

A Pragmatic and Scalable Solution for Free Riding Problem in Peer to Peer Networks

Mourad Amad¹, Djamil Aïssani¹,
Ahmed Meddahi², and Abdelmalek Boudries³

Abstract. Peer-to-Peer (P2P) systems are increasingly popular as an alternative to the traditional client-server model for many applications, particularly file sharing. Free-riding is one of the most serious problems encountered in Peer-to-peer (P2P) systems like BitTorrent.

Free riding is a serious problem in P2P networks for both Internet service providers (ISPs) and users. To users, the response time of query increases significantly because a few altruistic peers have to handle too many requests. In P2P file sharing systems, free-riders who use others' resources without sharing their own cause system-wide performance degradation. Therefore, the rational P2P application developers should develop some anti-free riding measures to deal with it.

In this paper, we propose a pragmatic and scalable peer to peer scheme based on a utility function that uses two parameters: collaboration of peers and detection of tentative of free riders. Performance evaluation of the proposed solution shows that results are globally satisfactory.

Keywords: Free riding, Free rider, Peer-to-Peer (P2P) network, Utility function.

Abdelmalek Amine
Otmane Ait Mohamed
Ladjel Bellatreche (Eds.)

Modeling Approaches and Algorithms for Advanced Computer Applications



Springer

Editors

Abdelmalek Amine
Département d'informatique
Saida University
Centre Universitaire Taher Moulay de Saida
Algeria

Ladjet Bellatreche
LIAS/ISAE-ENSMA
Futuroscope Chasseneuil Cedex
France

Otmane Ait Mohamed
Department of Electrical and Computer
Engineering
Concordia University
Montreal
Québec
Canada

ISSN 1860-949X

ISSN 1860-9503 (electronic)

ISBN 978-3-319-00559-1

ISBN 978-3-319-00560-7 (eBook)

DOI 10.1007/978-3-319-00560-7

Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2013937615

© Springer International Publishing Switzerland 2013

This work is subject to copyright. All rights are reserved by the Publisher, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilms or in any other physical way, and transmission or information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed. Exempted from this legal reservation are brief excerpts in connection with reviews or scholarly analysis or material supplied specifically for the purpose of being entered and executed on a computer system, for exclusive use by the purchaser of the work. Duplication of this publication or parts thereof is permitted only under the provisions of the Copyright Law of the Publisher's location, in its current version, and permission for use must always be obtained from Springer. Permissions for use may be obtained through RightsLink at the Copyright Clearance Center. Violations are liable to prosecution under the respective Copyright Law.

The use of general descriptive names, registered names, trademarks, service marks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

While the advice and information in this book are believed to be true and accurate at the date of publication, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

A Pragmatic and Scalable Solution for Free Riding Problem in Peer to Peer Networks

Mourad Amad¹, Djamil Aïssani¹,
Ahmed Meddahi², and Abdelmalek Boudries³

¹ Laboratory LAMOS, University of Bejaia, Algeria
amad.mourad@gmail.com, lamos_bejaia@hotmail.com

² Telecom Lille 1, France
ahmed.meddahi@telecom-lille1.eu

³ University of Bejaia, Algeria
abdelmalek.boudries@yahoo.fr

Abstract. Peer-to-Peer (P2P) systems are increasingly popular as an alternative to the traditional client-server model for many applications, particularly file sharing. Free-riding is one of the most serious problems encountered in Peer-to-peer (P2P) systems like BitTorrent.

Free riding is a serious problem in P2P networks for both Internet service providers (ISPs) and users. To users, the response time of query increases significantly because a few altruistic peers have to handle too many requests. In P2P file sharing systems, free-riders who use others' resources without sharing their own cause system-wide performance degradation. Therefore, the rational P2P application developers should develop some anti-free riding measures to deal with it.

In this paper, we propose a pragmatic and scalable peer to peer scheme based on a utility function that uses two parameters: collaboration of peers and detection of tentative of free riders. Performance evaluation of the proposed solution shows that results are globally satisfactory.

Keywords: Free riding, Free rider, Peer-to-Peer (P2P) network, Utility function.

1 Introduction

The decentralized peer-to-peer model (P2P) of communication has emerged as a viable alternative to the traditional client server based centralized model in the large scale distributed systems. There are mainly three P2P architectures: centralized, decentralized and hybrid. The first architecture distributes the storage services but the resource search process is already centralized in like server. Napster is a representative example of such system. The second architecture is divided on two types; structured and unstructured architecture. Decentralized structured architecture such as HPM [2] is cost effective in terms of routing (*optimized and greedy*). However, the semantic aspect and the maintenance cost are not well considered. Decentralized unstructured architecture is opposite to the

former one, the lookup service is not greedy (*flooding or probabilistic*). However, the topology maintenance and the semantic search are well considered. As an example of these architectures, we can cite: Gnutella . The later architecture (hybrid) consists of a set of super peers organized in decentralized architecture (*structured or unstructured*). Each super peer manages a small network in centralized manner such as in Napster. KaZaad and Skype are representative examples of this architecture.

