Analysis and Performance of Complete Homogeneous Communication Networks

Franco Robledo, Pablo Rodríguez-Bocca, Pablo Romero
Facultad de Ingeniería. Universidad de la República
Julio Herrera y Reissig 565
Montevideo, Uruguay
Email: (frobledo,prbocca,promero)@fing.edu.uy

***Abstract*—In this paper we address a fundamental problem in communication systems. A fully-connected system is modelled by a complete graph, where all nodes have identical capacities. A message is owned by a singleton. If he/she decides to forward the message simultaneously to several nodes, he/she will take longer (exactly, the number of simultaneous nodes times a single forwarding scheme). The only rule in this communication system is that a message can be forwarded by a node only if it fully known. The *makespan* is the completion time, precisely when the message is fully known by all nodes. The *average waiting time* is the average among the completion time of all individual nodes. The problem under study is to select the communication strategy that minimizes both the makespan and average waiting time. Intuition and current design of real networks say that *one-to-many* systems should perform better than *one-to-one* systems, however this is not usually true. A previous study claims that a sequential or *one-to-one* forwarding scheme minimizes the average waiting time, but they do not offer a proof. Here, a formal proof is included. Furthermore, we show that the sequential strategy minimizes the makespan as well. The paper is closed with comments on potential applications in scheduling of parallel machines, content delivery networks, peer-to-peer systems and rumour spreading.**

Keywords*—*Communication System, Makespan, Waiting Time.

I. MOTIVATION

The Internet is supported by the client-server architecture, where users connect with a specific server to download files. This architecture has some benefits. The service is both simple and highly predictable. However, the server infrastructure is not scalable when demand is increased. A natural idea to overcome this scalability issue is to consider content popularity, where the most popular contents can be shared by the users. The server invites users to communicate and offer those files which are normally replicated in the network. An abstraction of this concept is accomplished with peer-to-peer systems (P2P for short). They are self-organized virtual communities developed on the Internet Infrastructure, where users, called peers, share resources (content, bandwidth, CPU-time, memory) to others, basically because they have common interests. From a game-theoretic point of view, cooperation is better than competition. From an engineering point of view, we understand that the power of user-cooperation in P2P systems is maximized, but the real-life design is jeopardized by other factors. Indeed, broadband resources are better exploited with cooperation. The altruistic behaviour in P2P networks is achieved with incentives, using a *give-to-get* concept [1], [2]. Nevertheless, the design of a resilient P2P network has several challenges. Most important are: the Internet access infrastructure is usually asymmetric, hindering peer exchange; peers arrive and depart the system when they wish [3]; some peers (called free-riders) exploit network resources but do not contribute with the system; a failure in the underlying network usually damage the P2P service; there is an explicit trade-off between the full knowledge of the network (topology, peers resources) and payload, which directly impacts in the throughput and network performance.

The main purpose of this paper is to understand how business models, heterogeneous users and non-altruistic behaviour can damage the best cooperative scenario. If we could come back to history, we would ask ourselves which is the best communication system. This is a key question we address in this paper, and we answer it in the context of a fully-connected homogeneous system with completely altruistic users. Even though we motivate this paper by P2P systems, our main result apply to several communication systems: scheduling in parallel unrelated machines, social networks and content delivery networks, among many others.

Our work is inspired by a fundamental problem posed for the first time by Qiu and Srikant, where they state that *it should be clear that a good strategy* is the one-to-one forwarding scheme [4]. Even though the authors study the service capacity of a file sharing peer-to-peer system, its formulation is general enough. For practical purposes they find a closed formula for the average waiting time following a one-to-one forwarding scheme when the population N is a power of two. In [5], a formal proof that the one-to-one forwarding scheme achieves the minimum waiting time is included, when the population is a power of two. Here, we formally prove that it is not only good, but also optimal, for both makespan and waiting time measures. The result holds for an arbitrary population size. We remark that the optimum forwarding scheme rarely appears in real-life systems. We discuss this phenomenon showing the gap between this theoretical result and real-life implementations of communication systems. The main contributions of this paper are two-fold:

1) The best solution of a fundamental communication system is introduced, with a formal proof.
2) A discussion of the gap between the best theoretical forwarding scheme and real-life implementations is provided.

