Proceedings of CIICT2008

A GAME-THEORETIC ANALYSIS OF TCP VEGAS AND FAST TCP

Zhenhua Liu[1], Xiaoping Zhang[2], Dong Wang[3], Youjian Zhao[3], Hongtao Guan[3]

Department of Computer Science and Technology, Tsinghua University
Beijing, P. R. China (100084)
(8610) 62785818-6858
[1]liu-zh02@mails.tsinghua.edu.cn; [2]zhxp@tsinghua.edu.cn; [3]{wangmqi, zhaoyj, ght}@csnet1.cs.tsinghua.edu.cn

Keywords: TCP; Game Theory; Price of Anarchy; Tragedy of the Commons.

Abstract

In this paper, we evaluate the impact of selfish behavior through a game-theoretic analysis of TCP Vegas and FAST TCP. By theoretical analysis, we obtain the Nash Equilibrium in these games and study the loss of efficiency under Nash Equilibrium. Our results show that if the users are selfish, then the efficiency of the network as a whole relies on the utility function of individual users. The efficiency of the system degrades under some conditions and remains optimal under other conditions. This motivates us to borrow the notions of Price of Anarchy and Tragedy of the Commons in Game Theory to describe this efficiency loss. We calculate the Price of Anarchy and provide the necessary and sufficient condition under which the TCP Vegas Game (FAST TCP Game) is a Tragedy of the Commons. Finally, we validate the theoretical results by the simulations carried out using NS-2.

1 Introduction

About two decades ago, congestion collapses of the Internet led to the development of TCP's congestion control algorithm [22]. Like other communication systems, TCP's congestion control aims at efficient and fair share of common and usually scarce network resource among competing users. The design of such congestion control algorithm is difficult because the availability of resources and set of competing users vary over time unpredictably [3]. These constrains, together with the distributed nature of the control operation, make feedback control a good choice. Based on the different measure of congestion, current TCP versions can be divided into two classes: loss-based versions, such as TCP Reno and its enhancement [22,23,24,12,8,11,16]; and delay-based versions, such as TCP Vegas [10] and FAST TCP [3].

However, one main problem with all these versions is that they all assume that users are cooperative and always respect the protocol rule, and they do not provide any guarantee or verifying whether the users are cooperative or selfish. Many authors have noticed the conventional TCP end-to-end congestion control algorithms are voluntary and critically rely on users' cooperation [1,14,17,19,13,9,18,7]. In [1], the authors show that selfish behavior can result in system instability and inefficiency for many combinations of TCP version (loss-based versions) and AQM scheme implemented at routers. So the reason that the Internet is still functioning correctly is either the users are socially responsible or it is too difficult to modify the end-host protocol to behave selfishly. However, network operators can rely on neither of the two reasons.

In this paper we evaluate the impact of users' selfish behavior through a game-theoretic analysis of the delay-based versions of TCP (TCP Vegas and FAST TCP). We define the *TCP Vegas Game* (*FAST TCP Game*) in which TCP users (corresponding to flows) in a network can adjust their congestion control parameter α [1] to maximize their utilities.

It seems natural to suppose that selfish user could always gain by using more aggressive congestion control (larger α in our model). However, more aggressive congestion control can lead to larger queueing delay, and even loss. Since all users favor smaller queueing delay and the loss recovery is not a perfect process, there is always some "cost" resulted from more aggressive congestion control. Users must trade off between potential benefit and cost to maximize their own utilities. From the Game Theory's point of view, the Nash Equilibrium [2] reflects the balance between the benefit and the cost related to the parameter setting, so we analyze the parameter settings at the Nash Equilibrium.