P2P networks are confronted to many problems: routing performance, semantic search, security aspect, firewall detection and traversal, anonymity of users, free riding, network management performance, etc. In this paper, we study the free riding problem. Free riding means exploiting P2P network resources (*through searching, downloading objects, or using services*) without contributing to the P2P network at desirable levels.

Free-riding problem is not just observed in P2P systems. This type of rational behavior also appears in the context of strategic network formation, selfish routing, mobile ad hoc networks and congestion control. However, the prevalence of free-riding behavior in P2P systems can be attributed to some specific characteristics, including decentralization, high churn rate (*users dynamically joining and leaving the system*), availability of cheap identities (*pseudonyms*), hidden actions (*e.g. nodes not forwarding messages to reduce computing and communication cost*), and collusion (*users forming groups to maximize benefit*) [9]. In other words, free rider peers in P2P networks are already high consumers, and low or never producers.

Within a short time of its release, the Gnutella network found about 75% of its users free-riding, i.e., downloading files from the network without uploading any content [1]. In the same study, they found that only 10% of peers provide almost 90 % of services in the P2P network. Most peers share no more than ten files or some unpopular files, which are not queried or downloaded by others at all [1].

Many proposed free riding solutions have been proposed in the literature. However, most of them are much cost in terms of implementation or suffer from the scalability problem. In this paper, we try to give a new solution to free riding problem in P2P networks, considering both cited obstacles.

The rest of this paper is organized as follow: next section gives a brief overview of peer to peer networks and related works on free riding in such networks. In section 3, we introduce our proposed model to reduce and isolate the free rider peers from our network. The proposed solution is programmatic and scalable. Performance evaluation of the proposed solution is illustrated in section 4. Finally, we conclude and give some perspectives for future works.

2 Background and Related Works

In this section, we give a background on P2P networks, and then we explain the most important solutions that have been given to address the free riding problem in such networks.

2.1 Background on P2P Networks

It is widely believed that the success of P2P file sharing systems depends upon the quality of service (*QoS*) offered by such systems. Many types of applications are investigated by P2P technology such as: file sharing, parallel and distributed computing, collaboration and communication and recently Internet telephony. However, file sharing applications are the most used. Many studies show that the most important part of Internet flows are from P2P file sharing applications. Hybrid P2P architecture is well adapted to file sharing applications. The maintenance cost is reduced and the scalability is considered. However, mobility and time connectivity of super peers are a challenge. In general, we choose peer with low mobility and long time connectivity as a super peer.

For our proposed model against free riding problem, we use the hybrid P2P architecture represented on Figure 1 as administration and interconnexion architecture. In this architecture, we distinguish two types of peers: ordinary peer and super peer. Each set of ordinary peers is managed by a super peer. This later is principally characterized by high speed, long time connectivity and high storage capability.

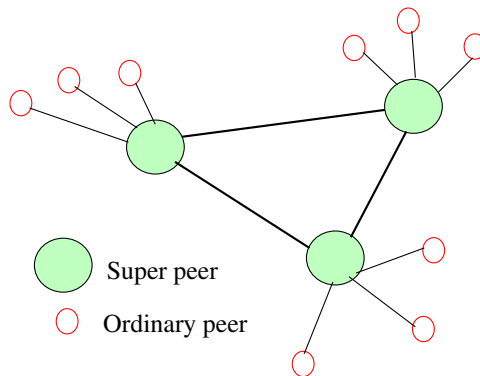


Fig. 1. Hybrid P2P architecture

Each peer (*ordinary peer or super peer*) shares their own resources and can search or request other resources. Ordinary node searches resources by asking its super peer. Super peer searches resources by asking its neighboring peers. Peer that searches resources and does not offering is called free rider.

