,3}(TaP_j) - \max_{i=1,2}(TiP_i)) / \max_{i=1,2}(TiP_i))$ for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=0.25$.

Fig. 12 (15) shows the overall system performance degradation ratio for $\mu_1=\mu_4=2.5$, $\mu_2=\mu_3=0.1$ ($\mu_1=\mu_4=5$, $\mu_2=\mu_3=0.5$) and the arrival rate λ varies from 1.5 to 5 (1 to 10). From this figure, we notice that Braess's paradox appears for $\lambda > 2.5(5.25)$ and the worst case ratio of performance degradation is achieved for $\lambda = 4.5(8.25)$ and is about 75.36(50.22)%.

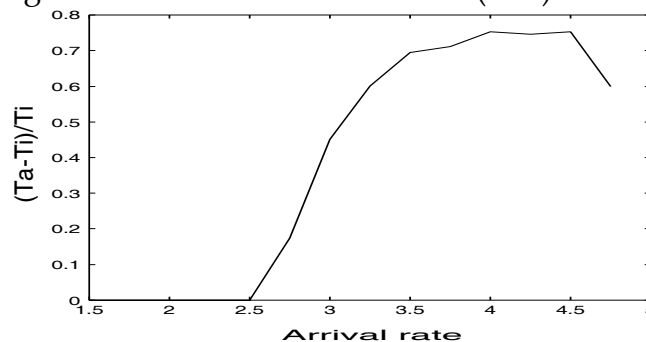


Fig. 12 the overall system performance degradation ratio for various values of λ while keeping $\mu_1=\mu_4=2.5$ and

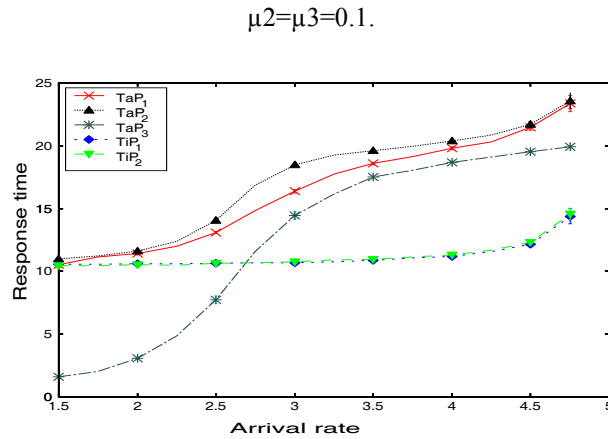


Fig. 13 mean response times of all the paths from entrance to exit in the initial and augmented networks for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=0.1$.

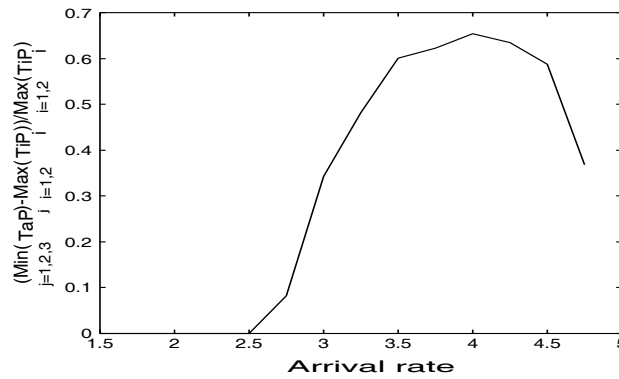


Fig. 14 the system performance degradation ration in a strong paradox case $((\min_{j=1,2,3}(TaP_j) - \max_{i=1,2}(TiP_i)) / \max_{i=1,2}(TiP_i))$ for various values of λ while keeping $\mu_1=\mu_4=2.5$ and $\mu_2=\mu_3=0.1$.

Fig. 13 (16) shows the mean response time of every path individually, from entrance to exit, in both initial and augmented networks. From that figure, we can notice that

$$\min_{j=1,2,3}(TaP_j) > \max_{i=1,2}(TiP_i) \text{ for the values of } 2.75 < \lambda \leq 4.75 \text{ (} 6 < \lambda < 9.25 \text{)}$$

which means that all the users of the augmented network in equilibrium suffer from performance degradation as a result of adding the communication line. Again, to estimate how much is the performance degradation, we computed $((\min_{j=1,2,3}(TaP_j) - \max_{i=1,2}(TiP_i)) / \max_{i=1,2}(TiP_i))$ as shown in Fig.14 (17) to be the measure of performance

degradation in that case. As shown from Fig. 14 (17), all the users of the augmented network suffer from performance degradation for the values of $2.75 < \lambda \leq 4.75$ ($6 < \lambda < 9.25$) and the worst case ratio of performance degradation is achieved for $\lambda = 4$ (8) and is about 65.35(27.44)%.

