

# Efficiency of repeated network interactions

Judith Timmer\*      Michel Mandjes†

February 9, 2007

## Abstract

In this paper we consider a network with interactions by two users. Each of them repeatedly issues download requests on the network. These requests may be unsuccessful due to congestion or non-congestion related errors. A user decides when to cancel a request (that is, what his impatience threshold is) and how long to wait before reissuing his request after cancellation of the previous request (that is, what his waiting time will be). This pair of impatience threshold and waiting time is his *strategy*. If a customer decides not to wait but to reissue his request immediately, that is, he sets his waiting time to zero, then he is said to use a so-called *restart strategy*. The goal of the user is to maximize the number of successful requests over a given time span.

We study optimal strategies for the users in a game-theoretic framework. We find that in case congestion is the only cause of unsuccessful requests then each of the users will be very patient and any waiting time is optimal. Hence, restart strategies are among the optimal strategies. Second, in case non-congestion related errors may occur, users will also set large impatience times, but now they will set waiting times to zero; in other words: they immediately reissue an unsuccessful download. In this case all optimal strategies are restart strategies. Hence, in both cases restart strategies are among the optimal strategies. Finally, implementing social optimal strategies instead of individual optimal ones cannot improve the efficiency of the network usage.

**Key words:** network efficiency, congestion, impatience, restart strategy, noncooperative game

**2000 Mathematics Subject Classification:** 90B18, 60K30, 91A10, 91A05

## 1 Introduction

The amount of traffic transmitted over the Internet traffic is ever increasing. The main part of this traffic consists of transfers like video, data, email, etc. The completion times of these

---

\*Corresponding author. Department of Applied Mathematics, University of Twente, P.O. Box 217, 7500 AE Enschede, the Netherlands. Phone: +31-53-4893419. Fax: +31-53-4893069. Email: [j.b.timmer@utwente.nl](mailto:j.b.timmer@utwente.nl).

†CWI, P.O. Box 94079, 1090 GB Amsterdam, the Netherlands. Email: [michel@cwi.nl](mailto:michel@cwi.nl).

tasks vary over time due to several causes. A first, evident, cause of varying completion times is Internet congestion: as the level of congestion fluctuates, the completion times do so as well. In the second place, one often observes that if a webpage takes a long time to download, then pressing the reload button may result in the page being promptly loaded. In this case we say the download request was hindered by *non-congestion related errors*. Users, or agents, on the Internet do not know which of these two causes, if any, occurs. Hence, it may be attractive for an agent to reissue his request immediately instead of reissuing it after a break. An agent might cancel a request if he feels he has been waiting too long; this personal maximum waiting time is called his *impatience threshold*. After canceling a request he might want to wait some time before putting down a new request if he believes this improves his chances on a completion time smaller than his impatience threshold. If he decides to use a zero waiting time, this agent is said to use a so-called *restart strategy* [2]. Such a strategy is often used on the web when a page seems to take too long to be retrieved: users impatiently press the reload button, and, as mentioned above, often the page is promptly downloaded. Upon completion of a request the agent spends some time reading or studying the material that was downloaded from the network. After finishing this, he immediately puts down a new request for a download. The problem under study is to determine for each agent an impatience threshold and a waiting time that maximize his number of successful requests over a given time span.

Various related problems have been studied in the literature. Chalasani, Jha, Shehory and Sycara [1] study restart strategies for agents on the web as a tool for reducing the expected completion times. Completion times are modeled by random variables with a certain distribution. The authors present optimal restart strategies for several distributions of completion times, as well as conditions under which these strategies indeed reduce the expected completion time. Schroeder, Boro and McCann [5] also consider immediate reissuing of requests. They consider four phases of agent interaction and determine restart strategies such that the mean completion time of requests for some sample data is minimized. Maurer and Huberman [2] conducted several simulations in which users decide whether or not to use the Internet. Their findings are that in case of multiple users there exist optimal restart strategies. Van Moorsel and Wolters [6] analyze and optimize the completion time for a class of jobs under restarts. Their restart times correspond to our impatience thresholds, so these times determine when to abort a download request. Their results include algorithms that optimize the restart policy and they identify characteristics of optimal restart times.

In this paper we have chosen to consider the most stylized setting possible: we study a resource used by two users (or: customers). A user not only decides when to cancel a request (that is, what his impatience threshold is) but also may decide to wait a certain time before reissuing his request (that is, what his waiting time is). If a customer decides not to wait then his strategy is a restart strategy. Furthermore we consider the two causes of unsuccessful

requests identified above: congestion due to simultaneous network usage, and non-congestion related errors.