2.2 Related Works on Free Riding

A number of solutions have been proposed to address the free-riding problem in peer-to-peer systems, particularly for file sharing applications or streaming applications [8]. The authors of [3] propose to introduce a utility function to capture contributions made by a user and an auditing scheme to ensure the integrity of

a utility function's values. The principal problem of this solution is the choice of the utility function. In [7], Give-to-get algorithm has been proposed; it encourages peers to upload data to neighbors which prove to be the best forwarders. This policy hurts free-riders any time their neighbors have the opportunity to forward to well-behaving peers instead. The problem of this solution is the scalability. Like shown on [6], when the peers are behind Nats/Firewalls, it is very difficult to prevent free riding problem. In their paper, the authors prove that it is impossible to prevent free-riding when more than half of the peers are firewalled. In [12], the authors show by experimentation using PlanetLab, a new aspect of the free-riding problem in BitTorrent. When a client obtains a larger than normal view of the BitTorrent swarm, it increases its chances to become unchoked by lechers and to discover seeders. Consequently, it is able to attain good download rates without uploading. A preventive solution of free riders in BitTorrent called reat-Before-Trick is then proposed in [11].

Authors of [13] propose an anti-free riding approach based on migration and workload balancing. The heart of this mechanism is to migrate some shared files from the overloaded peers to the neighboring free riders automatically and transparently, which enforces free riders to offer services when altruistic peers are heavily overloaded. Like in [3], a utility function based scheme is proposed in [10]. In this scheme, three utility factors have been considered: the total number of the shared files, the total size of data shared and the popularity of the data shared. As a consequence, the utility function becomes more efficient as the basic one.

In order to isolate the free rider peers from P2P networks without affecting the scalability of the system, we propose a pragmatic and scalable solution based on utility function with some specific parameters: positive collaboration of each peer by offering resources and signaling tentative of free riders. Most details of the proposed solution are given on section 3.

3 Proposed Solution

The aim of our approach is to encourage resources sharing by forcing each peer to contribute in order to be served. For this, hybrid P2P architecture is used. Peer which offers more services (*resources*) is well served. However, peer egoist is progressively suspected.

3.1 Functional Principle

The most important aspect of the proposed solution is the collaboration between peers. All peers should contribute efficiently in resources sharing. Otherwise, they will be progressively suspected. Notations and parameters used in this paper are illustrated on table 1.

At the join operation, the new node (*peer*) initializes its parameters as follow: $Serv=0$, $Dect=0$, $FR=0$. When this peer gives a positive response to any requestor, the super peer on which is attached increments the corresponding value

Serv. When a peer receives many request from the same requestor, it signal this anomaly to it super peer. This later increments the value of *Dec* corresponding to this peer. After each period *T*, the super peer examines both values *Serv* and *Dect* for each attached peer. For each unchanged couple (*Serv*, *Dect*), the super peer increments the value of *FR*. When *FR* is sufficiently incremented, the corresponding peer will be suspected (*considered as a free rider*).

When a super peer receives a request from neighboring peers, it makes it in the queuing, it serves it when it becomes in the head of queuing, after applying the priority function as shown in table 1.

Table 1. Notations

Symbol	Definition
N	Total number of peers in the network
Serv	Number of positive requests given by the current peer
Dect	Tentatives of free riding signaling by current peer
FR	Status of current peer, it is incremented if during period <i>T</i> , both values <i>Serv</i> and <i>Dect</i> are already the same.
Prior	$Prior=f(Serv, Dect)= Serv+Dect$ is a priority function

Figure 2 shows the functional principle of the proposed anti-free riding scheme.

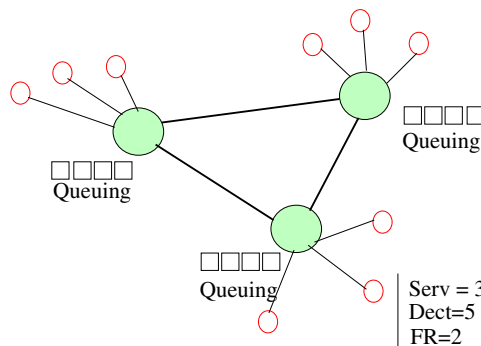


Fig. 2. Functional principle of the proposed scheme

3.2 Scenarios of Communication

Figure 3 shows a scenario of communication between ordinary node and its super peer. If ordinary node does not collaborate by offering resource, their requests will not be satisfied. It will be suspected by its super peer when its parameter *FR* is sufficiently incremented.

