How to cheat BitTorrent and why nobody does

David Hales **Simon Patarin**

Technical Report UBLCS-2005-12

May 2005

Department of Computer Science
University of Bologna
Mura Anteo Zamboni 7
40127 Bologna (Italy)

The University of Bologna Department of Computer Science Research Technical Reports are available in PDF and gzipped PostScript formats via anonymous FTP from the area `ftp.cs.unibo.it:/pub/TR/UBLCS` or via WWW at URL `http://www.cs.unibo.it/`. Plain-text abstracts organized by year are available in the directory `ABSTRACTS`.

Recent Titles from the UBLCS Technical Report Series

2004-11 *Mapping XQuery to Algebraic Expressions*, Magnani, M., Montesi, D., June 2004.

2004-12 *VDE: Virtual Distributed Ethernet*, Davoli, R., June 2004.

2004-13 *SchemaPath: Extending XML Schema for Co-Constraints*, Marinelli, P., Sacerdoti Coen, C., Vitali, F., June 2004.

2004-14 *Intelligent Web Servers as Agents*, Gaspari, M., Dragoni, N. Guidi, D., July 2004.

2004-15 *SIR: a Model of Social Reputation*, Mezzetti, N., October 2004.

2004-16 *Algorithms for Large Directed CARP Instances: Urban Solid Waste Collection Operational Support*, Maniezzo, V., October 2004.

2004-17 *Supporting e-Commerce Systems Formalization with Choreography Languages*, Bravetti, M., Guidi, C., Lucchi, R., Zavattaro, G., November 2004.

2004-18 *Decentralized Ranking in Large-Scale Overlay Networks*, Montresor, A., Jelasity, M., Babaoglu, O., December 2004.

2004-19 *Advanced Collective Communication in WDM Optical Rings*, Margara, L., Simon, J., Vassura, V., December 2004.

2005-1 *ARTIS: Design and Implementation of an Adaptive Middleware for Parallel and Distributed Simulation (Ph.D. Thesis)*, D'Angelo, G., March 2005.

2005-2 *Analysis and Prototype of a Metamodeling Environment for Engineering Grid Services (Ph.D. Thesis)*, Moretti, R., March 2005.

2005-3 *On some combinatorial optimization problems arising from computer networks (Ph.D. Thesis)*, Vassura, M., March 2005.

2005-4 *Experiences with Synthetic Network Emulation for Complex IP based Networks (Ph.D. Thesis)*, Cacciaguerra, S., March 2005.

2005-5 *Interactivity Maintenance for Event Synchronization in Massive Multiplayer Online Games (Ph.D. Thesis)*, Ferretti, S., March 2005.

2005-6 *Reasoning with preferences over temporal, uncertain, and conditional statements (Ph.D. Thesis)*, Venable, K. B., March 2005.

2005-7 *Whole Platform (Ph.D. Thesis)*, Solmi, R., March 2005.

2005-8 *Loss Functions and Structured Domains for Support Vector Machines (Ph.D. Thesis)*, Portera, F., March 2005.

2005-9 *A Reasoning Infrastructure to Support Cooperation of Intelligent Agents on the Semantic Grid*, Dragoni, N., Gaspari, M., Guidi, D., April 2005.

2005-10 *Fault Tolerant Knowledge Level Communication in Open Asynchronous Multi-Agent Systems*, Dragoni, N., Gaspari, M., April 2005.

2005-11 *The AEDSS Application Ontology: Enhanced Automatic Assessment of EDSS in Multiple Sclerosis*, Gaspari, M., Saletti, N., Scandellari, C., Stecchi, S., April 2005.

How to cheat BitTorrent and why nobody does[1]

David Hales[2] **Simon Patarin**[2]

Technical Report UBLCS-2005-12

May 2005

Abstract

The BitTorrent peer-to-peer file-sharing system attempts to build robustness to free-riding by implementing a tit-for-tat-like strategy within its protocol. It is often believed that this strategy alone is responsible for the the high-levels of cooperation found within the BitTorrent system. However, we highlight some of the weaknesses of the approach and indicate where it would be easy to cheat and free-ride. Given that cheating of this kind currently appears rare, this motivates the question: why is the system not dominated by free-riders?