An interesting feature of our model is that the customers do not know if any of these two causes occur. The goal of each customer is to maximize his number of successful requests over a given time span by choosing a suitable impatience threshold and waiting time. Since the choice of one agent influences the outcome of the other, the two customers are involved in a two-person noncooperative game. The layout of this game is simple but the game itself is difficult to analyze due to the fact that the times the customers spend on successful download requests are stochastic by assumption. Therefore, simulation is used to study equilibria of this game.

Our results are as follows. First, in case congestion is the only cause of unsuccessful requests then each of the users will be very patient and any waiting time is optimal. Hence, restart strategies are just one class of optimal strategies. Second, in case non-congestion related errors may occur users will also set large impatience times but now they will set waiting times to zero, that is, they immediately reissue an unsuccessful download. In this case all optimal strategies are restart strategies. Hence, in both cases restart strategies are among the optimal strategies. Finally, we argue that implementing social optimal strategies, rather than ‘individually optimal’ strategies, does not improve the efficiency of the network usage.

The outline of this paper is as follows. In the next section we introduce our model. We assume first that congestion is the only source of unsuccessful downloads. In Section 3 non-congestion related errors may occur. Thereafter, in Section 4 the efficiency of the network usage is studied. Section 5 contains concluding remarks.

## 2 The model

In this section we describe and study a model of repeated network interactions with congestion. Later, in Section 3 we extend our model to include non-congestion related errors.

Consider two customers who wish to download and read pages from a network. If both use the network at the same time, the network is congested, and we say that the network is busy and slow; otherwise the network is fast. Clearly, in congestion periods it takes relatively long to complete a download. If the network is congested while the customer tries to download his page, he may get impatient before the page is completely downloaded (more precisely: his impatience-threshold is reached). Then he cancels the download attempt, waits some time (because he notices that the network was congested) and issues a new download request. When issuing such a request, customers cannot see whether the network is congested or not, and in addition they do not know the characteristics of the other user.

Customer 1 wishes to download pages of size  $J_1$  ( $J_1$  is the time it takes to download such a page in an uncongested network). If downloading such a page takes longer than his impatience

threshold  $T_1$ , he will cancel the request and wait for a time  $W_1$  before starting a new attempt. If downloaded successfully, he reads the page in a time  $R_1$ , a realization of his exponential distributed reading time, after which he starts to download the next page. The impatience threshold should be such that  $T_1 \geq J_1$ , shorter thresholds make no sense because the page will never be completely loaded then, and  $T_1 \leq 2J_1$ , if the network is congested any page will be loaded twice as slow since both customers receive an equal share of the network capacity. Similarly we define the parameters  $J_2, T_2$  ( $J_2 \leq T_2 \leq 2J_2$ ),  $W_2$  and  $R_2$  for customer 2. Notice that if all parameters  $J_i, T_i, W_i$  and  $R_i$  would be exponentially distributed, then our model may be formulated as a Markov chain. Unfortunately, this chain cannot be analyzed in explicit terms. Hence, we consider a model with a single stochastic component.

Suppose first that congestion is the only cause of unsuccessful download requests (we come back to this in Section 3). At any time the situation  $s_i$  of customer  $i$  can be described by a five-tuple

$$s_i = (m_i, a_i, q_i, t_i, n_i).$$

In this situation  $m_i$  is the nearest point of time when customer  $i$  will change his action. For example, if a page is fully downloaded then he will start to read it. Thus, when the download request is completed, the customer's action changes from downloading to reading. The current action (waiting ( $w$ ), downloading ( $d$ ), or reading ( $r$ )) of customer  $i$  is described by  $a_i \in \{w, d, r\}$ . If the customer is currently downloading a webpage,  $a_i = d$ , the expected amount of the page to be downloaded at time  $m_i$  is  $q_i$ ,  $0 \leq q_i \leq J_i$ . Future actions may change this amount. Further, the download request should be fulfilled before time  $t_i$ , which is the time the current download started incremented by the impatience threshold  $T_i$ . Otherwise the customer becomes impatient and will cancel the download at time  $t_i$ . Finally,  $n_i$  is the number of pages already downloaded and read by this customer. Since customer  $i$  does not know whether or not the network is congested, he does not know  $m_i$  and  $q_i$  while he uses the network. The customer only notices a change of action if a page is downloaded successfully, if he becomes impatient or if he finished waiting or reading.

We assume the process starts at time  $t = 0$  and that both customers start downloading a page at that time. Thus the network is congested and the customers share the capacity of the network equally. It takes customer  $i$  a time  $2J_i$  to download his page while he becomes impatient at time  $T_i$ . Because  $T_i \leq 2J_i$  the customer will change action at time  $T_i$  when  $T_i/2 \leq J_i$  of the page will be loaded. Hence, initially

$$s_i = (T_i, d, T_i/2, T_i, 0).$$

The situations change at time  $t = \min\{m_1, m_2\}$ , the nearest point of time when one of the customers changes action. Suppose first that the situations only change due to a change of action by customer  $i$ , that is,  $t = m_i < m_k$ .