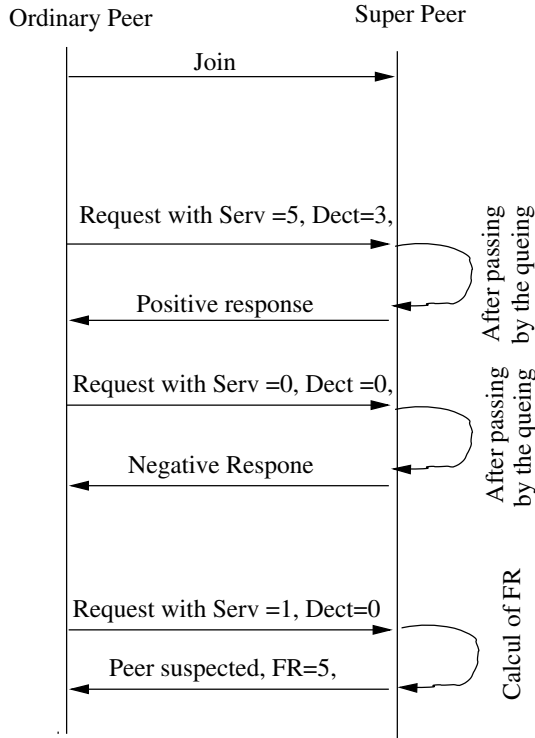


Fig. 3. Scenario of communication between ordinary peer and its super peer

Figure 4 illustrates a scenario of communication between two super peers. The requested super peer forwards the received request to its neighboring super peers if the requestor super peer has shared resources and collaborates positively in the system. Otherwise, its request will have a negative response. It is suspected by a neighboring super peer if its parameter *FR* is sufficiently incremented.

4 Performance Evaluation

In this section, we describe the simulations we performed and the corresponding results. We don't consider the mobility, join and leave when starting simulation.

4.1 Simulation Model

We generate randomly a network with 50 nodes, 7 of them are super nodes. We execute the application in two minutes; we generate randomly requests at arbitrary nodes.

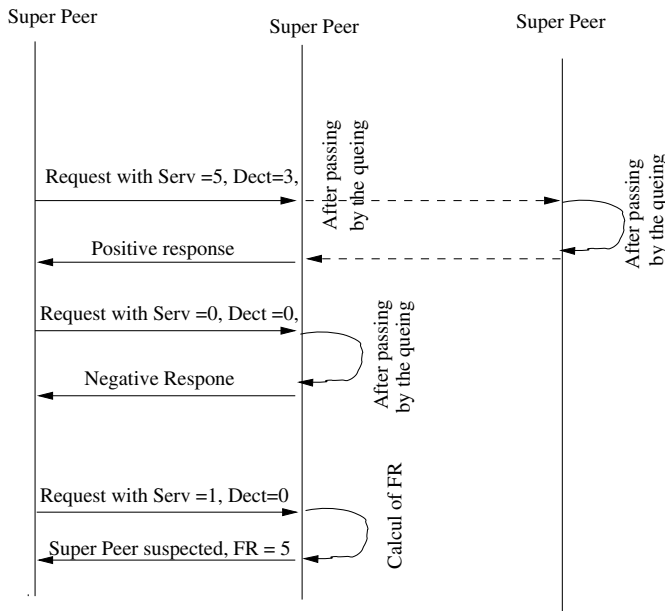


Fig. 4. Scenario of communication between two super peers

4.2 Simulation Parameters and Performance Metrics

As in [4], the following qualitative metrics can be considered for evaluating an anti-free riding scheme:

1. **Quality of service:** QoS parameters may include search time spent in resolving queries, high quality and quantity, and download time.
2. **Availability:** availability of content and services in a P2P network can be increased by forcing free riders to contribute.
3. **Scalability and load sharing:** when the cooperation in a P2P system increased, the load on those peers can be also shared by other peers that would otherwise be free riders. This will help the system to be more scalable where important number of download operations and search queries can be successfully executed on the system.
4. **Robustness:** mechanisms against free riding can make a P2P network more robust against disconnections operation and legal attacks of malicious nodes, which will increase network population in terms of available content and also in terms of number of nodes that are reachable.

As in [5], a number of quantitative parameters can be also considered: the number of downloaded files, the number of unsuccessful downloads, number of uploads, download cost, number of generated messages and fairness.

4.3 Model Implementation

The above described model is implemented using Java programming language. Each peer in the network is represented by a thread. In addition, a main thread is continuously executed, defining the simulation period.

4.4 Experimental Results

Figure 5 shows the detection rate of free riders. When the number of requests increment in the system, the number of detected free riders increments also.

Figure 6 shows the false negative alerts(*the number of peers that have been signaled as free riders but they are not*). When the number of request increments in the system, the number of peer that will be signaled as free riders with error is reduced.