We advance a hypothesis which argues that BitTorrent may resist free-riders in a way that has not been previously fully comprehended. Ironically, this process relies on what is commonly believed to be a weakness *of BitTorrent - the lack of meta-data search. One consequence of this is to partition the BitTorrent network into numerous isolated swarms - often with several independent swarms for an identical file - which is one of the necessary conditions for a kind of evolutionary group selective process, a process that has been recently identified in similar simulated systems.*

A further implication of the hypothesis is that, given the choice, users may choose unconditional altruism rather than the more restrictive reciprocal tit-for-tat approach as a result of the same group selective process.

1. Partially supported by the EU within the 6th Framework Programme under contract 001907 “Dynamically Evolving, Large Scale Information Systems” (DELIS).

2. Dept. of Computer Science, The University of Bologna, Mura Anteo Zamboni 7, 40127 Bologna, Italy.

1 Introduction

BitTorrent[3] is currently "king" of the popular file-sharing clients. Some reports have claimed that it accounts for the majority of peer-to-peer traffic on the internet others that Hollywood is doomed [10, 13]. Bram Cohen, the original inventor of BitTorrent, is now under high demand on the invited talks circuit.

However, leaving the media attention aside for one moment, we ask the question "why is Bittorent so popular". File-sharing systems come and go, Napster, GNUteller, Kazza, e-Donkey. What makes one more popular than another? The key to understanding this, we believe, is to understand the quality of the user perceived experience when running such clients. Essentially, if a user feels they are getting what they want from the client then they will tend to stick with it and perhaps even influence others to try it. So what do users want from a client? This, of course, is a very complex question. It is certainly the case that users want quick access to files that they desire but there are also other factors such as ease-of-use and social or psychological needs. For example, the inclusion of a chat option to talk to those with files of common interest may add value for some users. The feeling of being part of an online "community", often of a counter cultural hue, may satisfy certain social and psychological needs of users even when the impression is often illusory [9].

In this paper we argue, that by structuring the entire BitTorrent population into disconnected torrents (or swarms) and leaving the meta-data search outside of the BitTorrent system this may have the side-effect of promoting peer altruism. We argue that it may be this side-effect rather than the often cited "tit-for-tat" strategy that is responsible for discouraging free-riders on BitTorrent.

In the following sections we first discuss how the BitTorrent protocol works in practice and then discuss the "tit-for-tat" approach in general. We highlight the weaknesses inherent in both. We then advance our hypothesis concerning what might be stopping the exploitation of these weaknesses.

Finally we conclude with a general discussion concerning our hypothesis and future work that might test it.

2 How BitTorrent works

BitTorrent works by groups of users (called swarms) with an interest in downloading a single specific file (be this an mp3, dvd or executable file) coordinating and cooperating to speed-up the process. Cohen describes how this works in some detail in his paper [3]. Here we give enough relevant details to allow us to develop our hypothesis.

To release a file on the BitTorrent network, one needs to create a specific description file (commonly called a "torrent file") which contains the necessary information for clients to prepare the download and join the swarm. For our concerns, the main information stored in this file is the address of the "tracker".

Each swarm is managed by a "tracker": a centralized process which keeps track of all peers interested in a specific file (or group of files). The tracker does not host any actual contents, but only meta-data about it. Interactions between peers and the tracker are of two kinds: 1) a peer asks the tracker about other peers, and 2) peers keep the tracker updated with their own status regarding the download of the file. When asked for peers, a tracker will return a random list of other peers currently in the swarm. It uses the status information provided by the peers to maintain an up-to-date list. As the number of peers in a single swarm may become very large for popular files, the size of the returned list is usually bound (a maximum of 50 peers is typical).

Once a client has obtained a list of other peers, it will contact them to try to fetch the data it is looking for. In BitTorrent, file contents is split into small-sized pieces and each client maintains the list of the pieces it holds. After a handshake, peers exchange their piece lists so that each of them may determine whether the other has some pieces they are interested in obtaining.

