LEASE: An Economic approach to Leasing data items in Mobile-P2P networks to improve data availability

Anirban Mondal[1] Sanjay Kumar Madria[2] Masaru Kitsuregawa[1]

[1] Institute of Industrial Science
University of Tokyo, JAPAN
{anirban, kitsure}@tkl.iis.u-tokyo.ac.jp

[2] Department of Computer Science
University of Missouri-Rolla, USA
madrias@umr.edu

Abstract. This work proposes LEASE, a novel Mobile-P2P lease-based economic incentive model, in which data requestors need to pay the price (in virtual currency) of their requested data items to data-providers. In LEASE, data-providing mobile peers lease data items to free-riders, who do not have any data items to provide, in lieu of a lease payment. Thus, LEASE not only combats free-riding, but also entices free-riders to host data items, thereby improving network connectivity due to higher peer participation. In essence, LEASE facilitates the collaborative harnessing of limited mobile peer resources for improving data availability. Our performance study shows that LEASE indeed improves query response times and data availability in Mobile-P2P networks.

1 Introduction

In a Mobile Ad-hoc Peer-to-Peer (M-P2P) network, mobile peers (MPs) interact with each other in a peer-to-peer (P2P) fashion. Proliferation of mobile devices (e.g., laptops, PDAs, mobile phones) coupled with the ever-increasing popularity of the P2P paradigm (e.g., Kazaa [11], Gnutella [6]) strongly motivate M-P2P network applications. Mobile devices with support for wireless device-to-device P2P communication are beginning to be deployed such as Microsoft's Zune [9].

M-P2P applications facilitate mobile users in sharing information with each other *on-the-fly* in a P2P manner. A car user could request other car users for information e.g., locations of nearby parking slots and restaurants, and traffic reports a few miles ahead. A pedestrian could request an available taxi nearby his current location. Customers in a shopping mall could share information about the cheapest 'Levis' jeans or swap shopping catalogues. Mobile users could exchange songs or video-clips (as in a future mobile eBay market). Such P2P interactions among mobile users are generally not freely supported by existing wireless communication infrastructures. Our target applications mainly concern slow-moving objects e.g., cars on busy streets, people moving in a market-place or students in a campus.

Data availability in M-P2P networks is typically lower than in fixed networks due to frequent network partitioning arising from user movement and users switching 'on'/'off' their mobile devices. Moreover, a large percentage of MPs typically do not have any data to share with other MPs i.e., they are free-riders [10]. To exacerbate the problem, MPs generally have limited bandwidth, hence a data-providing MP can make available only few of its data items to be shared (i.e., the **shared data items**) based on the amount of bandwidth that it would like to share, but it has additional data items (i.e., the **unshared data items**) in the memory. Given the ephemeral nature of M-P2P environments, unshared data items may *expire* before they can be made available to M-P2P users, which further decreases data availability.

M-P2P data availability could be significantly improved if free-riders could be enticed to pool in their bandwidth resources by hosting unshared data items. Hence, we propose LEASE, a novel lease-based economic incentive model for effective collaborative data sharing among MPs with limited resources. In LEASE, data-providing MPs *lease* data items to those who do not have any data items to provide. A data item d (originally owned by MP P) is said to be **leased** by P to MP H when P provides d to H for a pre-specified lease period τ, in lieu of a lease payment (in *virtual currency*). During the period τ, H hosts d, and after τ expires, H deletes the copy of d at itself. Notably, P may lease d simultaneously to multiple MPs. In case any updates are required to the data (e.g., traffic reports in transportation application scenarios), P sends the updates to H. We shall henceforth refer to a data-providing MP P as a **provide-MP**, and the host MP H as a **host-MP**.

Each data item has a *price* (in *virtual currency*). Data item price depends on access frequency, data quality [13] (e.g., image resolution, audio quality) and the estimated response time for accessing the data item. A query issuing MP pays the *price* of the queried data item to the query-serving MP. Thus, LEASE provides an incentive for free-riding MPs to act as host-MPs so that they can earn **revenue** for issuing their own requests. **Revenue** of an MP is defined as the difference between the amount of virtual currency that it earns (by providing data) and the amount that it spends (by requesting data). Virtual currency is suitable for P2P environments due to high transaction costs of micro-payments in real currency [17]. Secure virtual currency payments have been discussed in [4].