From the optimization's perspective, it is shown in [5] that if user r chooses an amount ω_r to pay per unit time in order to receive a flow rate x_r proportional to ω_r and the network, as a whole, attempts to maximize $\Sigma_r\omega_r\log x_r$, then the optimal strategy (choosing ω_r) of each user is also the optimal strategy for the network as a whole. This means individual rationality necessarily leads to collective rationality. However, the pricing and payment mechanism is difficult, if not

[1] There are two parameters (α, β) in TCP Vegas. Throughout this paper we assume $\alpha = \beta$ for simplicity and the analysis in this paper can be easily extended to the situation where $\alpha \neq \beta$.

impossible, to implement, and the network optimization goal may not be so special, so in real network, individual rationality does *not necessarily* lead to collective rationality. Selfish users may overload the network and render the network as a whole suboptimal. This motivates us to study the Price of Anarchy in real circumstances. Furthermore, there is a close analogy in Game Theory called Tragedy of Commons problem [6], where each individual can improve his own utility by using more of a free resource, and the quality of the common resource degrades as it's been used more heavily. In equilibrium, the aggregated utility of all individuals is below that when they collaborate. So we also study the conditions under which the games in TCP Vegas and FAST TCP become Tragedy of the Commons.

Taking the above consideration into account, we seek to address the following questions in the paper:

(1) What are the parameter settings at Nash Equilibrium in the TCP Vegas Game and FAST TCP Game?
(2) How efficient is the network under Nash Equilibrium, or formally, how large is the Price of Anarchy?
(3) Under what condition is the TCP Vegas Game (FAST TCP Game) a Tragedy of the Commons?

Through both theoretical analysis and simulation validation in NS-2 [21], we make the following major contributions in this paper.

(1) First, we obtain the parameter settings of the users at the Nash Equilibrium in the TCP Vegas Game and FAST TCP Game under the circumstance of *generalized* utility function.
(2) Second, we provide the Price of Anarchy a in the TCP Vegas Game and FAST TCP Game also under the circumstance of *generalized* utility function and the conditions under which the TCP Vegas Game and FAST TCP Game are Tragedy of the Commons.
(3) Last but not least, we validate the results through simulation deployed in NS-2.

The remainder of the paper is organized as follows. In Section 2 we discuss related work. Section 3 presents the theoretical model. We provide the theoretical analysis results for the Nash Equilibrium, the Price of Anarchy, and Tragedy of the Commons in the TCP Vegas Game and FAST TCP Game in Section 4 and 5, respectively. We further use simulations in NS-2 to support the results in Section 6. Finally, Section 7 summaries the paper.

2 Related Work

[1] studies TCP games for several combinations of TCP versions and AQM schemes deployed at routers. But they only deal with some loss-based versions and we analyze the delay-base versions. The mathematical modelling in [9] bases on an over-simplified TCP called trivial TCP and they do not consider the impact of queueing delay and the work in [18] is based on two special utility functions. On contrast, our analysis is based on more realistic TCP Vegas and FAST TCP and we take the queueing delay into the generalized utility function. [7] considers the game where users set the connection number and in this paper we restrict our attention to the game where users only set up one connection but have freedom to set the parameter of congestion control algorithm.

3 The Model

In this section, we will define the TCP Vegas Game and FAST TCP Game. Formally, in the TCP Vegas game (FAST TCP Game, respectively), there are n ($n \geq 2$) TCP Vegas (FAST TCP) users (flows) competing for a *single* bottleneck link with capacity C and a buffer size B packets. Similar to [1,9,18], we also assume all users have infinite data to send and we only consider situations where the flows are symmetric (i.e., have identical RTT) and we only deal with symmetric Nash Equilibrium (i.e., Nash equilibrium where the congestion control parameters of the flows are all equal). Individual users are treated as players and in this paper we use users and players interchangeably. The feasible strategy set of user i, denoted by S_i, is $[\alpha^{min}, \alpha^{max}]$, where α^{min} and α^{max} are the smallest and largest value available for α_i. The feasible strategy space of the game is $S = S_1 \times S_2 \times \cdots \times S_n$. Then the action tuple is an n-dimension vector $\boldsymbol{\alpha}=(\alpha_1, \alpha_2, \cdots, \alpha_n) \in S$. For i, the actions other players taking is an $(n-1)$-dimension vector $\boldsymbol{\alpha}_{-i}=(\alpha_1, \alpha_2, \cdots, \alpha_{i-1}, \alpha_{i+1}, \cdots, \alpha_n)$. The objective of each user is to maximize its utility U_i by adjusting α_i. Under our model individual user solves the following problem.