3. see http://bittorrent.com

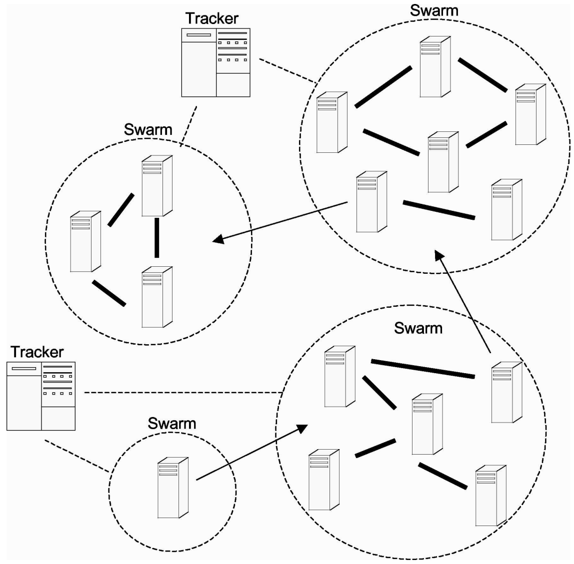

Figure 1. A schematic of a portion of a BitTorrent system. The trackers support swarms of peers each downloading the same file from each other. Thick lines indicate file data. They are constantly in flux due to the application of the TFT-like "choking" protocol. Trackers store references to each peer in each supported swarms. It is not unknown for trackers to support thousands of swarms and for swarms to contain hundreds of peers. Arrows show how peers might through user intervention move between swarms. Generally at least one peer in each swarm would be a "seeder" (usually more, but sometimes, of course, none).

The bandwidth being a limited resource, a single client cannot serve every peer interested in pieces it holds at the same time. The maximum number of peers served concurrently (i.e. the number of available slots) is configurable by the user. All other peers connected to a client (whether they are interested or not) which are *not* being served are said to *choked*. In consequence, each client implements an algorithm to choose which peers to choke and un-choke among those connected to him over time. The strategy proposed by BitTorrent is named "tit-for-tat", meaning that a client will preferably cooperate with the peers cooperating with him. Practically, this means that each client measures how fast it can download from each peer and, in turn, will serve those from whom it has the better download rates. This strategy is implemented for all but one slot which is attributed to an interested client, regardless of its upload rate. This so-called "optimistic unchoking" allows for the discovery of better peers than those currently selected (i.e. those with higher upload rates). This strategy, however, if implemented strictly, would considerably slow down the insertion of newcomers into a running swarm as, they obviously do not have anything to share at the beginning. Thus, clients that have nothing to share are given three time more chances to be selected by the optimistic unchoke. When a client has finished downloading a file it no longer has a download rate from other peers but it can still share (upload) pieces of the file. In this case the choking algorithm is applied by considering upload rate instead. Peers are selected based on how fast they can be uploaded to. This spreads the file faster. Such ″seeder″ peers that store the whole file are very important to the functioning of a swarm. If a swarm contains no seeders it may lead to a situation in which pieces of the file are missing from the swarm as a whole. In this sense the system requires at least some level of altruistic behaviour from ″seeders″. This requirement is evidence by the matra often repeated on BitTorrent websites: *leave your download running for a little while after you've got the entire file.* Figure 1 shows a rough schematic of some of the features of BitTorrent.

3 Tit-for-Tat is not the "best strategy"

It is generally believed that BitTorrent maintains high levels of cooperation because it uses a protocol based on the tit-for-tat (TFT) game strategy - as championed in Axelrod's, now classic, book "the evolution of cooperation" [2]. Axelrod held computer tournaments in which programs, submitted by different researchers, repeatedly played the canonical game of cooperation - the Prisoner's Dilemma (PD).

3.1 The Prisoner's Dilemma

In the iterated PD game two players each play a number of rounds. In each round each selects a move from two alternatives (C or D) then the round ends and each player receives a score (or pay-off). Figure 2 shows a so-called 'pay-off matrix' for the game. If both choose the 'cooperate' move then both get a 'reward' — the score R. If both select the 'defect' move they are 'punished' — they get the score P. If one player defects and the other cooperates then the defector gets T (the 'temptation' score), the other getting S (the 'sucker' score). When these pay-offs, which are numbers representing some kind of desirable utility (for example, money), obey the following constraints: $T > R > P > S$ and $2R > T + S$ then we say the game represents a Prisoner's Dilemma (PD). When both players cooperate this represents maximising of the collective good but when one player defects and another cooperates this represents a form of free-riding. The defector gains a higher score (the temptation) at the expense of the co-operator (who then becomes the 'sucker').