Leasing benefits both provide-MPs and host-MPs. It facilitates a provide-MP in earning revenue from its unshared data items even without hosting them, especially since unshared data items may expire. It helps a host-MP in earning revenue using other MPs' data items. In the absence of a lease model, MPs without any data to provide cannot earn any revenue, thereby decreasing the overall MP participation. In M-P2P networks, leasing is better than *buying* (permanent ownership transfer) since data items have expiry times, hence their value depreciates significantly over time. Moreover, host-MPs wish to host as many 'hot' data items as possible to maximize their revenues.

The main contributions of LEASE follow:

1. Its lease model entices even those users, who have no data to provide, to host data items, thereby improving data availability and MP revenues.
2. Its economic model discourages free-riding, which improves connectivity due to higher peer participation.

Higher peer participation leads to better data availability due to higher available bandwidth and better connectivity. Existing M-P2P replication schemes [8, 18] do not combat free-riding, while M-P2P incentive schemes [19, 20] do not entice free-riders, which have no data, to provide service.

Our performance study indicates that LEASE indeed improves query response times and data availability in M-P2P networks. To our knowledge, this is the first work to propose a lease-based economic model for M-P2P networks.

2 Related Work

Economic models have been discussed in [5, 7, 12] primarily for resource allocation in distributed systems. These works do not address unique M-P2P issues such as frequent network partitioning, mobile resource constraints, free-riding and incentives for peer participation. Incentive mechanisms for static P2P networks have been proposed in [10, 15]. However, pre-defined data access structures (e.g., distributed hash tables [16]) used in static P2P networks assume peers' availability and fixed topology, which makes incentive schemes for static P2P networks too static to be deployed in mobile ad-hoc networks (MANETs). Furthermore, the proposals in [10, 15] do not consider economic models to combat free-riding.

Incentive mechanisms have also been investigated for MANETs [3, 21], the main objective being to encourage an MP in forwarding information to other MPs. However, these works do not consider economic issues and M-P2P architecture. The **E-DCG+** replica allocation approach [8] for MANETs does not consider lease models, incentives and prices of data items. Interestingly, economic ideas for M-P2P networks have been discussed in [19, 20]. However, these works propose opportunistic dissemination of data in M-P2P networks with the aim of reaching as many peers as possible, while we address on-demand data dissemination. The work in [1] proposes an barter-based economic model, but it does not consider M-P2P issues.

In our earlier work [14], we proposed an economic model for data replication based on the price of data items. However, in contrast with this work, the proposal in [14] does not consider a lease-based approach. Moreover, in [14], each peer behaves autonomously without any co-ordination among themselves, while this work considers peer collaboration for improving data availability in M-P2P networks.

3 The LEASE Economic Model

Each provide-MP maintains recent read-write logs (including timestamps) of its own data items as well as details (e.g., lease duration) of the data items that it leases. This information helps provide-MPs to select their respective shared and unshared data items. Each host-MP maintains recent access information of data items based on queries that pass through itself. Such information facilitates host-MPs in selecting data items that they want to host. Available memory space of MPs, bandwidth and data item sizes may vary. We define the **load** of an MP as its job queue length normalized w.r.t. bandwidth.

Table 1 summarizes the notations used in this paper. Using the notations in Table 1, price μ of a data item is computed as follows:

$$\mu = \int_{t_1}^{t_2} \int_{0}^{\delta} (\eta \, dt \times (1/\delta^2) \, d\delta \times DQ \times BA_{M_S}) / J_{M_S,t_j} \qquad (1)$$

Symbol	Significance
d	A given data item
η	Recent Access frequency of d
DQ	Data quality of d
$size$	Size of d
Ex	Time to Expiry time of d
BA_{M_S}	Bandwidth of the query-serving MP for d
J_{M_S,t_j}	Job-queue length of the query-serving MP at time t_j

Table 1. Summary of Notations

where $[t_2 - t_1]$ represents a given time period and δ is the *Euclidean distance* between the query issuing MP M_I and the query serving MP M_S during the time of query issue. For unshared data items, the access frequency η refers to the number of access failures. DQ reflects the quality of data (e.g., image resolution, audio quality) provided by M_S for queries on d. The value of DQ is determined as in our previous work in [13], where we considered three discrete levels of DQ i.e., *high*, *medium* and *low*, their values being 1, 0.5 and 0.25 respectively. As BA_{M_S} increases and δ decreases, μ increases due to faster query response time. As J_{M_S,t_j} increases, μ decreases since M_S's response time for queries on d increases due to higher load.