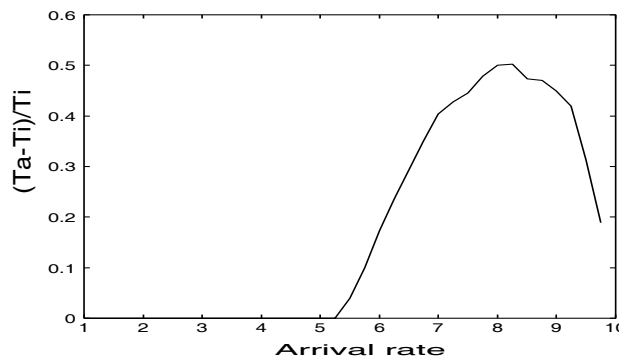


Fig. 15 the overall system performance degradation ratio for various values of λ while keeping $\mu_1=\mu_4=5$ and $\mu_2=\mu_3=0.5$.

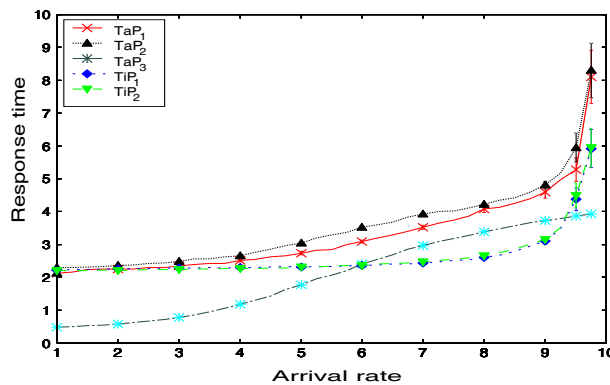


Fig. 16 mean response times of all the paths from entrance to exit in the initial and augmented networks for various values of λ while keeping $\mu_1=\mu_4=5$ and $\mu_2=\mu_3=0.5$.

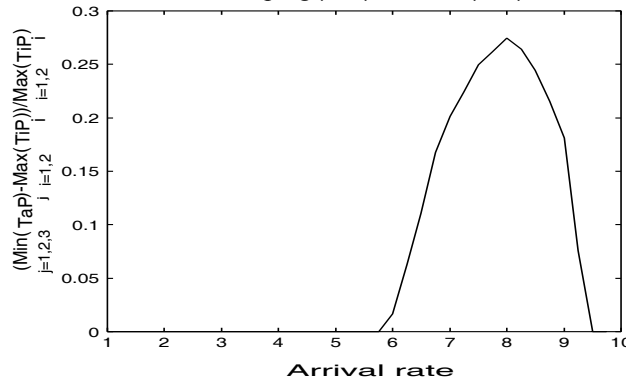


Fig. 17 the system performance degradation ration in a strong paradox case $((\min_{j=1,2,3}(TaP_j) - \max_{i=1,2}(TiPi)) / \max_{i=1,2}(TiPi))$ for various values of λ while keeping $\mu_1=\mu_4=5$ and $\mu_2=\mu_3=0.5$.

Generally from the previous examples, we can say that the degradation ratio in the considered computer network model decreases (increases) as the processing capacity of the two infinite servers increases (decreases) (i.e., the delay at the two infinite servers decreases (increases)) while keeping the processing capacity of the two FCFS servers (see figures 3, 7, 9 and 12). Also, we can conclude that the degradation ratio in the considered network model increases (decreases) as the processing capacity of the two FCFS servers increases (decreases) while keeping the processing capacity of the two infinite servers (see figures 3, 5 and 15).

5. Conclusion

Through a course of numerical experimentation using simulation, we studied the phenomenon of Braess's paradox on a distributed computer system in Wardrop equilibrium. Our simulation results showed that Braess's paradox can occur, namely in equilibrium the mean job response time in the augmented network, for some system parameter setting, can be greater than the mean job response time in the initial network. Indeed, two different types of paradox called *weak* and *strong* paradox have been characterized. Based on the definitions of the weak and strong paradox, we can say that what Calvert, Solomon and Ziedins reported is a weak paradox. One more point is that we report some more examples of weak paradox as well as some examples of strong paradox and we found that in the range of parameter values examined, the worst case ratio of performance degradation obtained in the considered network model is about 75(65)% for the case of weak (strong) paradox. Finally, from our simulation results, we can generally say that the degradation ratio in the considered network model decreases (increases) as the processing capacity of the two infinite servers increases (decreases) while keeping the processing capacity of the two FCFS servers and it increases (decreases) as the processing capacity of the two FCFS servers increases (decreases) while keeping the processing capacity of the two infinite servers.