	Cooperate	Defect
Cooperate	R, R	S, T
Defect	T, S	P, P

Figure 2. A payoff matrix for the two-player single round Prisoner's Dilemma (PD) game. Given $T > R > P > S \wedge 2R > T + S$ the Nash equilibrium is for both players to select Defect but both selecting Cooperate would produce higher social and individual returns. However, if either player selects Cooperate they are exposed to Defection by their opponent — hence the dilemma

3.2 Tit-for-tat does well but can always be bettered

Axelrod found that in a round-robin tournament the TFT strategy did best on average against the other submitted strategies. The TFT strategy is very simple, it starts by selecting a cooperative move and then for subsequent moves copies the last move made by its opponent. So if a TFT player meets another TFT player they cooperate all the time. However, if TFT meets another player who behaves "selfishly" and defects, then it can be punished with defection in future interactions. The so-called "shadow of the future" can therefore effect behaviour in the present. Cooperation resulting from TFT-like behaviour has been termed "reciprocal altruism" [16] and has been used to explain various animal and human cooperative behaviour.

However, what is less often mentioned is that Axelrod's work does not prove or demonstrate that TFT is the best strategy or that it can not be bettered by other less cooperative strategies. As stated by Axelrod [2] - *no strategy is best irrespective of the other strategies in the population*. This makes intuitive sense when we appreciate that the performance of a particular strategy in a population is a social phenomena not an individual one. To give an extreme example, if all players in a population always defected and we inserted a single TFT player, then that player would obviously do worse than all the other players since it would initially cooperate with each new player it met and consequently be suckered!

This lesson was starkly demonstrated in the recently organised 20th anniversary IPD computer tournaments [5] in which a strategy based on collusion with other submitted strategies controversially outperformed TFT. By submitting numerous strategies that acted as "masters" and "slaves", using a set of initial game interactions to signal which was which, the Southampton team could outperform cooperative TFT interactions by allowing the "masters" to constantly sucker the “slaves”. The slave strategies performed very badly of course, but their sacrifice boosted the “masters” to much higher scores than TFT could ever obtain. Although this result caused controversy, as some claimed this was no in the “spirit of the game”, it is a great example of the fact that, as Axelrod stated, there can never be a “best strategy” independent of the population of other strategies.

4 The “shadow of the future”

The TFT strategy works, as we discussed above, by punishing bad behaviour in the future: *if you cheat me today, I will cheat you tomorrow*. If you short-change your regular newsagent today, he might charge you double tomorrow. Given that individuals know there is a good chance of interacting in the future, it is often rational to not cheat in the present. This kind of reasoning has been termed the “shadow of the future” [2].

This appears intuitive in human and animal systems since individuals are incredibly good at recognising individuals in the future. When I go into my newsagent I know that I am dealing with the same person I was yesterday, and the chances are so does the shop owner. It’s costly and often impractical to try to pretend to be someone else in face-to-face interactions. However, this is not the case in automated online interactions between many kinds of peer clients over a network.

The technique of fixing a non-fakeable identity is a big issue in security for distributed online systems therefore. Where the stakes are high enough, when security is essential, then trusted third parties can be used to issue certificates of identity that can be verified - this is a mechanism operated by many online systems.

However, within distributed and open systems where certification would be costly or impractical other ad-hoc methods are used to attempt to bind an identify to a peer. Obviously, if these mechanism are violated then it is comparatively easy for one peer to fool other peers into believing that they are a different peer from the one they met before. A single peer can therefore collect multiple identities - often termed a “Sybil Attack” [4] and therefore escape the “shadow of the future”.

5 Faking Identity in BitTorrent

In the context of BitTorrent, identity is signalled to the tracker and to other peers using a 20-byte string. A unique identity is generated by the client for each swarm it participates in. This identity is used for every interaction with the tracker and is sent to other peers during the handshake at the beginning of each connection. The tracker, when it returns a list of peers, sends the identity of each of them, in addition to the address and port to connect to.

When a client initiates a connection, it checks whether the identity it receives in the handshake matches the one it has obtained from the tracker. If this is not the case, it drops the connection. The recipient of the connection, however, cannot perform this sort of checking as it has only a limited view of the peers in the swarm. Moreover, trackers do not provide an interface to perform online identity checks.