This paper is organized as follows. Section II presents the evolution of cooperative systems as a realistic framework.

From this section, it should be clear that P2P networks fully disseminate files in a one-to-many fashion, in strong contrast with our main result. The theoretical problem under study is presented in Section III. The mathematical analysis provides a full solution of the problem, which is derived in Section IV. The gap between theory and real-life applications is considered in Section V. Section VI contains the main conclusions and trends for future work.

II. Cooperative Systems and Business

Napster was the first popular and centralized peer-to-peer network [6]. Users were able to download MP3 music files and many others. The Recording Industry Association of America (RIAA) highly criticized these peer-to-peer networks complaining that the sales went down. Definitely, digital music associations were right, and Napster ended in the thumb after a legal process. Clearly, the hierarchical structure of Napster avoided a key element in the design of peer-to-peer networks: the *anonymity*. However, the new paradigm woke-up the imagination of platform providers and users as well. Gnutella arrived with a very different concept. Now, all peers are clients and servers at the same time, normally called servents [7]. These peers find neighbours with elementary primitives of communication, and look for new contents via flooding. This completely distributed architecture showed its strength to disseminate popular contents, via the cooperation of neighbours. However, flooding was not effective for rare contents. To make things worse, the names could incidentally crash for different files, users were not forced to cooperate, and malicious peers could even conspire with one peer, disturbing its neighbours. Many other structured network overlays were deployed in the beginnings of this century: Emule/Edonkey [8] and BitTorrent [2] for instance. BitTorrent, created by Bram Cohen, is an unstructured network overlay designed for fast distribution and replication of media contents. The new concept is inspired in incentives: *give to get*. The tit-for-tat solution of a game theory problem (the Iterated Dilemmas Prisoner) was included in this new design philosophy, and promotes an altruistic behavior of players [9]. Several current implementations of P2P networks are BitTorrent based, even for bandwidth sensitive applications such as live-streaming [10].

It is worth to remark that the forwarding scheme in real-life pee-to-peer applications follows a one-to-many strategy. Furthermore, many other communication systems (social networks, cellular systems) cooperate with a mesh-like virtual topology. This is in strong contrast with the best theoretical communication scheme assuming an altruistic fully-connected scenario, as the following sections confirm.

III. Problem

We are given N peers with identical capacity b (in bits per second) and a message with size M (measured in bits). A singleton has the message, and at time $t_1 = 0$ he/she forwards the message to one or to many other peers. Let us denote $\tau = M/b$ the time-slot following one-to-one forwarding time. If he/she decides to forward the message to c nodes, it will take $c\tau$ seconds to perform the forwarding task. Let us denote $0 = t_1 \leq t_2 \leq \ldots \leq t_N$ the corresponding completion times of the N peers in this cooperative system. The *makespan* is t_N, while the *average waiting time*, $\bar{t}$, is the average over the set $\{t_1, \ldots, t_N\}$. Clearly, $\bar{t} \leq t_N$. In a *one-to-many* forwarding scheme, every peer selects a fixed number c of peers to forward the message. In general, in a simultaneous forwarding scheme there is some peer i that, at time t_i, simultaneously forwards the message to more than one peer. In contrast, the only remaining strategy is a sequential or *one-to-one* forwarding strategy.

Here we formally prove that the one-to-one forwarding strategy is optimal, in the sense that it minimizes both the makespan and the average waiting time simultaneously. It is worth to remark that only elementary analysis is considered. For short, we will use $n = \lceil log_2(N) \rceil$ and $n_c = \lceil log_c(N) \rceil$.