$a_i$	$a_k$	new $s_i$	new $s_k$
$d$	$d, r, w$	$(t + W_i, w, q_i, t_i, n_i)$ if $J_i > q_i$ , $(t + R_i, r, q_i, t_i, n_i)$ else	$(m_k, a_k, q_k, t_k, n_k)$ if $a_k \neq d$ , $(J_k + m_k/2 - q_k + t/2, d, J_k, t_k, n_k)$ else if $J_k + m_k/2 - q_k + t/2 \leq t_k$ , $(t_k, d, q_k - m_k/2 + t_k - t/2, t_k, n_k)$ else
$r, w$	$d$	$(t + T_i, d, T_i/2, t + T_i, n_i + \delta_i)$	$(2J_k + 2m_k - 2q_k - t, d, J_k, t_k, n_k)$ if $2J_k + 2m_k - 2q_k - t \leq t_k$ , $(t_k, d, q_k - m_k + t_k/2 + t/2, t_k, n_k)$ else
$r, w$	$r, w$	$(t + J_i, d, J_i, t + T_i, n_i + \delta_i)$	$(m_k, a_k, q_k, t_k, n_k)$

Table 2.1: *Change of situations at time  $t = m_i < m_k$  when customer  $i$  changes action.*

Assume first that customer  $i$  was downloading,  $a_i = d$ . Then he will change action at time  $t$  because he successfully downloaded a page,  $q_i = J_i$ , or because he gets impatient,  $t = t_i$ . If the download was successful the customer will start reading the page at time  $t = m_i$ . Reading the page takes a time  $R_i$  and changes the situation of customer  $i$  to  $s_i = (t + R_i, r, q_i, t_i, n_i)$ . If the customer gets impatient before finishing the download,  $t = t_i$ , he will wait a time  $W_i$  before trying a new attempt,  $s_i = (t + W_i, w, q_i, t_i, n_i)$ . Now that customer  $i$  becomes inactive, the situation of the other customer, customer  $k$  may change. If this customer was reading or waiting, then his situation remains unchanged. Otherwise, if customer  $k$  was also downloading then the network was congested up to time  $t$ , although the customer did not know this. At time  $t$  customer  $k$  already downloaded the amount  $q_k - (m_k - t)/2$ . (Recall that  $q_k$  is the expected amount downloaded by customer  $k$  at time  $m_k > t$ .) Now that the network is not congested as of time  $t$  the remaining amount  $J_k - (q_k - (m_k - t)/2)$  can be downloaded twice as fast as before. If this is completed before the customer gets impatient,  $t + J_k - (q_k - (m_k - t)/2) \leq t_k$  then his situation becomes  $s_k = (t + J_k - (q_k - (m_k - t)/2), d, J_k, t_k, n_k)$ . Otherwise, another  $t_k - t$  can be downloaded before getting impatient,  $s_k = (t_k, d, q_k - (m_k - t)/2 + t_k - t, t_k, n_k)$ .

If customer  $i$  was reading or waiting then his change of action at time  $t = m_i$  means that he will start a new download. If the other customer is downloading,  $a_k = d$  then the network is congested as of time  $t$ , although the customers do not know this. Hence, customer  $i$  will be impatient before completing his download while customer  $k$  is confronted with a slower network. If customer  $k$  was reading or waiting, then customer  $i$  expects a successful download. For details see Table 2.1 where  $\delta_i$  is a 0-1 variable indicating whether or not customer  $i$  finished reading:  $\delta_i = 1$  if  $a_i = r$  and  $\delta_i = 0$  if  $a_i = w$ .

$a_i$	$a_k$	new $s_i$	new $s_k$
$d$	$d, r, w$	$(t + R_i, r, q_i, t_i, n_i)$ if $J_i = q_i$ , $(t + W_i, w, q_i, t_i, n_i)$ else	$(t + J_k, d, J_k, t + T_k, n_k + \delta_k)$ if $a_k \neq d$ $(t + R_k, r, q_k, t_k, n_k)$ else if $J_k = q_k$ , $(t + W_k, w, q_k, t_k, n_k)$ else
$r, w$	$r, w$	$(t + T_i, d, T_i/2, t + T_i, n_i + \delta_i)$	$(t + T_k, d, T_k/2, t + T_k, n_k + \delta_k)$

Table 2.2: *Change of situations at time  $t = m_i = m_k$  when both customers change action.*

It may also happen that both customers change actions at time  $t$ ,  $t = m_1 = m_2$ . The resulting changes of situations are described in Table 2.2.