A client that would want to fake its own identity could do it very easily. As long as trackers do not allow online identity checks (based on IP addresses and ports), it is sufficient to have one identity to interact with the tracker and with the connecting peers, plus one distinct identity for every other peer it connects to. Given the current liberalism of the current implementations it is not even necessary to remember the identity used with a given peer and the (fake) identity could be created randomly for each new outgoing connection. In the eventuality where trackers and (legitimate) clients would become more cautious, it would be necessary to open a new port for each identity used and register to the tracker using these parameters. As long as trackers support clients connecting from behind a firewall, this subterfuge cannot be defeated.

6 Why does BitTorrent work?

Given that, currently at least, BitTorrent is "king" of the file-sharers and yet, it would seem, is so easy to cheat, why does it work? Also, as stated previously, the system relies to some extent on the pure altruism of "seeders" who have nothing to gain from continuing to serve the file. Why is there so much cooperation going on? Why doesn't selfish behaviour swamp the system? One obvious answer is that people are simply more cooperative and altruistic than a worst case kind of economic rationality would suggest - that people don't always act selfishly when they could. This is certainly true and has been demonstrated in experimental settings [12] and identified even in real competitive markets [8]. But there seems little reason why such behaviour would preferentially attach to only the BitTorrent client. This kind of altruistic behaviour would benefit other systems too - like e-Donkey etc. But we know that selfishness in such systems is very high [1]. So why does BitTorrent appear to be doing so well?

As we have discussed above, the claim that it is due solely to the TFT-like interaction protocol can not be the full explanation since it can not account for the pure altruism of seeders, isn't the best strategy in any case and can easily be subverted with a little hacking of the client.

In the following sections we sketch an alternative theory that might explain the phenomena. Firstly, we need to discuss very briefly the way BitTorrent handles meta-data.

6.1 Leaving Meta-Data to the Users

One aspect that place BitTorrent apart from many other file-sharing systems is the way that meta-information concerning content is distributed. BitTorrent, infact, does not distribute meta-information at all. In order to download a particular file using BitTorrent the user must supply the details of the specific file which are included in a .torrent file. How the user gets this .torrent file is not a concern of the BitTorrent client.

Hence, in order to get .torrent files the user must use other mechanisms - such as the web, where sites exist that run trackers and list active torrent files. There are many such websites (the now defunct Supernova.org became very popular in its day) catering to different tastes and audiences. Some are subscription only, not available to the public, most contain active message boards and appear to support a kind of user community in which people ask favours from others (such as re-seeding a swarm). Users can also, e-mail torrents they have found between their social networks.

6.2 Torrents as Tribes

This "leave it to the user" approach to meta-data is sometimes considered a weakness of BitTorrent, however, *we hypothesise this is actually a key strength of the system and helps to support altruism and cooperation.*

It does this in two ways. Firstly, by fostering cooperative in-groups of like-minded users with common interests and isolating them, to some extent, from casual users finding meta-data with simple queries - to find and register onto some torrent websites is time-consuming and requires users who know what they are looking for so reducing the casual user (who may be less altruistic). Secondly, and more significantly, each individual torrent swarm is logically isolated from all other torrent swarms, even those sharing the same files running on different trackers.

What this latter point means is this: if a less than cooperative client, a bad guy, enters a swarm then that swarm will, on the whole, perform less well than other swarms composed of all good-guys. It is true that the bad guy will do well at a cost to the other swarm members. However, users may decide to manually leave the swarm if they feel they are not getting a good enough performance from it . Almost all BitTorrent clients show the user the upload vs. download ratios for the swarms they are members of thus allowing them to monitor the quality of the service they are receiving from the swarm. Anecdotally, it is not infrequent for users to remove themselves from swarms that deliver a poor ratio.