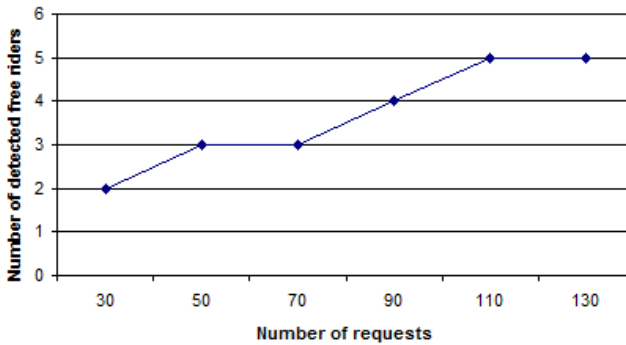


Fig. 5. Free riders detection rate

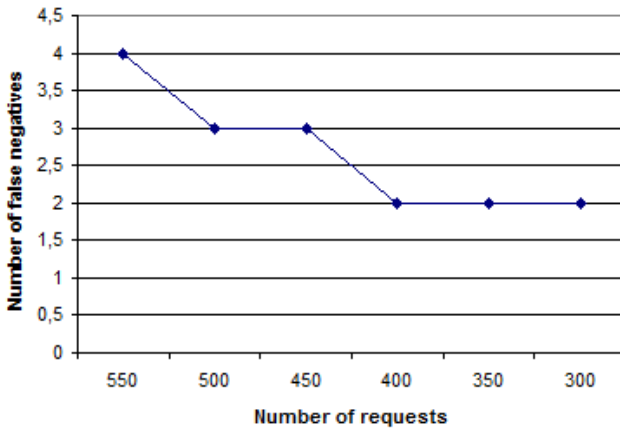


Fig. 6. False negative alerts

5 Conclusion and Perspectives

Free riding is a great challenge to the development and maintenance of Peer-to-Peer (P2P) networks, particularly for file sharing applications.

The free riding problem affects the system in two significant ways. First, the number of files in the system grows very slowly or becomes limited. The number of popular files may become even smaller as the time goes by. This adversely affects user's interest in the system and they eventually pull out of the system. When users who share popular files pull out of the system, the system becomes poorer in terms of the amount of files shared. This is a vicious cycle and it may eventually lead to the collapse of the system. Second, if only a few peers share popular files, all the downloading requests are directed towards those peers. This causes those peers to become hot spots, or exposed to DoS attacks, and overloading their machines and causing congestion on their network. Peers frequently experiencing CPU overloads or network congestion due to the P2P system may exit the system if it affects their other routine activities.

In this paper, we have proposed a new scheme for suspecting free riders from P2P network. The proposed solution is principally based on a specific utility function with priority operation. Peer with high collaboration with upload operation will be good served. Performance evaluation of the new scheme shows that is scalable and easy to implement.

In terms of future work, we envision to generalize and ameliorate our solution in order to take into consideration the free riding problem with two axes: lookup and file sharing.

Acknowledgement. The authors would to thank Dr. Soufiene DJAHEL from Dublin University (*Ireland*) for their constructive remarks.

References

1. Adar, E., Huberman, B.A.: Free Riding on Gnutella. *First Monday* 5(10) (2000)
2. Amad, M., Meddahi, A., Aïssani, D., Zhang, Z.: HPM:A Novel Hierarchical Peer-To-Peer Model for Lookup Acceleration With Provision of Physical Proximity. *Journal of Network and Computer Applications (JNCA)* 35, 1818–1830 (2012)
3. Ham, M., Agha, G.: ARA: A Robust Audit to Prevent Free-Riding in P2P Networks. In: 5th IEEE International Conference on Peer-to-Peer Computing (2005)
4. Karakaya, M., Körpeoglu, I., Ulusoy, Ö.: A Distributed and Measurement-based Framework Against Free Riding in Peer-to-Peer Networks. In: *Peer-to-Peer Computing*, pp. 276–277 (2004)
5. Karakaya, M., Körpeoglu, I., Ulusoy, Ö.: Counteracting free riding in Peer-to-Peer networks. *Computer Networks* 52, 675–694 (2008)
6. Mol, J.J.D., Pouwelse, J.A., Epema, D.H.J., Sips, H.J.: Free-riding, Fairness, and Firewalls in P2P File-Sharing. In: *Eighth International Conference on Peer-to-Peer Computing* (2008)
7. Mol, J.J.D., Pouwelse, J.A., Meulpolder, M., Epema, D.H.J., Sips, H.J.: Give-to-Get: An Algorithm for P2P Video-on-Demand. In: *Proc. SPIE 6818 Multimedia Computing and Networking* (2008)