IV. Solution

A straightforward calculation provides the makespan and average waiting time in the one-to-one forwarding scheme:

Lemma 1: The makespan in the one-to-one forwarding scheme is $n\tau$.

Proof: By definition of the one-to-one forwarding scheme, there is one peer at time $T_0 = 0$, and there are 2^{i-1} peers whose completion time is $T_i = i\tau$. Therefore, the message is fully owned by 2^i peers at time $i\tau$, for $i = 1, \ldots, n-1$. The remaining $N - 2^{n-1}$ peers receive the message at time $n\tau$. ∎

Lemma 2: The average waiting time in the one-to-one forwarding scheme is $\bar{t} = \frac{\tau}{N}(nN - 2^n + 1)$.

Proof:

$$\begin{aligned}\bar{t} &= \frac{1}{N}[\sum_{i=1}^{n-1} 2^{i-1} i\tau + (N - 2^{n-1})\tau] \\ &= \frac{\tau}{N}[(n2^{n-1} - 2^n + 1) + (N - 2^{n-1})n] \\ &= \frac{\tau}{N}(nN - 2^n + 1).\end{aligned}$$

∎

Since $n = \lceil log_2(N) \rceil$, we conclude that both the makespan and average waiting time grow logarithmically with the population of the system and linearly with respect to the time-slot τ when the one-to-one strategy is considered. Let us contrast the result with a one-to-many strategy in what follows, where each peer forwards the message to $c-1$ different peers, for some $c > 2$.

Lemma 3: The makespan in the one-to-many forwarding scheme of type $c-1$ is $n_c(c-1)\tau$.

Proof: By definition of the one-to-many forwarding scheme of type $c-1$, there are $N_i = c^{i-1}(c-1)$ peers whose completion time is $T_i = i(c-1)\tau$, and one peer at time $T_0 = 0$. Therefore, the message is fully owned by c^i peers at time T_i, for $i = 1, \ldots, n_c - 1$. The remaining $N - c^{n_c-1}$ peers receive the message at time $T_{n_c} = n_c(c-1)\tau$. ∎

Lemma 4: The average waiting time in a one-to-many forwarding scheme of type $c-1$ is $\bar{t}_{c-1} = \frac{\tau}{N}[n_c(c-1)N - c^{n_c} + 1]$.

Proof:

$$\begin{aligned}\bar{t}_{c-1} &= \frac{1}{N}[\sum_{i=1}^{n_c-1} N_i T_i + (N - c^{n_c-1})T_{n_c} \\ &= \frac{1}{N}[\sum_{i=1}^{n_c-1} c^{i-1}(c-1)i(c-1)\tau + (N - c^{n_c-1})T_{n_c}] \\ &= \frac{\tau}{N}[n_c c^{n_c-1}(c-1) - (c^{n_c}-1) + (N - c^{n_c-1})T_{n_c}] \\ &= \frac{\tau}{N}[n_c(c-1)N - c^{n_c} + 1].\end{aligned}$$

■

We can check that $\bar{t}_{c-1}$ equals $\bar{t}$ when $c = 2$, as expected. In fact, the one-to-one strategy is the one-to-many if $c-1 = 1$. On the other hand, we will see that $\bar{t}_{c-1} > \bar{t}$ for every $c > 2$. The makespan is studied first:

Lemma 5: The makespan in the one-to-one strategy is never greater than in the one-to-many strategy.

Proof: If $c = 2$ we see that $n_2 = n$, so $(c-1)n_c = n$. It suffices to prove that $(c-1)n_c \geq n$ for any $c \geq 3$, being $n = \lceil log_2(N) \rceil$ and $n_c = \lceil log_c(N) \rceil$:

$$\begin{aligned}(c-1)n_c \geq log_c(N^{c-1}) &= (c-1)log_c(2)log_2(N) \\ &= log_c(2^{c-1})log_2(N) > log_2(N);\end{aligned}$$

where the last inequality follows from the fact that $2^{c-1} > c$ whenever $c \geq 3$. Since $(c-1)n_c$ is an integer, we obtain that $(c-1)n_c \geq \lceil log_2(N) \rceil$, and the result follows. ■