Each customer wishes to maximize the expected number of pages he can download and read in some time interval  $x$  by choosing a suitable stationary strategy pair of impatience threshold and waiting time that solves his optimization problem

$$\max_{T_i, W_i} n_i(T_i, W_i, T_k, W_k) \quad (2.1)$$

given the strategy  $(T_k, W_k)$  of the other user. This dependence on each other's strategies implies that the two customers are actually involved in a two-player noncooperative game. Each customer tries to optimize his objective function given the strategy of the other. The strategy pairs  $(T_i^*, W_i^*)$  and  $(T_k^*, W_k^*)$  of the customers are called *Nash equilibrium* strategies, [4], or optimal strategies, if none of the customers can download and read more pages by unilateral deviation. Hence,

$$n_i(T_i^*, W_i^*, T_k^*, W_k^*) \geq n_i(T_i, W_i, T_k^*, W_k^*)$$

for all customers  $i$  and all strategies  $(T_i, W_i)$ . The Tables 2.1 and 2.2 indicate that the analysis of this game with its repeated network interactions is difficult and complex because of the stochastic reading times although the basic model is simple. Conventional methods in noncooperative game theory cannot handle stochastic components, and so, it is hard to determine equilibria of this game. Therefore, simulation is used to study optimal strategies in this two-person network. For this, assume that the reading times of customer  $i$  are exponentially distributed with parameter  $l_i$ . (Hence, customer  $i$  has an expected reading time of  $1/l_i$ .) Our simulation model is based on the processes as described in the Tables 2.1 and 2.2.

To solve the optimization problems of the customers, sequential optimization is used. Consider the situation in which the customers first optimize over their impatience thresholds  $T_i$ ,  $i = 1, 2$ , while keeping the waiting time  $W_i$  fixed, and thereafter optimize over their waiting times. Simulation results, using a 95% confidence interval, are:

<i>Cust. 1</i>	$T_2 = 5.5$	5.6	5.7	5.8	5.9	<b>6.0</b>
$T_1 = 3.75$	0.00	0.00	0.00	0.00	0.00	40.37
3.80	0.00	0.00	0.00	0.00	0.00	43.10
3.85	0.00	0.00	0.00	0.00	0.00	44.85
3.90	0.00	0.00	0.00	0.00	0.00	45.53
3.95	0.00	0.00	0.00	0.00	0.00	46.13
<b>4.00</b>	49.85	50.11	50.24	50.30	50.70	<b>50.90</b>

<i>Cust. 2</i>	$T_2 = 5.5$	5.6	5.7	5.8	5.9	<b>6.0</b>
$T_1 = 3.75$	0.00	0.00	0.00	0.00	0.00	32.06
3.80	0.00	0.00	0.00	0.00	0.00	33.98
3.85	0.00	0.00	0.00	0.00	0.00	35.06
3.90	0.00	0.00	0.00	0.00	0.00	34.98
3.95	0.00	0.00	0.00	0.00	0.00	35.39
<b>4.00</b>	34.73	35.10	35.45	35.78	35.84	<b>36.04</b>

Table 2.3: Simulations with small job differences. *The tables above show the average number of pages downloaded and read by customer 1 (upper table) and customer 2 during a time period of length 300 over 1000 simulation runs. In these simulations the job sizes are  $J_1 = 2$  and  $J_2 = 3$ , the reading times are exponentially distributed with parameters  $l_1 = 1/3$  and  $l_2 = 1/4$ , and the waiting times are  $W_i = 0$ . All numbers in the tables are rounded to two decimals. The numbers in boldface indicate the unique equilibrium  $T_i = 2J_i$ ,  $i = 1, 2$ .*

- Similar job sizes (e.g.  $J_1 = 2$ ,  $J_2 = 3$ ): equilibrium strategies satisfy  $T_i^* = 2J_i$  and  $W_i^* \geq 0$  arbitrary,  $i = 1, 2$ . (See Table 2.3 for an example.)
- Large differences in job sizes (e.g.  $J_1 = 1$ ,  $J_2 = 30$ ): without loss of generality assume  $J_1 < J_2$ , then equilibrium strategies satisfy  $T_1^* = 2J_1$ ,  $T_2^* \in [f_2, 2J_2]$  with  $J_2 < f_2 < 2J_2$  and  $W_i^* \geq 0$  arbitrary. (See Table 2.4 for an example.)

Figure 2.1 shows upper and lower bounds on the equilibrium values of  $T_2$  as a function of the differences in job size. At first, when job sizes are similar customer 2 will be as patient as possible,  $T_2 = 2J_2$ , as noted above. As the differences in job size grow, the lower bound reaches  $1.5J_2$  while the upper bound remains at  $2J_2$ . Both bounds are determined by means of a 95% confidence interval.