If the results observed in this study hold generally, we think that more exhaustive research into these problems is worth pursuing in order to gain insight into the optimal design and QoS (quality of service) management of distributed computer systems, communication networks, Computing GRID etc.

REFERENCES

- [1] M. Patriksson, *The traffic Assignment Problem -Models and Methods*, (VSP, Utrecht, 1994).
- [2] Arora, N. and Sen, S., Resolving social dilemmas using genetic algorithms: Initial results, *Proc. of the 7th International Conference on Genetic Algorithms*, Michigan State University, 1997, 689-695.
- [3] Calvert, B., Solomon, W. and Ziedins, I., Braess's paradox in a queuing network with state-dependent routing, *J. Appl. Prob.*, 34, 1997, 134-154.
- [4] Cohen, J. E. and Jeffries, C., Congestion resulting from increased capacity in single server queuing networks, *IEEE/ACM Trans. Networking*, 5, 1997, 305-310.
- [5] Cohen, J. E. and Kelly, F. P., A paradox of congestion in a queueing networks, *J. Appl. Prob.*, 27, 1990, 730-734.
- [6] Kelly, F. P., Network routing, *Phil. Trans. R. Soc. Lond.*, A 337, 1991, 343-367.
- [7] R. B. Myerson., *Game Theory: Analysis of Conflict*, (Harvard University Press, Cambridge, Massachusetts, 1991).
- [8] Kameda, H. and Pourtallier, O., Paradoxes in distributed Decisions on optimal load balancing for networks of homogeneous computers, *J. of the ACM*, 49, 2002, 407-433.
- [9] El-Zoghdy, S. F., Kameda, H. and Li J., Numerical Studies on Paradoxes in Non-Cooperative Distributed Computer Systems, *Game Theory and Applications*, 9, 2003, 1-16.
- [10] El-Zoghdy, S. F., Kameda, H. and Li J., Numerical Studies on a Paradox for Non-Cooperative Static Load Balancing in Distributed Computer Systems, *Computers and Operation Research*, 33, 2006, 345-355.
- [11] El-Zoghdy, S. F., Studies on Braess-Like Paradoxes for Non-Cooperative Dynamic Load Balancing in Distributed Computer Systems, *Proc. of the IASTED International Conference on Parallel and Distributed Computing and Networks*, 2006, 238-243.
- [12] El-Zoghdy, S. F., Kameda, H. and Li J., A comparative study of static and dynamic individually optimal load balancing policies, *Proc. of the IASTED International Conference on Networks, Parallel and Distributed Processing and Applications*, 2002, 200-205.
- [13] Korilis, Y. A., Lazer, A. A. and Orda, A., Architecting noncooperative networks, *IEEE J. Selected Areas in Communications*, 13, 1995, 1241-1251.
- [14] Korilis, Y. A., Lazer, A. A. and Orda, A., Capacity allocation under non-cooperative routing, *IEEE Trans. Automatic Control*, 42, 1997, 309-325.
- [15] Korilis, Y. A., Lazer, A. A. and Orda, A., Avoiding the Braess paradox in noncooperative networks, *J. Appl. Prob.*, 36, 1999 211-222.
- [16] Orda, A., Rom, R. and Shimkin, N., Competitive routing in multiuser communication networks, *IEEE/ACM Trans. Networking*, 1, 1993, 614-627.
- [17] Morios, M. and Paul, S., The price of selfish routing, *Proc. of the 33rd ACM Annual Symposium of theory of computing*, 2000, 510-519.
- [18] Roughgarden, T. and Tardos, E., How bad is selfish routing?, *Proc. of the 41th IEEE Annual Symposium on Foundation of Computer Science*, 2000, 93-102.
- [19] Roughgarden, T., Designing networks for selfish users is hard, *Proc. of the 42th IEEE Annual Symposium on Foundation of Computer Science*, 2001, 472-481.
- [20] Libman, L. and Orda, A., The Designer's Perspective to Atomic Non-cooperative Networks, *IEEE/ACM Trans. Networking*, 7, 1999, 875-884.
- [21] LIBMAN, L. AND ORDA, A. (2001) Atomic resource sharing in non-cooperative networks, *Telecommunication systems*, 17, 385-409.
- [22] Bin Lu, Hongbin Zhang, Grid Load Balancing Scheduling Algorithm Based on Statistics Thinking, *icys, 2008 The 9th International Conference for Young Computer Scientists*, 2008, 288-292