8. Qin, F., Ge, L., Liu, Q., Liu, J.: Free Riding Analysis of Peer-to-Peer Streaming Systems. *Journal of Computational Information Systems* 7(3), 721–728 (2011)
9. Rahman, M.R.: A Survey of Incentive Mechanisms in Peer-to-Peer Systems. Technical Report CS-2009-22, University of Waterloo
10. Ramaswamy, L., Liu, L.: Free Riding: A New Challenge to Peer-to-Peer File Sharing Systems. In: *HICSS* (2003)
11. Shin, K., Reeves, D.S., Rhee, I.: Treat-Before-Trick: Free-riding Prevention for BitTorrent-like Peer-to-Peer Networks. In: *The 2009 IEEE International Symposium on Parallel and Distributed Processing (IPDPS 2009)*, pp. 1–12 (2009)
12. Sirivianos, M., Park, J.H., Chen, R., Yang, X.: Free-riding in BitTorrent Networks with the Large View Exploit. In: *IPTPS* (2007)
13. Yu, Y., Jin, H.: Discourage free riding in Peer-to-Peer file sharing systems with file migration and workload balancing approach. *Front. Comput. Sci. China* 1(4), 436–449 (2007), doi:10.1007/s11704-007-0042-z

Author Index

- Abdelmalek, Abdelhafid 185
Abdi, Mustapha Kamel 429
Achouri, Abdelghani 37
Aïssani, Djamil 135
Aissaoui, Ouanes 399
Ait-Yakoub, Zina 75
Aliouat, Makhlof 115
Aliouat, Zibouda 115
Aloui, Nadia 273
Amad, Mourad 135
Ameur, Yamine Ait 3
Amine, Abdelmalek 315
Amirat, Abdelkrim 399
Aribi, Nouredine 47
Arous, Mokdad 367
Assoul-Semmar, Yasmina 409
Atil, Fadila 399
Azouaou, Faiçal 419

Beggas, Mounir 227
Benamira, Adel 337
Bendimerad, Nawel 105
Bendjebar, Safia 389
Benferhat, Salem 125
Benkahla, Naima 305
Bensaber, Boucif Amar 37
Bensari, Mouchira 195
Berkani, Lamia 237, 419
Bilami, Azeddine 145, 195
Biskri, Ismaïl 37
Boniol, Frédéric 1
Bouabana-Tebibel, Thouraya 125
Bouagar, Saliha 57
Boudebza, Souâad 419
Boudries, Abdelmalek 135

Boughanem, Mohand 7
Bouju, Alain 249
Boukhalfa, Kamel 285
Bouridane, Ahmed 155
Boussaid, Omar 205
Bouzar-Benlabiod, Lydia 125
Brahim, Farou 27

Chalal, Rachid 205
Chaoui, Allaoua 347
Chebieb, Abdelkrim 3
Cherif, Chahira 429
Chikh, Mohammed Amine 9
Chitroub, Salim 155

Déplanche, Anne-Marie 379
Descoteaux, Steve 37
Djebar, Yacine 259
Djouadi, Yassine 75
Driff, Lydia Nahla 237

El Amine Lazouni, Mohammed 9
El Habib Daho, Mostafa 9

Fareh, Messaouda 205
Feham, Mohammed 185
Foudil, Cherif 347

Gacem, Amina 285
Gargouri, Faïez 273
Graini, Leila 217
Guersi, Nouredine 259
Guessoum, Ahmed 237
Guyennet, Herve 175

- Hafi, Houda 145
 Hamid, Seridi 27
 Hamou-Lhadj, Abdelwahab 5
 Haroun, Smail 17
 Hassini, Abdelatif 305
 Herman, Akdag 27
 Houacine, Amrane 85
 Housseno, Samer 295

 Imedjdouben, Fayçal 85

 Jouis, Christophe 37

 Kaushik, Saroj 357
 Kechar, Bouabdellah 105
 Kelaiaia, Abdessalem 65
 Kerkouche, Elhillali 347
 Khalfaoui, Khaled 347
 Khouas-Outkid, Saliha 409
 Kimour, Mohamed Tahar 259

 Lafifi, Yacine 389
 Laforest, Frederique 227
 Larabi, Slimane 57
 Laskri, Mohamed Tayeb 227
 Lebbah, Yahia 47
 Lehsaini, Mohamed 175
 Louiza, Mansour 165

 Madida, Khuphukile 329
 Mahmoudi, Saïd 9
 Malki, Jamal 249
 Meddahi, Ahmed 135

 Médini, Lionel 227
 Melouah, Ahlem 95
 Meraoumia, Abdallah 155
 Merouani, Hayet Farida 65
 Mokadem, Djelloul 315

 Nouali, Omar 419

 Rachedi, Abdelkrim 329
 Rompré, Louis 37

 Saïdouni, Djamel-Eddine 337, 367
 Samira, Moussaoui 165
 Saouchi, Kaddour 217
 Seghir, Amirouche Nait 17
 Settouti, Nesma 9
 Shehzad, Muhamad Naeem 379
 Simonet, Ana 295
 Simonet, Michel 295
 Slimane, Zohra 185