The revenue earned by an MP M equals ($\sum_{i=1}^{p}(\mu_i \times accs_i)$, where p is the number of data items made available by M, and μ_i and $accs_i$ are the price and access frequency of the i^{th} data item respectively. Similarly, the revenue spent by M equals ($\sum_{i=1}^{q}(\mu_i \times accr_i)$, where q is the number of items queried by M, and μ_i and $accr_i$ are the price and access frequency of M for the i^{th} item respectively.

Role of the provide-MPs and host-MPs

A provide-MP P makes available at itself (i.e., *shares*) data items, with higher revenue-earning potential γ for maximizing its revenue, while leasing out some of its *unshared* data items. Given that $\mu_{i,P}$ is the price of a data item i at P and $acc_{i,P}$ is the recent access frequency of i, γ = $\mu_{i,P} \times acc_{i,P}$. (For data items that P is currently not making available, $acc_{i,P}$ is the number of times a query failed to obtain the data item at P.) P avoids leasing frequently updated data items due to the high communication overhead (e.g., energy, bandwidth) required for maintaining the consistency of such items. Periodically, P broadcasts its list of unshared data items, which have low write frequencies, for finding prospective host-MPs to host these items.

P selects host-MPs by accepting bids for its given data item d based on the quality of service and connectivity of the MPs. P leases d to higher-bidding MPs since MPs with better resources for providing good service would bid higher since they can earn more revenue from d. We define the connectivity of an MP as the number of its one-hop neighbours. P prefers to lease d to MPs with higher connectivity to facilitate it in sharing its data items with as many MPs as possible, thereby enabling it to earn more revenue.

Given an unshared data item d, P decides the number of copies of d to be leased based on the revenue λ that it wishes to obtain from leasing d. P computes λ as follows:

$$\lambda = 0.5 \int_{t_1}^{t_2} (\eta_d \times \mu_d) \tag{2}$$

where $[t_2 - t_1]$ is a given time period, η_d is the number of failed queries on d, and μ_d is the price of d. In Equation 2, the term ($\eta_d \times \mu_d$) reflects P's estimated lost revenue due to not making d available. Observe that λ is 50% of P's estimated lost revenues. Thus, the estimated revenue from leased data items is shared *equally* between the provide-MP and the host-MPs to ensure fairness. Furthermore, allowing the host-MPs to earn 50% of the revenues from d provides adequate incentive for them to host d since they also incur energy and bandwidth-related costs due to downloads of d. Hence, P essentially sums up the bids for d starting from the highest bid until the total value of the bids is greater than or equal to λ. Then, P leases d to the corresponding MPs that made these bids. Notably, unlike existing works, we determine the number of copies based on revenue.

Host-MPs decide which data items to bid for as well as their bid values based on the queries for these items that pass through themselves. A host-MP H bids for data items with higher revenue-earning potential γ for maximizing its revenue. The number of data items for which H bids depends upon its available bandwidth and memory space. Given a data item d, H bids the amount β of currency for d based on d's revenue-earning potential, which depends upon d's popularity, quality, size, estimated expiry time, amount of bandwidth that it would likely make available for d and its current job-queue length. (Recall that d's price at H depends upon H's bandwidth and job-queue length.) Using Table 1 (see Section 3), H computes β as follows:

$$\beta = \int_{t_1}^{t_2} (\eta\, dt \times DQ \times Ex \times BA_{M_S})/(size \times J_{M_S,t_j}) \tag{3}$$

where $[t_2 - t_1]$ represents a given time period. The access frequency η is based on the queries for d that passed through H. A data item *expires* when its access frequency falls below a certain application-dependent threshold. Data items with higher time to expiry facilitate H in earning more revenue by hosting d. Higher bandwidth of H implies better response time for queries on d, while larger job-queue length signifies higher load on H, thereby increasing response time. Smaller-sized data items help H to maximize its revenue per unit of its limited memory space.

Data providers periodically broadcast the unique identifiers of host-MPs, to whom they have leased their data items. Thus, MPs can download *updated* copies of data items from the authorized lease-holders, thereby improving the quality of service. (Provide-MPs send updates only to authorized host-MPs.) In case a host-MP H illegitimately hosts a given data item d or if H continues to host d after its lease period of d has expired, other MPs (e.g., relay MPs through which messages for downloads of d would pass) would inform the corresponding provide-MP P, and P would blacklist H. Periodically, provide-MPs broadcast their list of blacklisted MPs. Blacklisted MPs have to pay double the lease payment the next time they want to lease data items from any provide-MP, which acts as a deterrent.