$USER_i(U_i; \boldsymbol{\alpha}_{-i})$:

$$\max U_i(\alpha_i) \tag{1}$$

over

$$\alpha_i \in [\alpha^{\min}, \alpha^{\max}] \tag{2}$$

And the Nash Equilibrium $\boldsymbol{\alpha}^{ne}$ is defined as:

$$\alpha_i^{ne} = \mathrm{argmax}_{\alpha_i} U_i(\boldsymbol{\alpha}_{-i}^{ne}, \alpha_i), \forall i \tag{3}$$

Furthermore, under our model, the individual user considers both the throughput x_i and queueing delay q_i. So we can further write $U_i(\alpha_i) = F_i(x_i, q_i)$. For normal utility function,

$$\frac{\partial F_i}{\partial x_i} > 0, \frac{\partial^2 F_i}{\partial {x_i}^2} < 0, \frac{\partial^2 F_i(x_i)}{\partial x_i \, \partial x_j} = 0, \text{for } x_i \geq 0 \tag{4}$$

$$\frac{\partial F_i}{\partial q_i} < 0, \frac{\partial^2 F_i}{\partial {q_i}^2} > 0, \frac{\partial^2 F_i(q_i)}{\partial q_i \, \partial q_j} = 0, \text{for } q_i \geq 0 \tag{5}$$

The objective of the network as a whole is to maximize the aggregated utility $U = \sum_i U_i$. Under our model the system as a whole solves the following problem.

$SYSTEM_i(U)$:

$$\max\ U(\boldsymbol{\alpha}) \tag{6}$$

over

$$\boldsymbol{\alpha} \in [\alpha^{\min}, \alpha^{\max}]^n \tag{7}$$

The system optimal action vector is defined as:

$$\boldsymbol{\alpha}^{opt} = \mathrm{argmax}_{\boldsymbol{\alpha}} U(\boldsymbol{\alpha}) \tag{8}$$

4 TCP Vegas Game

In this section we analyze the TCP Vegas Game to answer the three questions in Section 1 for the TCP Vegas Game.

4.1 Nash Equilibrium

In order to obtain the Nash Equilibrium of TCP Vegas Game, we must first derive the relationship between x_i, q_i and α_i. Because most routers implement FIFO DropTail, we will analyze this scheme.

TCP Vegas updates the congestion window every RTT according to:

$$\mathrm{w} \leftarrow \mathrm{w} + sgn(\alpha - \mathrm{w} \cdot \mathrm{RTT_{min}}/\mathrm{RTT}) \qquad (9)$$

where $\mathrm{RTT_{min}}$ is the minimum RTT observed so far and $sgn(z) = -1$ if $z < 0$, 0 if $z = 0$, and 1 if $z > 0$. Then we can see that if the buffer is larger enough so that $B \geq \sum_i \alpha_i$, TCP Vegas will reach the equilibrium under which $x_i = \alpha_i C / \sum_j \alpha_j$, and $q_i = \sum_j \alpha_j / C$. If the buffer is not large enough, there will be packet loss before TCP Vegas reach the equilibrium and TCP Vegas will behave more likely TCP Reno [20]. Since we focus on the game under TCP Vegas, we first analysis the case where the buffer is larger enough. The discussion of the situation of insufficient buffer is omitted due to the space limitation.