A technical lemma will be used in the main result:

Lemma 6: Given two partitions of $N = \sum_{i=1}^{m} x_i = \sum_{i=1}^{m} y_i$ such that $x_i \geq y_i \geq 0, \forall i = 1, \ldots, m-1$ and $0 \leq x_m < y_m$. Consider an arrange of times $0 \leq t_1 \leq t_2 \leq \ldots \leq t_m$, and a partition for each t_i, $t_i = \sum_{j=1}^{m_i} t_{ij}$, where $0 \leq t_{ij} \leq t_i$. Given any related partition of x_i, $x_i = \sum_{j=1}^{m_i} x_{ij}$, then $\overline{W}_x = \frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{m_i} x_{ij}t_{ij}$ is strictly lower than $\overline{W}_y = \frac{1}{N}\sum_{i=1}^{m} y_i t_i$.

Proof:

$$\begin{aligned}\overline{W}_x &= \frac{1}{N}\sum_{i=1}^{m}\sum_{j=1}^{m_i} x_{ij}t_{ij} < \frac{1}{N}\sum_{i=1}^{m} x_i t_i \\ &= \frac{1}{N}(\sum_{i=1}^{m-1} x_i t_i + x_m t_m) \\ &= \frac{1}{N}(\sum_{i=1}^{m-1}(x_i - y_i)t_i + \sum_{i=1}^{m-1} y_i t_i - y_m t_m + x_m t_m) \\ &= \overline{W}_y + \sum_{i=1}^{m-1}(x_i - y_i)t_i - (y_m - x_m)t_m \\ &= \overline{W}_y + \sum_{i=1}^{m-1}(x_i - y_i)t_i - (\sum_{i=1}^{m}(x_i - y_i))t_m < \overline{W}_y.\end{aligned}$$

■

In words, if more peers own the message at any time t_i using strategy x instead of y ($x_i \geq y_i, i = 1, \ldots, m-1$) and the population is constant (N is constant, so $x_m < y_m$), then x outperforms y in terms of average waiting time.

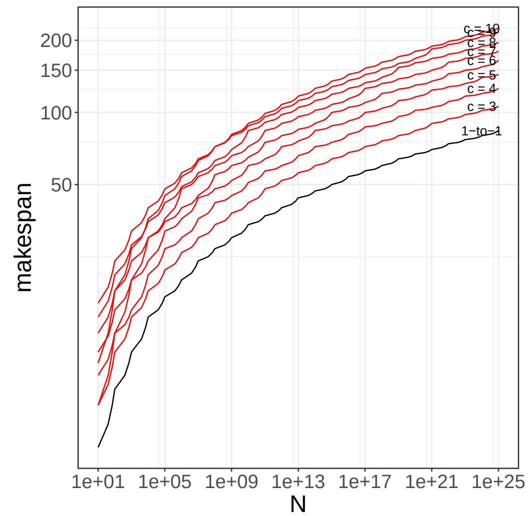

Fig. 1. Makespan for different strategies.

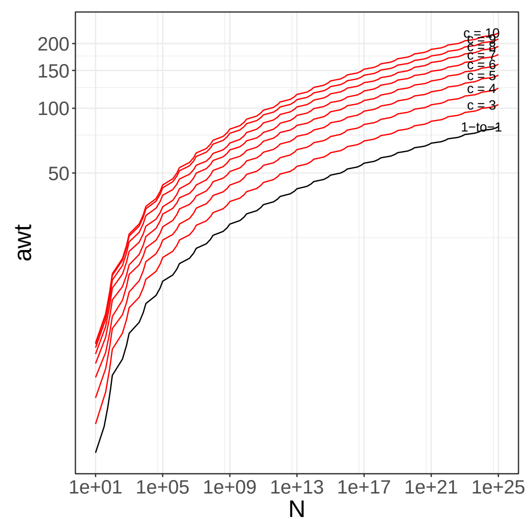

Fig. 2. Average waiting time for different strategies.

