

# Top- $k$ query processing in mobile-P2P networks using economic incentive schemes

Nilesh Padhariya<sup>1</sup> · Anirban Mondal<sup>2</sup> · Sanjay Kumar Madria<sup>3</sup>

Received: 6 October 2014 / Accepted: 8 July 2015 / Published online: 4 August 2015  
© Springer Science+Business Media New York 2015

**Abstract** This work proposes the E-Top system for the efficient processing of top- $k$  queries in mobile ad hoc peer to peer (M-P2P) networks using economic incentive schemes. In E-Top, brokers facilitate top- $k$  query processing in lieu of a commission. E-Top issues economic rewards to the mobile peers, which send relevant data items (i.e., those that contribute to the top- $k$  query result), and penalizes peers otherwise, thereby optimizing the communication traffic. Peers use the payoffs (rewards/penalties) as a means of feedback to re-evaluate the scores of their items for re-ranking purposes. The main contributions of E-Top are three-fold. First, it proposes two economic incentive schemes, namely ETK and ETK+, in which peers act individually towards top- $k$  query processing. Second, it extends ETK and ETK+ to propose a peer group-based economic incentive scheme ETG. Third, our performance evaluation shows that our schemes are indeed effective in improving the performance of top- $k$  queries in terms of query response times and accuracy at reasonable communication traffic cost.

**Keywords** Economic incentive schemes · Top- $k$  query processing · Data availability · 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) strongly motivate M-P2P applications.

Suppose Alice wants to find the top- $k$  restaurants with “happy hours” (or “manager’s special hours”) within 1 km of her current location. Top- $k$  is determined based on the parameters (e.g., star rating, price and distance from the point of query reference) selected by the user. A broker can facilitate such range-constrained top- $k$  queries by soliciting information from the MPs in its vicinity, and it can then compare this information with its current top- $k$  list of restaurants to generate the top- $k$  result to be provided to the query-issuing MP. The broker compiles its current top- $k$  list by periodically collecting information from various sources such as the Web and social networking sites.

In a similar vein, another application could involve a parking lot, where MPs can collect information about available parking slots and charges, and then they can inform the brokers. The parking slot availability information has to be current and therefore, the broker can compare such current information with its current list of parking slots. The broker can then provide the top- $k$  available slots to the query-issuing MP in terms of price or distance (from the MP’s current location). Similarly, an MP may want to find the top- $k$  stores selling Levis jeans in a shopping mall with criteria such as (low) price during a specific time duration.

---

✉ Sanjay Kumar Madria  
madrias@mst.edu

Nilesh Padhariya  
nileshp@iiitd.ac.in

Anirban Mondal  
anirban2345@gmail.com

<sup>1</sup> Indraprastha Institute of Information Technology, Delhi, India

<sup>2</sup> Xerox Research, Bengaluru, India

<sup>3</sup> Missouri University of Science, Technology,  
Rolla, MO 65409, USA

We assume that all the mobile peers are autonomous and heterogeneous, but they work for the given application. The system is decentralized with the exception that the multiple brokers could be there and they are pre-selected i.e., there are certain peers that are pre-defined as brokers during the network configuration phase in the beginning. In general, we assume that brokers are trusted entities with relatively more resources. Moreover, brokers are those nodes that do not make wide-area movements. For example, a parking lot manager with a PDA/laptop/smartphone may act as a broker for the parking application. Similarly, the managers of some of the restaurants in a given neighbourhood can serve as brokers using their PDAs/laptops/smartphones. Thus, brokers manage the mobile peers in their vicinity and provide value-added services. As we shall see later, brokers facilitate top- $k$  query processing in lieu of a commission.

Incidentally, ad hoc queries in our system are of spatio-temporal nature (e.g., parking slot availability information), thereby implying that the query results can change dynamically. Hence, they cannot be answered by the broker without obtaining information from other MPs. For example, such spatio-temporal aspects may concern the ambience of a restaurant or how crowded a shopping mall is during a given point in time. Observe that this would change temporally based on time and days.

Notably, this research will also contribute towards CrowdDB [13], which uses human input via crowdsourcing to process queries that cannot be answered by database systems or search engines. Additionally, such M-P2P interactions among peers are generally not freely supported by existing wireless communication infrastructures. The inherently ephemeral nature of M-P2P environments suggests that *timeliness* of data delivery is of paramount importance in these applications, thereby necessitating query deadlines. For example, an MP looking for top- $k$  restaurants with “happy hours” would generally prefer to receive the answer within a specified deadline.

Incidentally, Amazon.com has developed Mechanical Turk [1], which is an online marketplace for match-making between the requirements of businesses and the skill sets of developers. Developers can select from a large pool of tasks based on their skill sets. Similar to our work, the Mechanical Turk system also provides economic incentives. Observe that technologies, such as WiFi and Bluetooth networks, are nowadays adequately capable of providing a platform for incentive-based mobile P2P collaborations.