Host-MPs make the lease payments to provide-MPs at the time of expiry of the lease so that host-MPs can earn revenue from hosting data items before they pay for the lease. This facilitates seamless integration of newly joined MPs, which may initially be unable to make the lease payment. Host-MPs, which fail to make the lease payment at the end of the lease expiry period, are blacklisted, thereby deterring malicious MPs from abusing the leasing system.

Algorithm *LEASE_Provide_MP*
Spc: Its available memory space
(1) Sort all its data items in in descending order of their revenue-earning
potential γ into a list L.
(2) for each data item d in L
/* WF_d is d's write frequency, TH_{WF} is the write frequency threshold */
(3) if ($WF_d < TH_{WF}$)
(4) if ($size_d \leq$ Spc) /* $size_d$ is the size of d */
(5) Fill up its memory space with d
(6) Spc = Spc - $size_d$
(7) if (Spc = = 0) **exit**
(8) Create set CL comprising its **unshared** data items
/* CL is the set of candidate data items for lease */
(9) Broadcast the set CL to its n-hop neighbours
(10) for each data item d in CL
(11) Receive bids from prospective host-MPs, which wish to host d
(12) Arrange the bids in descending order of bid value
(13) Bid_{Sum} = 0
(14) for each bid β from host-MP i
(15) Bid_{Sum} = Bid_{Sum} + β
(16) if $Bid_{Sum} \leq \lambda$
(17) Add i to set $Host_d$
(18) if set $Host_d$ is non-empty
(19) Lease d to the MPs in set $Host_d$ with bid values as lease payment
(20) Initialize set $Host_d$ by making it a NULL set
end

Fig. 1. LEASE algorithm for provide-MP

4 Algorithms in LEASE

Figure 1 depicts the algorithm for a provide-MP P. In line 3, write frequency WF_d of a data item d is computed as (nw_d / τ), where nw_d is the number of writes on d and τ is the lease period. Write frequency threshold TH_{WF} is computed as the average write frequency of all the shared and unshared items in P. In Line 9 of Figure 1, n= 3 or n =4 were found to be reasonable values for our application scenarios (as indicated by preliminary experimental results). In Line 9, P's broadcast message contains the unshared data items and their prices to help prospective host-MPs to determine their bid values. In Lines 14-16, the values of λ and β are computed by Equations 2 and 3 respectively.

Algorithm *LEASE_Host_MP*
CL_i: Candidate data items for lease from provide-MP i
Spc: Its available memory space
(1) for each provide-MP i
(2) Receive broadcast message from i containing items for lease
(3) Add all data items in CL_i to a set $bigCL$
(4) Sort all data items in $bigCL$ in descending order of γ
(5) for each data item d in $bigCL$
(6) /* $size_d$ is the size of d */
(7) if ($size_d \leq Spc$)
(8) Add d to a set BID
(9) Spc = Spc - $size_d$
(10) if (Spc = = 0) **exit**
(11) for each data item d in set BID
(12) Send the bid of β_d to the corresponding provide-MP
(13) if bid is successful
(14) Obtain d from corresponding provide-MP with β_d as lease payment
end

Fig. 2. LEASE algorithm for host-MP

Figure 2 depicts the algorithm executed by a host-MP H to facilitate it in *simulating* the choice of data items that it should bid for. H may not necessarily be able to obtain a lease for all the data items that it bids for since other MPs may outbid H, hence it is a *simulation*. Thus, H greedily *simulates* the filling up of its memory space by data items with higher value of γ. (γ is computed in Section 3). In Lines 12-14, the value of β_d is computed by Equation 3.