Lemma 7: The average waiting time in the one-to-one strategy is never greater than in the one-to-many strategy.

Proof: When $t_i = i(c-1)\tau$ we know that c^i peers have the message using the one-to-many forwarding scheme of type $c-1$ versus $2^{i(c-1)}$ using the one-to-one forwarding scheme. Following Lemma 6, it suffices to prove that $2^{i(c-1)} \geq c^i$ whenever $c \geq 3$. Taking logarithms in both sides it is equivalent to show that $c - 1 \geq log_2(c)$, which is true. ■

In order to illustrate the previous results, Figures 1 and 2 present the makespan and average waiting times respectively for one-to-one and one-to-many strategies in representative cases.

Lemma 8 (Local Replacement): If we are given a strategy where some peer x forwards the message to k new peers in a given time-slot $[t, t+T]$, and there exists an alternative strategy where x forwards the message to $k' > k$ peers in the same time-slot, then the local replacement for the alternative strategy in x reduces both the makespan and average waiting time if all the k' nodes behave as in the original strategy.

Proof: During the specific time-slot $[t, t+T]$, the message is fully owned by more peers. By Lemma 6, the local replacement has lower average waiting time. Analogously, there are more successors of x, so they feed more peers and the makespan is lower as well. ∎

Theorem 1 (Main Result): The one-to-one forwarding scheme is optimal for both makespan and average waiting time.

Proof: If some peer deliberately produces a positive delay in the forwarding, there is a corresponding shift in both makespan and average waiting time. Therefore, delays are not included in an optimal strategy. If some peer x forwards the message to $c-1>1$ nodes, we can consider a local replacement into the one-to-one strategy for x. By Lemmas 5 and 7, the one-to-one forwarding scheme offers lower makespan and average waiting times. By Lemma 8, a local replacement improves both measures. A local replacement is conducted in every node that forwards the message to many nodes. The result is a one-to-one forwarding scheme. ∎

V. Discussion

As far as we know, the problem was posed for the first time by Xiangying Yang and Gustavo de Veciana [4]. The authors study the service capacity of a file sharing peer-to-peer system, and the problem under study serves as a fluid model for replication. They literally state that *it should be clear that a good strategy* is the one-to-one forwarding scheme. For practical purposes they find a closed formula for the average waiting time following a one-to-one forwarding scheme when N is a power of two. In [5], a formal proof of Lemma 7 is provided when the population is a power of two. Here, we formally prove that it is not only good, but also optimal, for both makespan and waiting time measures. The result holds for an arbitrary population size.

Theorem 1 is counterintuitive, and could be used in several fields of knowledge. For instance, the earliest-finish-time in the context of parallel computing systems is precisely our makespan, and forwarding strategies are identified with a formal scheduling on this machines [11], [12]. The main goal in a Content Delivery Network is to minimize the delivery time, which is strictly related with makespan and average waiting time [13], [14], [15]. The time needed to distribute information in a social network, or a virus by an epidemic, are one of the main factors studied in these disciplines [16], [17].

Several real networks use one-to-many forwarding schemes. This suggests that in practice at least one assumption does not hold. First, we remark that full connectivity holds in overlay networks, but does not hold in most real-life scenarios, such as social networks. Second, there is no matching between modelling and reality when identical capacity is assumed. Last but not least, in an information-centric network the behaviour of nodes could be affected with information. As future work, we would like to have a better understanding on the gaps between modelling and real-life systems. Observe that we considered cooperative systems as our framework and terminology, but our theoretical analysis holds under our (quite restricted) assumptions.