This latter process, if indeed practised often enough by enough users, would lead to a kind of user driven group selection process. This process has already been observed and simulated in the context of human and peer-to-peer systems [7, 14, 15, 6]. We categorise such systems as

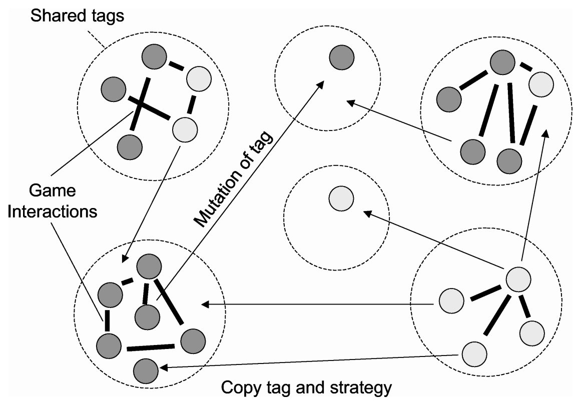

Figure 3. A schematic of the process of tribal formation using so-called "tags" from [7]. This group selective process selects for altruistic behaviour in both the domain of the the abstract Prisoner's Dilemma game and a simulated file-sharing system [6]. The small shaded circles represent peers, the thick lines game interactions and the arrows possible moves that peers can make. Peers are shaded to indicate their strategy - either selfish or altruistic.

"tribal systems" since they promote cooperation by the dynamic formation and dissolution of "tribes" or groups of peers. By partitioning the population into a finite set of such tribes, those tribes that offer better performance to their members tend to prosper at the expense of those filled with free-riders or bad guys.

The interesting thing about this process is that it can be driven by purely selfish and greedy behaviour at the peer level but produce high levels of altruism at the "tribe level'.

The best way to think of this is as a kind of recruitment process in which peers wish to increase their own benefit from the system. Assuming peers can move between "tribes' (in our case, exit and enter swarms) they will tend to be attracted to those tribes that can offer them more benefit (in this case, download speed). We would expect this to be impaired by free-loaders. Therefore tribes with fewer freeloaders tend to grow whereas tribes with many tend to shrink - eventually "dying' if all peers leave it. This process has been observed to occur in a number of simulation models in which peers behave in a selfish and greedy way [6]. It is a kind of group selection process which supports high levels of altruistic behaviour and is robust to invasion by free-riders. Figure 3 shows a schematic of the dynamics of this group selective process based on tribes created from "tags' (rather than trackers).

We argue that the current architecture of BitTorrent appears to provide the right selective environment for this process to occur. This may offer a more plausible explanation for the high levels of altruism that appear within the system at present.

6.3 Implications and Hypotheses

If we are correct, this view allows us to make a number of implications or hypotheses that might appear counter-intuitive.

Firstly, if peer users were given the freedom to switch their clients between pure altruism and pure selfishness this may actually improve performance of the system since pure altruism would predominate. In the context of the choking algorithm this would mean the implementation of both a more altruistic sharing rule and a less altruistic rule than the TFT-like one. The group selective process should then select highly altruistic swarms since peers would leave ones full of selfish free-riders.

Secondly, if meta-search systems became fused into the BitTorrent protocol this might have a deleterious effect on altruism and cooperation. Why? Because by allowing all peer clients to quickly locate and simultaneously participate in many related swarms at the same time, the group selective process becomes weakened since the swarms effectively become fused into one large swarm. Additionally, this removal of the need for communities to be created around trackers, may remove some of the psychological advantages.

In both these cases, it would appear that there are so many unknowns that the only way to test these hypotheses would be via empirical experimentation in the ”wild” of the BitTorrent ecology. This, of course, is not impossible since BitTorrent is an open protocol. Modified clients would need to be released and gain some reasonable level of take-up. The data would need to be collected from both clients and trackers.

It would appear that a modified client that tested the first hypothesis would be practical to implement since it requires only modest changes to the choking and peer identification algorithms. This may be the subject of future work.

Testing the second hypothesis would be a much more complex task. However the evolution of the BitTorrent protocol may come to test this for us since there are reports of plans to integrate meta-data search in future versions [17].

There are of course certain kinds of “ethical” issues in experimenting in this way. If we were wrong in our first hypothesis then clients capable of selfish cheating behaviour might come to dominate the system and destroy its usefulness - though we think is very unlikely.

7 Conclusion

We have argued that the success of Bittorent is unlikely to be due purely to the use of the tit-fortat inspired protocol, as is often claimed. We argue that the real driving force behind the high cooperation might be the by-product of the lack of meta-data search within BitTorrent. This results in the creation of a number of disconnected “tribes” at both the swarm and the tracker level. The users are active in the tribal dynamics by selecting those tribes that best satisfy their needs hence tribes filled with free-riders will tend to die out. We compare this to existing simulations of tribal dynamics in both human and peer-to-peer systems.