The simulation result for similar job sizes can be explained as follows. Setting  $T_i = 2J_i$  implies that customer  $i$  is as patient as possible. Upon doing so, any download attempt will be successful since any download lasts for a time of at most  $2J_i$ . The customer never has to abort a download and consequently never has to wait for a time  $W_i$  before starting a new attempt. Hence, since all download attempts are successful the customer optimizes the number of pages he can read. Setting a waiting time  $W_i$  is superfluous, and hence any waiting time will be optimal.

<i>Cust. 1</i>	$T_2 = 45$	<b>48</b>	<b>51</b>	<b>54</b>	<b>57</b>	<b>58</b>	<b>59</b>	<b>60</b>
$T_1 = 1.80$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	37.92
1.85	0.00	0.00	0.00	0.00	0.00	0.00	0.00	37.29
1.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	29.95
1.95	0.00	0.00	0.00	0.00	0.00	0.00	0.00	39.39
<b>2.00</b>	86.49	<b>86.32</b>	<b>86.62</b>	<b>86.51</b>	<b>86.62</b>	<b>86.13</b>	<b>86.20</b>	<b>86.67</b>

<i>Cust. 2</i>	$T_2 = 45$	<b>48</b>	<b>51</b>	<b>54</b>	<b>57</b>	<b>58</b>	<b>59</b>	<b>60</b>
$T_1 = 1.80$	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.42
1.85	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.41
1.90	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.89
1.95	0.00	0.00	0.00	0.00	0.00	0.00	0.00	2.49
<b>2.00</b>	3.15	<b>3.17</b>	<b>3.18</b>	<b>3.16</b>	<b>3.16</b>	<b>3.17</b>	<b>3.20</b>	<b>3.16</b>

Table 2.4: Simulations with large job differences. *The tables above show the average number of pages downloaded and read by customer 1 (upper table) and customer 2 during a time period of length 300 over 1000 simulation runs. In these simulations  $J_1 = 1$ ,  $J_2 = 30$ ,  $l_1 = 1/2$  and  $l_2 = 1/45$ , and  $W_i = 0$ . All numbers in the tables are rounded to two decimals. The numbers in boldface indicate the equilibria  $(T_1, T_2) = (2J_1, f_2)$  with  $8J_2/5 \leq f_2 \leq 2J_2$ .*

### 3 Extended model with non-congestion related errors

The previous analysis only considered unsuccessful downloads of webpages due to congestion on the network. In this section we extend our model to include non-congestion related errors as a second cause of unsuccessful download requests. Assume that at the beginning of each download attempt such an error takes place with a probability  $p$ . If it occurs, we assume that the download request is completely ignored, that is, to the network it seems as if the responsible customer did not put down the request. After a time  $T_i$  the customer notices that his download request is not fulfilled and hence, he will cancel the request and wait for a time  $W_i$  before putting down a new one.

If action  $e$  denotes the fact that a customer is confronted with a non-congestion related error although he himself is not aware of it, then the initial situations in the download process at time  $t = 0$  are

$$\left\{ \begin{array}{ll} s_i = (T_i, d, T_i/2, T_i, 0), \quad i = 1, 2, & \text{with probability } (1-p)^2, \\ s_1 = (T_1, e, 0, T_1, 0), \quad s_2 = (J_2, d, J_2, T_2, 0) & \text{with probability } p(1-p), \\ s_1 = (J_1, d, J_1, T_1, 0), \quad s_2 = (T_2, e, 0, T_2, 0) & \text{with probability } p(1-p), \\ s_i = (T_i, e, 0, T_i, 0), \quad i = 1, 2, & \text{with probability } p^2. \end{array} \right.$$

As of the next decision moment  $t = \min\{m_1, m_2\}$  the process evolves as described in the Tables 3.1 ( $t = m_i < m_k$ ) and 3.2 ( $t = m_i = m_k$ ). Notice that for a probability  $p = 0$  these processes boil down to those without non-congestion related errors as described in Section 2.