VI. Conclusions and Trends for Future Work

In this paper we show that under fully symmetric complete scenarios a one-to-one forwarding scheme provides both the lowest makespan and average waiting time. The goodness of this strategy was suggested by previous authors in the context of peer-to-peer systems for average waiting times. This fundamental result has potential applications to scheduling in parallel unrelated machines, social networks and content delivery networks, among many other complex systems. As a future work, we would like to extend our analysis to incomplete graphs with heterogeneous and dynamic nodes. Observe that the optimum forwarding scheme is completely deterministic. As future work, we would like to better understand the gap between the theoretical predictions from this paper and real-life applications such as social networks and cellular systems.

References

[1] B. Yu and M. P. Singh, *Incentive Mechanisms for Peer-to-Peer Systems*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, pp. 77–88.

[2] B. Cohen, "Incentives Build Robustness in BitTorrent," *www.bramcohen.com*, vol. 1, pp. 1–5, May 2003.

[3] Y. Zhou, L. Chen, C. Yang, and D. M. Chiu, "Video popularity dynamics and its implication for replication," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1273–1285, Aug 2015.

[4] X. Yang and G. de Veciana, "Service Capacity of Peer to Peer Networks," in *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM'04)*, vol. 4, 2004, pp. 2242–2252.

[5] P. Romero, "Mathematical Analysis of Scheduling Policies in Peer-to-Peer video streaming networks," Ph.D. dissertation, Universidad de la República, Montevideo, Uruguay, November 2012.

[6] B. Carlsson and R. Gustavsson, "The Rise and Fall of Napster - An Evolutionary Approach," in *AMT '01: Proceedings of the 6th International Computer Science Conference on Active Media Technology*. London, UK: Springer-Verlag, 2001, pp. 347–354.

[7] E. Adar and B. A. Huberman, "Free riding on gnutella," *First Monday*, vol. 5, no. 10, sep 2000.

[8] A. Puig-centelles, O. Ripolles, and M. Chover, "P2P Traffic Measurements on the Emule System," in *Proceedings of the International Conference on Telecommunications, Networks and Systems 2007 (MCCSIS 2007)*. London, UK: IADIS Digital Library, 2007, pp. 51–58.

[9] R. Axelrod, *The Evolution of Cooperation*. New York, NY, USA: Basic Books, 1984.

[10] A. Barrios, M. Barrios, D. D. Vera, P. Rodríguez-Bocca, and C. Rostagnol, "GoalBit: A Free and Open Source Peer-to-Peer Streaming Network," in *Proceedings of the 19th International Conference on Multimedia (ACM MULTIMEDIA 2011), Open Source Software Competition*. New York, NY, USA: ACM, November 2011, pp. 727–730.

[11] T. L. Casavant and J. G. Kuhl, "A taxonomy of scheduling in general-purpose distributed computing systems," *IEEE Transactions on Software Engineering*, vol. 14, no. 2, pp. 141–154, Feb 1988.

[12] H. Topcuoglu, S. Hariri, and M.-Y. Wu, "Performance-effective and low-complexity task scheduling for heterogeneous computing," *IEEE Transactions on Parallel and Distributed Systems*, vol. 13, no. 3, pp. 260–274, Mar 2002.

[13] D. Qiu and R. Srikant, "Modeling and performance analysis of bittorrent-like peer-to-peer networks," in *Proceedings of the 2004 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, ser. SIGCOMM '04. New York, NY, USA: ACM, 2004, pp. 367–378.

[14] Y. Liu, Y. Guo, and C. Liang, "A survey on peer-to-peer video streaming systems," *Peer-to-Peer Networking and Applications*, vol. 1, no. 1, pp. 18–28, 2008.

[15] A. Vakali and G. Pallis, "Content delivery networks: status and trends," *IEEE Internet Computing*, vol. 7, no. 6, pp. 68–74, Nov 2003.

[16] K. Saito, R. Nakano, and M. Kimura, "Prediction of information diffusion probabilities for independent cascade model," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*. Springer, 2008, pp. 67–75.

[17] D. J. Daley, J. Gani, and J. M. Gani, *Epidemic modelling: an introduction*. Cambridge University Press, 2001, vol. 15.