 Tiwari, Shivendra 357
 Touati, Saïd 17
 Trinquet, Yvon 379

 Urunuela, Richard 379

 Vincent, Cécile 249

 Wannous, Rouaa 249

 Zahra, Fatma Zohra 409

Contents

Invited Talks

New Challenges for Future Avionic Architectures	1
<i>Frédéric Boniol</i>	
Checking System Substitutability: An Application to Interactive Systems	3
<i>Yamine Ait Ameer, Abdelkrim Chebieb</i>	
The Role of Software Tracing in Software Maintenance	5
<i>Abdelwahab Hamou-Lhadj</i>	
Information Retrieval and Social Media	7
<i>Mohand Boughanem</i>	
Track 1: Computational Intelligence (CI)	
Machine Learning Tool for Automatic ASA Detection	9
<i>Mohammed El Amine Lazouni, Mostafa El Habib Daho, Nesma Settouti, Mohammed Amine Chikh, Saïd Mahmoudi</i>	
Stator Faults Detection and Diagnosis in Reactor Coolant Pump Using Kohonen Self-organizing Map	17
<i>Smail Haroun, Amirouche Nait Seghir, Saïd Touati</i>	
A New Approach for the Extraction of Moving Objects	27
<i>Farou Brahim, Seridi Hamid, Akdag Herman</i>	
Seeking for High Level Lexical Association in Texts	37
<i>Ismail Biskri, Louis Rompré, Christophe Jouis, Abdelghani Achouri, Steve Descoteaux, Boucif Amar Bensaber</i>	

Statistical and Constraint Programming Approaches for Parameter Elicitation in Lexicographic Ordering	47
<i>Noureddine Aribi, Yahia Lebbah</i>	
Some Global Measures for Shape Retrieval	57
<i>Saliha Bouagar, Slimane Larabi</i>	
Clustering with Probabilistic Topic Models on Arabic Texts	65
<i>Abdessalem Kelaiaia, Hayet Farida Merouani</i>	
Generating GCI's Axioms from Objects Descriptions in \mathcal{EL}-Description Logics	75
<i>Zina Ait-Yakoub, Yassine Djouadi</i>	
Automatic Phonetization of Arabic Text	85
<i>Fayçal Imedjdouben, Amrane Houacine</i>	
A Novel Region Growing Segmentation Algorithm for Mass Extraction in Mammograms	95
<i>Ahlem Melouah</i>	
Track 2: Security Network Technologies (SNT)	
Coverage Enhancement in Wireless Video-Based Sensor Networks with Rotating Capabilities	105
<i>Nawel Bendimerad, Bouabdellah Kechar</i>	
Improving Wireless Sensor Networks Robustness through Multi-level Fault Tolerant Routing Protocol	115
<i>Zibouda Aliouat, Makhlouf Aliouat</i>	
A New Approach for QCL-Based Alert Correlation Process	125
<i>Lydia Bouzar-Benlabiod, Salem Benferhat, Thouraya Bouabana-Tebibel</i>	
A Pragmatic and Scalable Solution for Free Riding Problem in Peer to Peer Networks	135
<i>Mourad Amad, Djamil Aïssani, Ahmed Meddahi, Abdelmalek Boudries</i>	
Cooperative Strategy to Secure Mobile P2P Network	145
<i>Houda Hafi, Azeddine Bilami</i>	
An Efficient Palmprint Identification System Using Multispectral and Hyperspectral Imaging	155
<i>Abdallah Meraoumia, Salim Chitroub, Ahmed Bouridane</i>	
RDAP: Requested Data Accessibility Protocol for Vehicular Sensor Networks	165
<i>Mansour Louiza, Moussaoui Samira</i>	