We advance a number of hypothesis which our theory suggests including the production of both cheating peers and unconditionally altruistic peers. We argue that releasing such peers into the “wild” of the BitTorrent ecology would not damage BitTorrent but could actually increase the system level performance because unconditional altruism would tend to be selected and predominate.

In order to fully test these hypotheses we would need to construct and release such clients into the ”wild” and collect data from them. This may be the subject of future work.

It’s a sobering thought to consider that, perhaps, the most bandwidth hungry applications on the internet today work by complex social mechanisms we don’t yet understand. However, this is less sobering when one realises that since peer-to-peer systems are really just computationally supported human social systems then we should expect the same issues to arise as we observe within human social systems - namely the question “what’s going on?”.

Acknowledgements

This work would not have been possible without perceptive discussions with many people particularly those in the Bologna group including: Mark Jelasity, Ozalp Babaoglu and Alberto Montresor. This work partially supported by the EU within the 6th Framework Program under contract 001907 (DELIS).

References

[1] E. Adar and B. Huberman. Free Riding on Gnutella. *First Monday* Volume 5, No. 10. (2000).

[2] R. Axelrod. *The evolution of cooperation.* N.Y.: Basic Books, (1984).

[3] B. Cohen. Incentives Build Robustness in BitTorrent. Presented at the *1st Workshop on the Economics of Peer-2-Peer Systems*, June 5-6, 2003, Berkley, CA. Available at: http://www.sims.berkeley.edu/research/conferences/p2pecon/

[4] J. Douceur. The sybil attack. *In Proc. of the IPTPS02 Workshop, Cambridge, MA (USA), March 2002.* 2002 Available at: citeseer.ist.psu.edu/douceur02sybil.html

[5] Wendy M. Grossman. New Tack Wins Prisoner's Dilemma. *Wired, Oct. 13, 2004.* Available at: http://www.wired.com/news/culture/0,1284,65317,00.html

[6] D. Hales. From selfish nodes to cooperative networks — emergent link based incentives in peer-to-peer networks. In *Proc. of the 4th IEEE International Conference on Peer-to-Peer Computing (P2P2004)*. IEEE Computer Soc. Press, 2004.

[7] D. Hales. Cooperation without Space or Memory: Tags, Groups and the Prisoner's Dilemma. In *Moss and Davidsson (eds.) Multi-Agent-Based Simulation.* LNAI 1979:157-166. Springer. Berlin. 2000

[8] A. P. Kirman and N. J. Vriend. Evolving Market Structure: An ACE Model of Price Dispersion and Loyalty. *Journal of Economic Dynamics and Control*, 25, Nos. 3/4, 459-502, (2001).

[9] Strahilevitz, Lior.: Charismatic Code, Social Norms, and the Emergence of Cooperation on the File-Swapping Networks. Virginia Law Review, Vol. 89, 2003 Available at: http://ssrn.com/abstract=329700

[10] Thomas Mennecke BitTorrent Remains Powerhouse Network

Slyck News, January 31, 2005. Available at: http://www.slyck.com/news.php?story=649

[11] J. F. Nash. Equilibrium Points in N-Person Games, *Proc. Natl. Acad. Sci.* USA 36, 48-49, (1950).

[12] W. Poundstone. *Prisoner's Dilemma.* Doubleday, New York, 1992

[13] Adam Pasick. File-sharing network thrives beneath the radar Yahoo Technology News, Thursday November 4, 2004. Available at: http://in.tech.yahoo.com/041103/137/2ho4i.html

[14] R. Riolo, M. D. Cohen, R. Axelrod. Cooperation without Reciprocity. *Nature* 414, 441-443, (2001).

[15] K. Sigmund and M. Nowak: Tides of tolerance. *Nature* 414, 403-405, (2001).

[16] R. Tivers. The evolution of reciprocal altruism. *Q. Rev. Biol.* 46, 35-57. 1971

[17] Harnessing BitTorrents Storm. *Red Herring Magazine*, Print Issue, February 21, 2005, Available at: http://www.redherring.com