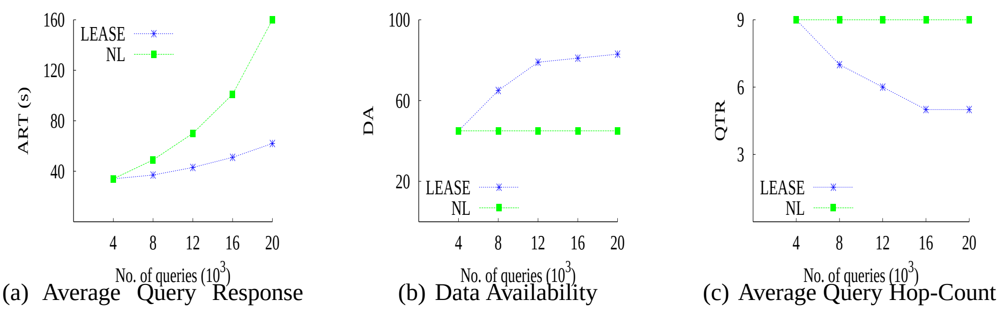
5 Performance Evaluation

MPs move according to the *Random Waypoint Model* [2] within a region of area 1000 metres $\times$1000 metres. The *Random Waypoint Model* is appropriate for our application scenarios, which involve random movement of users. A total of 100 MPs comprise 30 data-providers and 70 free-riders (which provide no data). Each data-provider owns 8 data items comprising 4 *shared* items and 4 *unshared items*. Each query is a request for a single data item. 20 queries/second are issued in the network, the number of queries directed to each MP being determined by a highly skewed Zipf distribution with Zipf factor of 0.9. Communication range of all MPs is a circle of 100 metre radius. Table 2 summarizes our performance study parameters.

Performance metrics are **average response time** (**ART**) of a query, **data availability** (**DA**) and **average querying traffic** (**QTR**). ART equals $((1/N_Q)\sum_{i=1}^{N_Q}(T_f - T_i))$,

Parameter	Default value	Variations
No. of MPs (N_{MP})	100	20,40,60,80
Zipf factor (ZF)	0.9	
Queries/second	20	
Bandwidth between MPs	28 Kbps to 100 Kbps	
Probability of MP availability	50% to 85%	
Size of a data item	50 Kb to 350 Kb	
Memory space of each MP	1 MB to 1.5 MB	
Speed of an MP	1 metre/s to 10 metres/s	
Size of message headers	220 bytes	

Table 2. Performance Study Parameters



(a) Average Query Response Time (b) Data Availability (c) Average Query Hop-Count

Fig. 3. Performance of LEASE

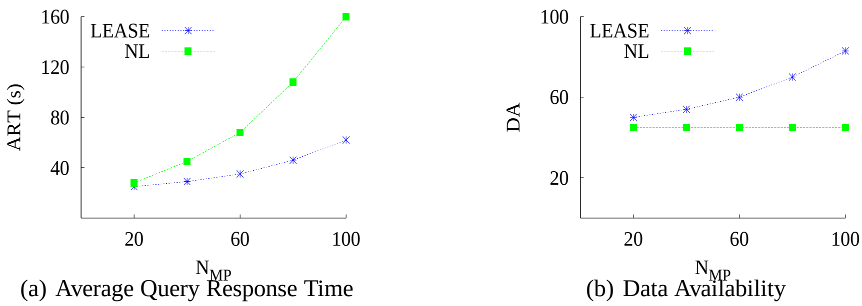
where T_i is the query issuing time, T_f is the time of the query result reaching the query issuing MP, and N_Q is the total number of queries. ART includes data download time, and is computed only for successful queries. DA equals $((N_S/N_Q) \times 100)$, where N_S is the number of successful queries and N_Q is the total number of queries. Queries can fail due to MPs being switched 'off' or due to network partitioning. QTR is the average number of hops per query.

As reference, we adapt a non-economic model **NL (No_Lease)** since existing M-P2P proposals do not address economic lease-based models. In NL, leasing is not performed and querying is broadcast-based. As NL does not provide incentives for free-riders to become host-MPs, only a single copy of any given data item d exists at the owner of d.

Performance of LEASE: Figure 3 depicts the performance of LEASE using default values of the parameters in Table 2. Leasing procedures are initiated only after the first 4000 queries, hence both LEASE and NL initially show comparable performance. The ART of both LEASE and NL increases with time due to the skewed workload (ZF = 0.9), which overloads some of the MPs that store 'hot' data items, thereby forcing queries to incur high waiting times and consequently high ART. However, over time, the economic incentives of LEASE entice more MPs to host data items, thereby increasing the resources (e.g., bandwidth, memory space) in the network for creating multiple (leased) copies for the same data item to facilitate load-balancing as well as reduction of QTR. Moreover, LEASE considers the connectivity of host-MPs, which further decreases its QTR, thereby decreasing ART. In Figure 3b, DA eventually plateaus for LEASE due to reasons such as network partitioning and unavailability of some of the MPs.

In contrast, the non-economic nature of NL does not entice the free-riders to host data items via leasing, thus the ART of NL keeps increasing due to overloading of MPs storing 'hot' data items. For NL, DA remains relatively constant since it depends only on the probability of availability of the MPs. The QTR for NL remains relatively constant as only one copy of any given data item d exists in the network.