Improvement of LEACH for Fault-Tolerance in Sensor Networks	175
<i>Mohamed Lehsaini, Herve Guyennet</i>	
Locally Distributed Handover Decision Making for Seamless Connectivity in Multihomed Moving Networks	185
<i>Zohra Slimane, Abdelhafid Abdelmalek, Mohammed Feham</i>	
A New Hybrid Authentication Protocol to Secure Data Communications in Mobile Networks	195
<i>Mouchira Bensari, Azeddine Bilami</i>	
Track 3: Information Technology (IT)	
Mapping System for Merging Ontologies	205
<i>Messaouda Fareh, Omar Boussaid, Rachid Chalal</i>	
80 Gb/s WDM Communication System Based on Spectral Slicing of Continuum Generating by Chirped Pulse Propagation in Low Normal Dispersion Photonic Crystal Fiber	217
<i>Leila Graini, Kaddour Saouchi</i>	
Fuzzy Logic Based Utility Function for Context-Aware Adaptation Planning	227
<i>Mounir Beggas, Lionel Médini, Frederique Laforest, Mohamed Tayeb Laskri</i>	
Social Validation of Learning Objects in Online Communities of Practice Using Semantic and Machine Learning Techniques	237
<i>Lamia Berkani, Lydia Nahla Driff, Ahmed Guessoum</i>	
Modelling Mobile Object Activities Based on Trajectory Ontology Rules Considering Spatial Relationship Rules	249
<i>Rouaa Wannous, Jamal Malki, Alain Bouju, Cécile Vincent</i>	
A Model-Based on Role for Software Product-Line Evolving Variability	259
<i>Yacine Djebar, Noureddine Guersi, Mohamed Tahar Kimour</i>	
An Approach for the Reuse of Learning Annotations Based on Ontology Techniques	273
<i>Nadia Aloui, Faïez Gargouri</i>	
Very Large Workloads Based Approach to Efficiently Partition Data Warehouses	285
<i>Amina Gacem, Kamel Boukhalifa</i>	

Evaluation of the Influence of Two-Level Clustering with BUB-Trees Indexing on the Optimization of Range Queries	295
<i>Samer Housseno, Ana Simonet, Michel Simonet</i>	
Identification of Terrestrial Vegetation by MSG-SEVIRI Radiometer and Follow-Up of Its Temporal Evolution	305
<i>Naima Benkahla, Abdelatif Hassini</i>	
Automatic Detection of Contours of Circular Geologic Structures on Active Remote Sensing Images Using the Gradient Vector Flow Active Contour	315
<i>Djelloul Mokadem, Abdelmalek Amine</i>	
NALD: Nucleic Acids and Ligands Database	329
<i>Abdelkrim Rachedi, Khuphukile Madida</i>	
Track 4: Computer Systems and Applications (CSA)	
Maximality-Based Labeled Transition Systems Normal Form . . .	337
<i>Adel Benamira, Djamel-Eddine Saïdouni</i>	
Automatic Generation of SPL Structurally Valid Products Using Graph Transformations Approach	347
<i>Khaled Khalfaoui, Allaoua Chaoui, Cherif Foudil, Elhillali Kerkouche</i>	
Modeling On-the-Spot Learning: Storage, Landmarks Weighting Heuristic and Annotation Algorithm	357
<i>Shivendra Tiwari, Saroj Kaushik</i>	
Towards an Integrated Specification and Analysis of Functional and Temporal Properties: Part I: Functional Aspect Verification	367
<i>Mokdad Arous, Djamel-Eddine Saïdouni</i>	
Overhead Control in DP-Fair Work Conserving Real-Time Multiprocessor Scheduling	379
<i>Muhamad Naeem Shehzad, Anne-Marie Déplanche, Yvon Trinquet, Richard Urunuela</i>	
Initializing the Tutor Model Using K-Means Algorithm	389
<i>Safia Bendjebar, Yacine Lafifi</i>	
Towards a Generic Reconfigurable Framework for Self-adaptation of Distributed Component-Based Application . . .	399
<i>Ouanes Aïssaoui, Fadila Atil, Abdelkrim Amirat</i>	

Dynamic Bayesian Networks in Dynamic Reliability and Proposition of a Generic Method for Dynamic Reliability Estimation 409
Fatma Zohra Zahra, Saliha Khouas-Oukid, Yasmina Assoul-Semmar

Semantic Annotations and Context Reasoning to Enhance Knowledge Reuse in e-Learning 419
Souâad Boudebza, Lamia Berkani, Faïçal Azouaou, Omar Nouali

Change Impact Study by Bayesian networks 429
Chahira Cherif, Mustapha Kamel Abdi

Author Index 439

Abstract. Peer-to-Peer (P2P) systems are increasingly popular as an alternative to the traditional client-server model for many applications, particularly file sharing. Free-riding is one of the most serious problems encountered in Peer-to-peer (P2P) systems like BitTorrent.

Free riding is a serious problem in P2P networks for both Internet service providers (ISPs) and users. To users, the response time of query increases significantly because a few altruistic peers have to handle too many requests. In P2P file sharing systems, free-riders who use others' resources without sharing their own cause system-wide performance degradation. Therefore, the rational P2P application developers should develop some anti-free riding measures to deal with it.

In this paper, we propose a pragmatic and scalable peer to peer scheme based on a utility function that uses two parameters: collaboration of peers and detection of tentative of free riders. Performance evaluation of the proposed solution shows that results are globally satisfactory.

Keywords: Free riding, Free rider, Peer-to-Peer (P2P) network, Utility function.