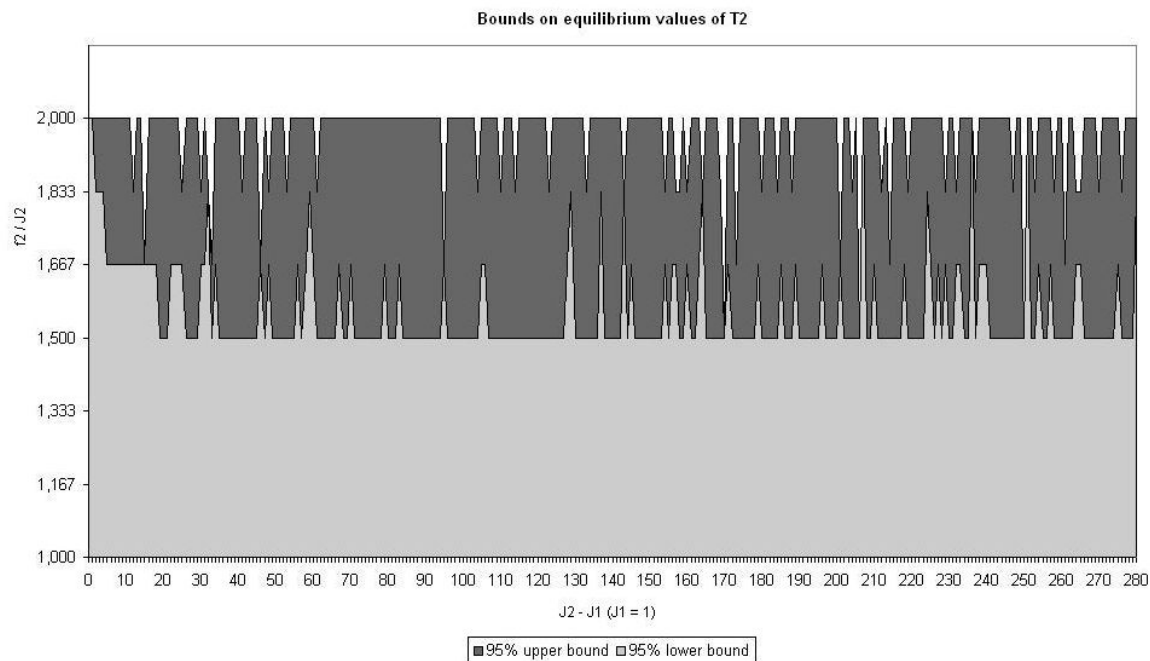


Figure 2.1: Bounds of  $T_2$  as a function of job size differences. *The results shown are the averages over 1000 simulations over a period of length 400 in which  $f_2 \in \{J_2, \frac{7}{6}J_2, \dots, 2J_2\}$ .*

Simulation results with  $p = 0.10$  are as follows:

- Similar job sizes (e.g.  $J_1 = 2, J_2 = 3$ ): a unique equilibrium strategy  $T_i^* = 2J_i$  and  $W_i^* = 0, i = 1, 2$ .
- Small differences in job sizes (e.g.  $J_1 = 1, J_2 = 6$ ): without loss of generality assume  $J_1 < J_2$ , then equilibrium strategies satisfy  $T_1^* = 2J_1, T_2^* \in [f_2, 2J_2]$  with  $J_2 < f_2 < 2J_2$  and  $W_i^* = 0$ .
- Large differences in job sizes (e.g.  $J_1 = 1, J_2 = 50$ ): assume  $J_1 < J_2$ , then equilibrium strategies satisfy  $T_1^* = 2J_1, T_2^* \in [f_{2l}, f_{2h}]$  with  $J_2 < f_{2l} < f_{2h} < 2J_2$  and  $W_i^* = 0$ .

Notice that in all equilibria the customer with the smallest job size, say customer 1, will be as patient as possible,  $T_1^* = 2J_1$ . Further notice that none of the customers waits after canceling an unsuccessful download request. Both of them choose for immediate reattempts, which has a negative effect on the network congestion. These restarts seems logical since a customer may conclude from an unsuccessful download request under  $T_i = 2J_i$  that it was caused by a non-congestion related error. Therefore, it makes no sense to wait and the customer chooses for an immediate repetition of the download request. Hence, under the presence of non-congestion related errors all equilibrium strategies are restart strategies.

$a_i$	$a_k$	new $s_i$	new $s_k$
$d$	$d, e, r, w$	$(t + W_i, w, q_i, t_i, n_i)$ if $J_i > q_i$ , $(t + R_i, r, q_i, t_i, n_i)$ else	$(m_k, a_k, q_k, t_k, n_k)$ if $a_k \neq d$ , $(J_k + m_k/2 - q_k + t/2, d, J_k, t_k, n_k)$ else if $J_k + m_k/2 - q_k + t/2 \leq t_k$ , $(t_k, d, q_k - m_k/2 + t_k - t/2, t_k, n_k)$ else
$e$	$d, e, r, w$	$(t + W_i, w, 0, t_i, n_i)$	$(m_k, a_k, q_k, t_k, n_k)$
$r, w$	$d, e, r, w$	with probability $p$ : $(t + T_i, e, 0, t + T_i, n_i + \delta_i)$ with probability $1 - p$ : $(t + J_i, d, J_i, t + T_i, n_i + \delta_i)$ if $a_k \neq d$ , $(t + T_i, d, T_i/2, t + T_i, n_i + \delta_i)$ else	$(m_k, a_k, q_k, t_k, n_k)$ if $a_k \neq d$ , $(2J_k + 2m_k - 2q_k - t, d, J_k, t_k, n_k)$ else if $2J_k + 2m_k - 2q_k - t \leq t_k$ , $(t_k, d, q_k - m_k + t_k/2 + t/2, t_k, n_k)$ else