Existing economic schemes for distributed systems [15, 27] and static P2P networks [14, 24, 30] do not address top- $k$  queries and M-P2P issues such as frequent network partitioning and mobile resource constraints. Economic incentive schemes for mobile ad-hoc networks (MANETs) [5] and M-P2P networks [40, 42] do not address top- $k$  query processing. Furthermore, the top- $k$  query processing

approaches [18, 22, 23, 28, 32, 38, 41] do not consider economic incentive schemes and M-P2P architecture.

Incidentally, data availability in M-P2P networks is typically lower than in fixed networks due to frequent network partitioning [21] arising from peer movement and/or peers autonomously switching ‘off’ their mobile devices. Data availability is further exacerbated due to rampant free-riding [14, 24, 30], which is characteristic of P2P environments. Furthermore, MPs generally have limited resources (e.g., bandwidth, energy, memory space). Since sending/receiving messages expend the limited energy resources of MPs, minimizing the communication traffic becomes a necessity to address energy constraints. Thus, economic incentive schemes become a necessity to entice resource-constrained MPs with incentives to provide data for answering queries.

This work proposes the **E-Top** system for addressing efficient top- $k$  query processing in M-P2P networks. In E-Top, we have considered that a given query-issuer sends location-based *range-constrained* top- $k$  queries to the M-P2P network. Brokers facilitate top- $k$  query processing in lieu of a commission. E-Top requires a query-issuing MP to pay a *price* (in *virtual currency*) for obtaining its queried top- $k$  result. This price is used for making payments for incentivizing *rankers* (i.e., peers that send data items to answer the query), brokers and relay peers. Thus, an MP has to earn adequate currency by providing *service* (as a broker, ranker or relay peer) before it can issue its own top- $k$  queries, thereby discouraging free-riding.

E-Top issues economic rewards to the rankers, which send relevant data items (i.e., those that contribute to the top- $k$  query result), and penalizes peers for sending irrelevant items. This incentivizes MPs to send only those data items (to the broker), which have a higher probability of being in the top- $k$  results, thereby optimizing the communication traffic. MPs use the rewards/penalties as feedback to re-evaluate their items’ scores. We shall henceforth use the term **payoffs** to refer to rewards/penalties.

The main contributions of E-Top are three-fold:

1. It proposes two economic incentive schemes, namely ETK and ETK+, in which MPs act individually towards top- $k$  query processing. These schemes assign payoffs to MPs for incentivizing participation and for enabling them to re-evaluate their data item scores.
2. It extends ETK and ETK+ to propose a peer group-based economic incentive scheme ETG, which defines three payoff allocation approaches.
3. It is indeed effective in improving the performance of top- $k$  queries in terms of query response times and accuracy at reasonable communication traffic cost, as demonstrated by our performance evaluation.

E-Top also discourages free-riding due to its economic nature. ETK and ETK+ differ in that while ETK performs

equal distribution of payoffs to the rankers, ETK+ uses a weighted distribution. In ETG, ad hoc groups of MPs are formed in the vicinity of the query location. Each group has a leader for coordinating the top- $k$  query processing. In contrast with ETK and ETK+, where individual MPs directly send their top- $k$  items to the broker, query processing in ETG proceeds by means of group members sending their individual top- $k$  items to the group leader. The group leader selects (i.e., ‘filters’) the top- $k$  items to be sent to the broker based on the relative frequencies of the items in the individual top- $k$  lists. In our application scenarios, some of the restaurant managers in the vicinity of the query location can be the group leaders.

In the three approaches deployed by ETG for payoff allocation among group members for any given top- $k$  query, group penalties are equally distributed; thus the schemes differ in their allocation of group rewards. Group rewards are allocated in the following three ways i.e., equally, based on the number of relevant items sent and based on the revenue earned from those items. The group leader receives a percentage of the group rewards as a commission, thereby incentivizing it to participate. Group-based collaboration provides better incentivization since it is likely to lead to higher rewards and lower penalties due to the following reasons. First, MPs risk a lower amount of individual penalties due to the sharing of penalties among group members. Second, MPs have a higher probability of obtaining rewards because the ‘filtering’ performed by the group leader ensures that collective top- $k$  answers from group members are likely to be of higher quality (i.e., more relevant and accurate) than individual answers.

To the best of our knowledge, none of the existing top- $k$  query processing schemes in M-P2P environment uses incentives. Hence, as reference, we adapt an existing non-incentive-based top- $k$  processing scheme for MANETs. We designate this scheme as NETK (Non-Economic Top- $K$ ), proposed in [20]. Although NETK does not provide incentives to the MPs, it is closest to our top- $k$  query processing scheme. Notably, NETK does not incorporate the notion of item re-ranking as no feedback has been sent back to the MPs, who participated in the top- $k$  query processing.