For each user, $U_i(\alpha_i) = F_i(x_i, q_i) = F_i\left(\alpha_i C / \sum_j \alpha_j,\ \sum_j \alpha_j / C\right)$. So his optimal strategy of α_i^* meets the following first-order condition $\left.\frac{\partial U_i(\alpha_i)}{\partial \alpha_i}\right|_{\alpha_i=\alpha_i^*} = \left.\frac{\partial F_i}{\partial x_i}\cdot\frac{\partial x_i}{\partial \alpha_i} + \frac{\partial F_i}{\partial q_i}\cdot\frac{\partial q_i}{\partial \alpha_i}\right|_{\alpha_i=\alpha_i^*} = \left.\frac{\partial F_i}{\partial x_i}\cdot\frac{C(\sum_j \alpha_j - \alpha_i)}{(\sum_j \alpha_j)^2} + \frac{\partial F_i}{\partial q_i}\cdot\frac{1}{C}\right|_{\alpha_i=\alpha_i^*} = 0$. We then consider the second-order condition. From Equation (4) and (5), we obtain $\frac{\partial^2 U_i(\alpha_i)}{\partial {\alpha_i}^2} = \frac{\partial^2 F_i}{\partial {x_i}^2}\cdot\frac{C(\sum_j \alpha_j - \alpha_i)}{(\sum_j \alpha_j)^2} - \frac{\partial F_i}{\partial x_i}\cdot\frac{2C(\sum_j \alpha_j - \alpha_i)}{(\sum_j \alpha_j)^3} + \frac{\partial^2 F_i}{\partial {q_i}^2}\cdot\frac{1}{C} < 0$. So the crossing point of n best-response function is the Nash Equilibrium $\boldsymbol{\alpha}^{\mathrm{ne}}$.

In order to make this analysis more clear, we consider the following class of utility functions: $U_i(\alpha_i) = F_i(x_i, q_i) = \frac{{x_i}^a}{{q_i}^b}, 0 < a < 1, b > 0$. We can explain this utility function as follows: players of the TCP Vegas Game favor high throughput and dislike large queueing delay, so the utility function is positively correlated with the throughput and negatively correlated with the queueing delay, and the parameter a and b stand for the extent to which players of the TCP Vegas Game favor high throughput and dislike large queueing delay, respectively. The constraints on the value range, a and b, are because of the nature of the normal utility shown in Equation (4) and (5).

For this class of utility functions, we need to solve the Nash Equilibrium from the n first-order conditions:

$$\begin{cases} \frac{C^{a+b}{\alpha_1}^{a-1}(a\sum_j \alpha_j - (a+b)\alpha_1)}{(\sum_j \alpha_j)^{a+b+1}} = 0 \\ \frac{C^{a+b}{\alpha_2}^{a-1}(a\sum_j \alpha_j - (a+b)\alpha_2)}{(\sum_j \alpha_j)^{a+b+1}} = 0 \\ \cdots \\ \frac{C^{a+b}{\alpha_n}^{a-1}(a\sum_j \alpha_j - (a+b)\alpha_n)}{(\sum_j \alpha_j)^{a+b+1}} = 0 \end{cases} \qquad (10)$$

Consider the nature of the symmetry Nash Equilibrium, $\alpha_i^{\mathrm{ne}} = \alpha_j^{\mathrm{ne}}, \forall i, \forall j$ and we use α^{ne} to denote this value. Equation (10) is always satisfied under condition $b = (n-1)a$. This means if the positive effect of high throughput and negative effect of larger queueing delay from setting larger α_i completely counteract, then players of the TCP Vegas Game have no preference on different α_i and at Nash Equilibrium, $\alpha_i^{\mathrm{ne}} = \alpha_j^{\mathrm{ne}} = \alpha^{\mathrm{ne}}, \forall i, \forall j$ and α^{ne} can be any positive value from $[\alpha^{\min}, \alpha^{\max}]$. Without this condition we cannot obtain the Nash Equilibrium from Equation (10). Instead, the Nash Equilibrium is achieved at the boundary point. It is not hard to obtain that $\alpha_i^* \to +\infty$ when $b < (n-1)a$ and $\alpha_i^* \to 0$ when $b > (n-1)a$. This means if the positive effect of high throughput from setting larger α_i is larger than negative effect of larger queueing delay (under condition $b < (n-1)a$), then players of the TCP Vegas Game will favor larger α_i and at Nash Equilibrium $\alpha_i^{\mathrm{ne}} = \alpha^{\max}, \forall i$; if the positive effect of high throughput is smaller than negative effect (under condition $b < (n-1)a$), then players of the TCP Vegas Game will favor smaller α_i and at Nash Equilibrium $\alpha_i^{\mathrm{ne}} = \alpha^{\min}, \forall i$.