Table 3.1: *Change of situations at time  $t = m_i < m_k$  in the presence of non-congestion related errors.*

## 4 Efficiency of network usage

Motivated by the results in the previous section we investigate now whether the equilibrium outcomes  $n_i(T^*, W^*)$  of both customers can be improved upon. That is, is it possible for both customers to reach outcomes larger than  $n_i(T^*, W^*)$  in a framework different from the previously-used noncooperative game? To answer this question, we consider a social welfare framework, which considers the utility of the society, the two-customer network. This utility depends on both customers' strategies and in its optimum the usage of the network will be at least as efficient as under individual optimization. Define the social welfare  $S$  as follows (cf. [3]):

$$S(T, W) = \frac{1}{1 - \alpha} \sum_{i=1}^2 (n_i(T, W))^{1-\alpha}.$$

The goal is to optimize the social welfare  $S(T, W)$  for a given number  $\alpha$ . The above formulation of the social welfare includes several well-known utility optimization criteria. For example, for  $\alpha \rightarrow 0$  the total number of pages read is maximized; for  $\alpha \rightarrow 1$  proportional fairness (maximize  $\sum_{i=1}^2 \log(n_i(T, W))$ ) is obtained; for  $\alpha \rightarrow 2$  the potential loss is minimized; and finally, for  $\alpha \rightarrow \infty$  we obtain max-min fairness.

$a_i$	$a_k$	new $s_i$	new $s_k$
$d, e$	$d, e$	$(t + R_i, r, q_i, t_i, n_i)$ if $J_i = q_i$ , $(t + W_i, w, q_i, t_i, n_i)$ if $J_i \neq q_i$ or $a_i = e$	$(t + R_k, r, q_k, t_k, n_k)$ if $J_k = q_k$ , $(t + W_k, w, q_k, t_k, n_k)$ if $J_k \neq q_k$ or $a_k = e$
$d, e$	$r, w$	$(t + R_i, r, q_i, t_i, n_i)$ if $J_i = q_i$ , $(t + W_i, w, q_i, t_i, n_i)$ if $J_i \neq q_i$ or $a_i = e$	with probability $1 - p$ : $(t + J_k, d, J_k, t + T_k, n_k + \delta_k)$ with probability $p$ : $(t + T_k, e, 0, t + T_k, n_k + \delta_k)$
$r, w$	$r, w$	with probability $(1 - p)^2$ : $(t + T_i, d, T_i/2, t + T_i, n_i + \delta_i)$   $(t + T_k, d, T_k/2, t + T_k, n_k + \delta_k)$	
		with probability $p(1 - p)$ : $(t + T_i, e, 0, t + T_i, n_i + \delta_i)$   $(t + J_k, d, J_k, t + T_k, n_k + \delta_k)$	
		with probability $p(1 - p)$ : $(t + J_i, d, J_i, t + T_i, n_i + \delta_i)$   $(t + T_k, e, 0, t + T_k, n_k + \delta_k)$	
		with probability $p^2$ : $(t + T_i, e, 0, t + T_i, n_i + \delta_i)$   $(t + T_k, e, 0, t + T_k, n_k + \delta_k)$	

Table 3.2: *Change of situations at time  $t = m_i = m_k$  in the presence of non-congestion related errors.*

To implement the social optimal outcome  $(\hat{T}^\alpha, \hat{W}^\alpha)$ , we assume that each of the customers pays a price  $z$  per download request. As a consequence, each customer's optimization problem changes from  $\max_{T_i, W_i} n_i(T, W)$  to  $\max_{T_i, W_i} u_i(T, W)$  with

$$u_i(T, W) = n_i(T, W) - z(n_i(T, W) + \bar{n}_i(T, W)),$$

where  $\bar{n}_i(T, W)$  is the number of unsuccessful download requests of customer  $i$ . We say that the customers can implement the social optimal outcome if each of them optimizes his utility  $u_i(T, W)$  by selecting  $(\hat{T}_i^\alpha, \hat{W}_i^\alpha)$  for a given price  $z$ . That is, it is a Nash equilibrium:  $u_i(\hat{T}^\alpha, \hat{W}^\alpha) \geq u_i(T_i, W_i, \hat{T}_k^\alpha, \hat{W}_k^\alpha)$  for all strategies  $(T_i, W_i)$ .