Effect of variations in the number of MPs: To test LEASE's scalability, we varied the number N_{MP} of MPs, while keeping the number of queries proportional to N_{MP}. In each case, 30% of the MPs were data-providers, the rest being free-riders. As the results in Figure 4 indicate, ART increases for both approaches with increasing N_{MP} due to larger network size. At higher values of N_{MP}, LEASE outperforms NL due to the reasons explained for Figure 3.As N_{MP} decreases, the performance gap decreases due to limited leasing opportunities, which results in lesser number of copies for leased data items, thereby making the effect of leasing less prominent.



(a) Average Query Response Time (b) Data Availability

Fig. 4. Effect of varying the number of MPs

6 Conclusion

We have proposed LEASE, a novel Mobile-P2P lease-based economic incentive model, in which data requestors need to pay the price (in virtual currency) of their requested data items to data-providers. In LEASE, data-providing mobile peers lease data items to free-riders, who do not have any data items to provide, in lieu of a lease payment. Thus, LEASE not only combats free-riding, but also entices free-riders to host data items, thereby improving network connectivity due to higher peer participation. In essence, LEASE facilitates the collaborative harnessing of limited mobile peer resources for improving data availability. Our performance study shows that LEASE indeed improves query response times and data availability in M-P2P networks.

References

1. K.G. Anagnostakis and M.B. Greenwald. Exchange-based incentive mechanisms for peer-to-peer file sharing. *Proc. ICDCS*, 2004.
2. J. Broch, D.A. Maltz, D.B. Johnson, Y.C. Hu, and J. Jetcheva. A performance comparison of multi-hop wireless ad hoc network routing protocol. *Proc. MOBICOM*, 1998.
3. L. Buttyan and J.P. Hubaux. Stimulating cooperation in self-organizing mobile ad hoc networks. *Proc. ACM/Kluwer Mobile Networks and Applications*, 8(5), 2003.
4. E. Elrufaie and D. Turner. Bidding in P2P content distribution networks using the lightweight currency paradigm. *Proc. ITCC*, 2004.
5. D.F. Ferguson, Y. Yemini, and C. Nikolaou. Microeconomic algorithms for load balancing in distributed computer systems. *Proc. ICDCS*, pages 491–499, 1988.
6. Gnutella. http://www.gnutella.com/.
7. C. Grothoff. An excess-based economic model for resource allocation in peer-to-peer networks. *Proc. Wirtschaftsinformatik*, 2003.
8. T. Hara and S.K. Madria. Data replication for improving data accessibility in ad hoc networks. *IEEE Transactions on Mobile Computing*, 2006.
9. http://www.microsoft.com/presspass/presskits/zune/default.mspx.
10. S. Kamvar, M. Schlosser, and H. Garcia-Molina. Incentives for combatting free-riding on P2P networks. *Proc. Euro-Par*, 2003.
11. Kazaa. http://www.kazaa.com/.
12. J. F. Kurose and R. Simha. A microeconomic approach to optimal resource allocation in distributed computer systems. *Proc. IEEE Trans. Computers*, 38(5):705–717, 1989.
13. A. Mondal, S.K. Madria, and M. Kitsuregawa. CADRE: A collaborative replica allocation and deallocation approach for Mobile-P2P networks. *Proc. IDEAS*, 2006.
14. A. Mondal, S.K. Madria, and M. Kitsuregawa. EcoRep: An economic model for efficient dynamic replication in Mobile-P2P networks. *Proc. COMAD*, 2006.
15. First Workshop on the Economics of P2P Systems. http://www.sims.berkeley.edu/research/conferences/p2pecon. 2003.
16. I. Stoica, R. Morris, D. Karger, M.F. Kaashoek, and H. Balakrishnan. Chord: A scalable peer-to-peer lookup service for internet applications. *Proc. ACM SIGCOMM*, 2001.
17. D.A. Turner and K.W. Ross. A lightweight currency paradigm for the P2P resource market. *Proc. Electronic Commerce Research*, 2004.
18. O. Wolfson, S. Jajodia, and Y. Huang. An adaptive data replication algorithm. *Proc. ACM TODS*, 22(4):255–314, June 1997.
19. O. Wolfson, B. Xu, and A.P. Sistla. An economic model for resource exchange in mobile Peer-to-Peer networks. *Proc. SSDBM*, 2004.
20. B. Xu, O. Wolfson, and N. Rishe. Benefit and pricing of spatio-temporal information in Mobile Peer-to-Peer networks. *Proc. HICSS-39*, 2006.
21. S. Zhong, J. Chen, and Y.R. Yang. Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks. *Proc. IEEE INFOCOM*, 2003.