We can obtain the same result from the best-response analysis [2]. Without loss of generality, we consider the best-response of player 1 to the actions of other players. If all the other players set $\alpha^{(0)}$ at the beginning, from the first order condition, we can get $\alpha_i^{(1)}(\boldsymbol{\alpha}_{-i} = (\alpha^{(0)}, \alpha^{(0)}, \cdots, \alpha^{(0)})) = \frac{a(n-1)}{b}\alpha^{(0)}$ This means $\alpha^{(1)} = \frac{a(n-1)}{b}\alpha^{(0)}$. Then this process goes on and the Nash Equilibrium is achieved at the stable state. Then we can obtain

$$\alpha^{\mathrm{ne}} = \alpha^{(\infty)} = \left(\frac{a(n-1)}{b}\right)^{\infty}\alpha^{(0)} = \begin{cases} 0 \quad if\ 0 < \frac{a(n-1)}{b} < 1 \\ \alpha^{(0)}\ if\ \frac{a(n-1)}{b} = 1 \\ +\infty\ if\ \frac{a(n-1)}{b} > 1 \end{cases} \qquad (11)$$

This is the same with the results from Equation (10).

From all the analysis above, we have the following theorem:

Theorem 1 *(Nash Equilibrium)*: For the TCP Vegas Game with symmetric utility function $U_i(\alpha_i) = F_i(x_i, q_i) = \frac{{x_i}^a}{{q_i}^b}, 0 < a < 1, b > 0, \forall i$, the symmetric Nash Equilibrium is:

$$\boldsymbol{\alpha}^{\mathrm{ne}} = \begin{cases} (\alpha^{\min}, \alpha^{\min}, \cdots, \alpha^{\min}) & if\ 0 < \frac{a(n-1)}{b} < 1 \\ (\alpha^{\mathrm{ne}}, \alpha^{\mathrm{ne}}, \cdots, \alpha^{\mathrm{ne}}) & if\ \frac{a(n-1)}{b} = 1 \\ (\alpha^{\max}, \alpha^{\max}, \cdots, \alpha^{\max}) & if\ \frac{a(n-1)}{b} > 1 \end{cases} \qquad (12)$$

Where $\alpha^{\mathrm{ne}} \in [\alpha^{\min}, \alpha^{\max}]$

4.2 Price of Anarchy

In order to quantify the loss of the efficiency of the worst Nash Equilibrium, we introduce the notion of Price of Anarchy [4]. Price of Anarchy is defined as the ratio of system cost at the worst Nash Equilibrium to the system performance at the system cost at the system optimal point. Here we consider the system cost as the $1/U(\boldsymbol{\alpha})$. So the Price of Anarchy can be calculated from the Equation (13):

$$\text{PoA} = U(\boldsymbol{\alpha}^{\text{opt}})/U(\boldsymbol{\alpha}^{\text{ne}}) \quad (13)$$