In our two-customer network it turns out that different values of  $\alpha$  have little influence on the social optimal strategies of the customers. Further, all social optimal strategies are implementable by the customers, that is, they are Nash equilibria. Since these equilibria have the same structure as the equilibria in the previous section, the efficiency of the network usage under social welfare is the same as under the previously studied individual optimization. Hence, social welfare cannot improve upon the efficiency of the network usage.

The following causes underly these results. First, notice that

$$u_i(T, W) = (1 - z)n_i(T, W) - z\bar{n}_i(T, W),$$

and maximizing the number of pages read  $n_i(T, W)$ , or the number of successful download requests, is equivalent to minimizing the number of unsuccessful download requests  $\bar{n}_i(T, W)$ . Hence, if the price  $z$  is not too high,  $z < 1$ , then maximizing  $u_i(T, W)$  boils down to maximizing  $n_i(T, W)$ . This explains why the social optimal strategies have the same structure as the Nash equilibria under individual optimization. Second, a Nash equilibrium is a kind of max-min fair solution. Hence, it corresponds to the social optimal strategy for  $\alpha \rightarrow \infty$ . Together with the first cause, this motivates the first result that the social optima are approximately equal for different values of  $\alpha$ . Third, all strategies are stationary over time by assumption. This restricts the strategy space of the customers. If we would allow for non-stationary strategies then one may expect to achieve efficiency improvements, but this is hard to analyze due to the stochastic reading times. On the other hand, stationary strategies are easier to use in practice than non-stationary ones and they go well with the stationary stochastic reading times.

## 5 Concluding remarks

In this paper we studied interactions of two customers on a network. Both customers repeatedly issue download requests. At first we considered a model in which congestion is the only cause of unsuccessful requests. It turned out that in all equilibrium strategies of the customers the length of the waiting times can be chosen arbitrarily. Hence, restart strategies, which have a waiting time of length zero, are among the equilibrium strategies. Hereafter the model was extended to include non-congestion related errors as a second cause of failed download attempts. Now, in all equilibrium strategies the optimal waiting times have a length of zero. Hence, all equilibrium strategies are restart strategies since the customer will immediately reissue his request after canceling it. Both the basic and the extended model confirm the optimality of restart strategies, as in [2].

Our results depend strongly on the fact that there are just two customers using the network. For example, consider in our extended model a very patient customer with impatience threshold  $T_i = 2J_i$ . If this customer is confronted with an unsuccessful download request then he knows for sure that this is caused by a non-congestion related error. If there are 3 customers then the analysis becomes considerably more complicated. First, the number of different cases due to different job sizes increases considerably. Second, it need not be the case that  $T_i = 3J_i$  is optimal in case of similar job sizes and congestion being the only cause of failed attempts. For example, if at all times at most 2 of the 3 users use the network then  $T_i = 2J_i$  would lead to a larger number of successful downloads. Further, if  $J_i$  is large enough then customers need not be willing to wait for a time of  $3J_i$ . More general, in case of  $n$  customers using the network, a customer would have to wait for a time of  $nJ_i$  before knowing for sure the cause of an unsuccessful request. But this seems very unlikely since impatience thresholds will be lower than  $nJ_i$  if  $n$  is large enough. Therefore, the results will change if more than 2

customers are considered. Hence, an interesting extension of this study is to investigate what happens to the optimality of restart strategies when the number of network users grows. It seems very likely that the optimality of restart strategies will fail to hold and that positive waiting times will be optimal since the customers are no longer able to deduce the cause of their failed attempts.

## References

- [1] Chalasani, P., Jha, S., Shehory, O., Sycara, K.: Query restart strategies for Web agents, *Proceedings of the second international conference on Autonomous agents*, ACM Press, New York, 1998, 124–131.
- [2] Maurer, S.M., Huberman, B.A.: Restart strategies and Internet congestion, *Journal of Economic Dynamics & Control*, Vol. 25 (2001), 641–654.
- [3] Mo, J., Walrand, J.: Fair end-to-end window-based congestion control, *IEEE/ACM Transactions on networking*, Vol. 8 (2000), 556–567.
- [4] Nash, J.: Non-cooperative games. *Annals of Mathematics*, Vol. 54 (1951), 286–295.
- [5] Schroeder, M., Boro, L., McCann, J.: Efficiency improvements for interactions of web-agents, *Proceedings of the fifth international conference on Autonomous agents* ACM Press, New York, 2001, 112–113.
- [6] Van Moorsel, A.P.A., Wolter, K.: Analysis and algorithms for restart, *Proceedings of the First International Conference on Quantitative Evaluation of Systems (QEST'04)*, IEEE Computer Society, 2004, 195–204.