Here we still consider the symmetric utility function $U_i(\alpha_i) = F_i(x_i, q_i) = \frac{x_i^a}{q_i^b}, 0 < a < 1, b > 0, \forall i$. The Nash Equilibrium is obtained from the last subsection, so we only need to derive the system optimal point. For the system, it aims at maximizing the aggregated utility $U(\boldsymbol{\alpha}) = \sum_i U_i(\alpha_i) = \sum_i \frac{x_i^a}{q_i^b} = \frac{C^{a+b}}{(\sum_i \alpha_i)^{a+b}} \sum_i \alpha_i^a$. Because when $0 < a < 1$, $\sum_i \alpha_i^a \le (\sum_i \alpha_i)^a$ and the inequation becomes equation if and only if all α_i are the same, at the system optimal point $\alpha_i^{\text{opt}} = \alpha_j^{\text{opt}} = \alpha^{\text{opt}}, \forall i, \forall j$. Then the system utility function is $U(\boldsymbol{\alpha}) = \frac{C^{a+b}}{n^{a+b-1}} \cdot \frac{1}{(\alpha^{\text{opt}})^b}$. Because $b > 0$, the system optimal point is $\boldsymbol{\alpha}^{\text{opt}} = (\alpha^{\min}, \alpha^{\min}, \cdots, \alpha^{\min})$.

From the analysis above, we have the following theorem about the Price of Anarchy.

Theorem 2 *(Price of Anarchy)*: For the TCP Vegas Game with symmetric utility function $U_i(\alpha_i) = F_i(x_i, q_i) = \frac{x_i^a}{q_i^b}, 0 < a < 1, b > 0, \forall i$, the Price of Anarchy is:

$$\text{PoA} = \begin{cases} 1 & \text{if } 0 < \frac{a(n-1)}{b} < 1 \\ \left(\frac{\alpha^{\max}}{\alpha^{\min}}\right)^b & \text{if } \frac{a(n-1)}{b} \ge 1 \end{cases} \quad (14)$$

4.3 Tragedy of the Commons

From [6], a game is a Tragedy of the Commons when the following two conditions hold:

(1) Condition 1: There is always an incentive for a new user to become selfish. This means that at Nash equilibrium, all users are selfish.

(2) Condition 2: The aggregated utility for selfish users in the Nash equilibrium is under the aggregated utility when all users collaborate.

Formally, we have the following theorem:

Theorem 3 *(Tragedy of the Commons)*: For the TCP Vegas Game with symmetric utility function $U_i(\alpha_i) = F_i(x_i, q_i) = \frac{x_i^a}{q_i^b}, 0 < a < 1, b > 0, \forall i$, $\frac{a(n-1)}{b} > 1$ is the necessary and sufficient condition for the game to be the Tragedy of the Commons.

5 FAST TCP Game

FAST TCP updates the congestion window periodically according to [3]:

$$\text{w} \leftarrow \text{w} + \gamma(\alpha - \text{w} \cdot \text{RTT}_{\min}/\text{RTT}) \quad (15)$$

where $\text{RTT}_{\min}$ is the minimum RTT observed so far. So we can regard FAST TCP as the fast version of TCP Vegas. So the Nash Equilibrium of the FAST TCP Game is the same as the TCP Vegas Game, and the analysis of the Price of Anarchy and the Tragedy of the Commons is exactly the same as that in the TCP Vegas Game.

6 Simulations

In this section, we will validate the analysis results in Section 4 and 5 using NS-2 simulations. We will first introduce the simulation methodology. Then we will provide the simulation results.

6.1 Simulation Methodology

Since we assume that all users compete for a single bottleneck link, we use the standard dumb-bell topology shown in Figure 1 for our simulations. User i traverses the path from S_i to D_i. In all our simulations, we set the bottleneck capacity C to 10Mbps and we fix n = 10. The buffer is large enough. The simulation methodology for arriving at the Nash Equilibrium is similar to [1] when users are allowed to vary the parameter α_i from 1 to 100. So $\alpha^{\min} = 1, \alpha^{\max} = 100$.

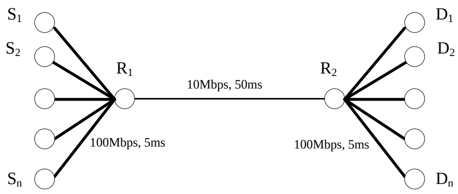
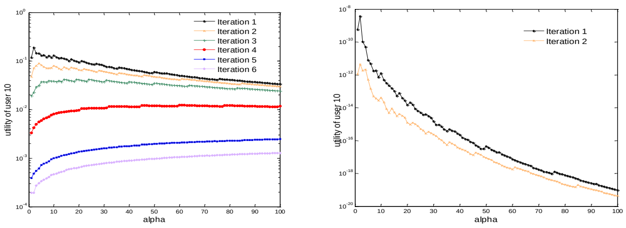


Figure 1: A single-bottleneck topology

We now brief introduce the methodology and readers can refer [1] for more details. We run the simulations in iterations. In the j^{th} iteration, we fix the parameters of User 1 to 9 to α^j, and User 10 varies it parameter α_{10} from 1 to 100 to find the optimal value $\alpha^{j,\text{best}}$. We use this optimal value as the value of α^{j+1} and go on with the $(j+1)^{th}$ iteration. The simulation stops when at some iteration k, $\alpha^{k,\text{best}} = \alpha^k$ and this value is the Nash Equilibrium value of α^{ne}. For the system as a whole, obviously at the system optimal point $\alpha_i^{\text{opt}} = \alpha_j^{\text{opt}} = \alpha^{\text{opt}}, \forall i, \forall j$. So we will vary α from 1 to 100 to find the system optimal value α^{opt}. We run simulations for TCP Vegas and FAST TCP. For each version, we consider two situations where b is large enough (a = 0.5, b = 10) and b is not large enough (a = 0.5, b =1).

6.2 Nash Equilibrium in TCP Vegas Game

The simulation results of the Nash Equilibrium in the TCP Vegas Game are shown in Figure 3. All the results of FAST TCP is exactly the same.



(a) (b)

Figure 3: Simulation Results for Nash Equilibrium in TCP Vegas Game. (a) a = 0.5, b = 1; (b) a = 0.5, b = 10.

As shown in Figure 3, for a = 0.5, b =10, Nash Equilibrium is $\boldsymbol{\alpha}^{\text{ne}} = (100, 100, \cdots, 100) = (\alpha^{\max}, \alpha^{\max}, \cdots, \alpha^{\max})$; for a = 0.5, b =10, the Nash Equilibrium is $\boldsymbol{\alpha}^{\text{ne}} = (2, 2, \cdots, 2) \approx (\alpha^{\min}, \alpha^{\min}, \cdots, \alpha^{\min})$, both are (almost) the same as the theoretical analysis results.

Simulation result for the system optimal point is shown in Figure 4. We can see that the system optimal point is always $\boldsymbol{\alpha}^{\text{opt}} = (1, 1, \cdots, 1) = (\alpha^{\min}, \alpha^{\min}, \cdots, \alpha^{\min})$, which is the same as the theoretical analysis result.

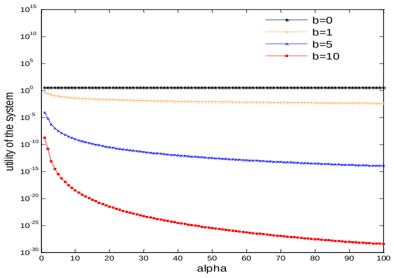


Figure 4: Simulation Results for system optimal point in TCP Vegas Game (a = 0.5)

7 Conclusion

In this paper we provide a game-theoretic analysis of TCP Vegas and FAST TCP. From the theoretical analysis and simulation results we answer all the three questions proposed in Section 1.1 and we show that selfish behavior can cause system inefficiency.

Acknowledgements

This research is supported by National Natural Science Foundation of China under Grant No. 90604029, 60773150 and National High Technology Research and Development Program of China under Grant No. 2077AA01Z219. The author would like to thank S. H. Low for his kind guidance.

References

[1] A. Akella, S. Seshan, K. Karp, S. Shenker, C. Papadimitrou. “Selfish behavior and stability of the Internet: a game-theoretic analysis of TCP”, *ACM SIGCOMM*, (2002).

[2] D. Fundenberg, J. Tirole. “Game Theory”, MIT Press, Cambridge, MA, (1991).

[3] D. X. Wei, C. Jin, S. H. Low, S. Hegde. “FAST TCP: motivation, architecture, algorithms, performance”, *IEEE/ACM Transactions on Networking*, 14(6), pp. 1246-1259, (2006).

[4] E. Koutsoupias, C. Papadimitriou. “Worst-case equilibria”, *Lecture Notes in Computer Science*, 1563, pp. 404–413, (1999).

[5] F. P. Kelly, A. Maulloo, D. Tan. “Rate control for communication networks: Shadow prices, proportional fairness and stability”, *Journal of Operations Research Society*, 49(3), pp. 237–252, (1998).

[6] G. Hardin. “The Tragedy of the Commons”, *Science*, 162, pp. 1243–1248, (1968).

[7] H. Zhang, D. Towsley, W. Gong. “TCP Connection Game: A Study on the Selfish Behavior of TCP Users”, *ICNP*, (2005).

[8] Janey Hoe. “Improving the startup behavior of a congestion control scheme for TCP”, *ACM SIGCOMM*, (1996).

[9] L. López, G. R. Almansa, S. Paquelet, A. Fernández. “A mathematical model for the TCP tragedy of the commons”, *Theoretical Computer Science*, 343(1-2), pp. 4-26, (2005).

[10] L. S. Brakmo, L. L. Peterson. “TCP Vegas: end-to-end congestion avoidance on a global Internet”, *IEEE Journal on Selected Areas in Communications*, 13(8), pp. 1465–80, (1995).

[11] M. Allman, V. Paxson, W. Stevens. “TCP congestion control”, *RFC 2581*, (1999).

[12] M. Mathis, J. Mahdavi, S. Floyd, A. Romanow. “TCP SelectiveAcknowledgment Options”, *RFC 2018*, (1996).

[13] M. T. Hsiao, A. Lazar. “A game-theoretic approach to decentralized flow control of markovian queue networks”, *Performance*, (1987).

[14] R. Garg, A. Kamra, V. Khurana. “A game-theoretic approach towards congestion control in communication networks”, *ACM Computer Communication Review* 32(3), pp. 47-61 (2002).

[15] R.Pan, B. Prabhakar, K.Ps ounis. “CHOKE, a stateless active queue management scheme for approximating fair bandwidth allocation”, *IEEE INFOCOMM*, (2000).

[16] S. Floyd, T. Henderson. “The NewReno modification to TCP’s Fast Recovery algorithm”, *RFC 2582*, (1999).

[17] S. Floyd, K. Fall. “Promoting the use of end-to end congestion control in the Internet”, *IEEE/ACM Transaction on Networking*, 7(4), pp. 458–471, (1999).

[18] S Molnár, S Kardos, T. A. Trinh. “Parameter Setting Games in TCP Vegas and FAST TCP”, 19*th International Teletraffic Congress*, (2005).

[19] S. Shenker. “Making greed work in networks: A game-theoretic analysis of switch service disciplines”, *IEEE/ACM Transactions on Networking*, 3(6), pp. 819-831, (1995).

[20] T. Bonald. “Comparison of TCP Reno and TCP Vegas via Fluid Approximation”, *Workshop on the Modeling of TCP*, (1998).

[21] The network simulator. http://www.isi.edu/nsnam/ns/.

[22] V. Jacobson. “Congestion avoidance and control”, *ACM Computer Communication Review*, 18(4), pp. 314–329, (1988).

[23] V. Jacobson, R. Braden, D. Borman. “TCP extensions for high performance”, *RFC 1323*, (1992).

[24] W. Stevens. “TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery algorithms”, *RFC 2001*, (1997).