The results of our performance evaluation indicate that ETG outperforms both ETK and ETK+ due to its group-based scheme, which better incentivizes MP collaboration in top- $k$  query processing due to effective sharing of rewards and penalties among group members. Moreover, ETK+ outperforms ETK due to its weighted distribution (of rewards and penalties to ranker MPs), which provides better incentives to ranker MPs than ETK’s equal distribution. ETK, ETK+ and ETG outperform NETK essentially due to the effectiveness of economic payoffs and item re-ranking.

The results also indicate that at higher values of  $k$ , query response times increase for all the schemes due to longer

query paths. This is because fewer nearby rankers are able to provide enough relevant data items pertaining to the top- $k$  query. Our schemes exhibit good scalability with increasing number of MPs because larger network implies the presence of more rankers. Our schemes exhibit improvement in performance as the communication range of MPs increases. This is because increase in communication range has the effect of bringing the MPs ‘nearer’ to each other, thereby improving data accessibility.

As the percentage of MP failures increases, our schemes degrade in performance partly due to decreased overall MP participation and partly because of failure of MPs that host data relevant to the top- $k$  queries. ETG performs best when the group sizes are neither too small nor too large. This is because medium-sized groups are better able to leverage the benefits of group-based collaboration.

The remainder of this paper is organized as follows. Section 2 reviews existing works, while Section 3 details the architecture of E-Top. Section 4 discusses the ETK and ETK+ economic incentive schemes in E-Top. Section 5 presents the peer group-based ETG economic incentive scheme of E-Top. Section 6 reports our performance study. We conclude in Section 7 with directions for future work.

## 2 Related work

This section provides an overview of existing works.

### 2.1 Top- $k$ query processing approaches

The proposals in [32, 41] discuss top- $k$  query processing in wireless sensor networks. For conserving the energy of the sensor nodes, the work in [32] minimizes redundant data transmissions by means of both a cluster-tree routing structure for locally aggregating objects as well as a cross-pruning technique for filtering purposes. The work in [41] exploits semantics and facilitates energy-efficiency by installing a filter at each sensor node to avoid unnecessary updates. The work in [28] uses a probabilistic approach towards cost-effectively selecting sensor nodes for processing continuous probabilistic queries in wireless sensor networks by reducing sensor data aggregation. The tutorial in [45] provides a comprehensive overview of top- $k$  query processing in wireless sensor networks.

The proposal in [18] discusses a message processing method for top- $k$  queries in MANETs for reducing the communication traffic. The work in [23] discusses location-based top- $k$  query processing for wireless broadcast environments using two R-tree variants, namely the broadcast aggregate R-tree and the bit-vector R-tree. The work in [29] presents a search engine geared towards providing

mobile users with top- $k$  web search results. The proposal in [38] addresses top- $k$  queries and aggregate queries for probabilistic databases with focus on data uncertainty and semantics. The work in [22] examines the optimization of top- $k$  queries in middleware by means of a cost-based optimization approach.

Notably, these existing top- $k$  query processing approaches in MANETs do not use incentives, thereby potentially resulting in low peer participation and consequently, causing low data availability. Furthermore, they do not incorporate M-P2P architecture.

## 2.2 Economic incentive schemes

Economic schemes for resource allocation in distributed systems [11, 12, 27] do not address M-P2P issues such as node mobility, free-riding, frequent network partitioning and mobile resource constraints. Economic models for resource allocation in P2P networks [15] do not address top- $k$  queries. Economic schemes for resource allocation in wireless ad hoc networks [31, 43, 44] consider a network-centric focus (unlike our data-centric focus) and they do not address top- $k$  queries.

Incentive-based schemes for encouraging peer participation in static P2P networks involve formal game-theoretic model for incentive-based P2P file-sharing systems [14], utility functions to capture peer contributions [19, 34], EigenTrust scores to capture participation criteria [24] and asymmetric incentives based on disparities between upload and download bandwidths [30]. However, these approaches are too static to be deployed in M-P2P networks because they assume peers' availability and fixed topology. Furthermore, they do not address mobile resource constraints (e.g., energy) and top- $k$  query processing issues.

The proposals in [4, 5, 7, 8, 39] address free-riding in MANETs. The work in [4] introduces a virtual currency to stimulate node cooperation. The works in [5, 46] use virtual currency to stimulate the cooperation of mobile nodes in forwarding messages. The auction-based iPass [7] incentive scheme and the works in [8, 39] also provide incentives for relaying messages. In particular, the proposals in [8, 39] concentrate on compensating forwarding cost in terms of battery power, memory and CPU cycles. However, these works do not consider top- $k$  query processing, payoffs to rankers and M-P2P architecture.

The work in [42] discusses an incentive scheme for the dissemination of information concerning spatio-temporal resources in M-P2P networks. The work in [40] considers opportunistic resource information dissemination in M-P2P transportation application scenarios. The works in [40, 42] primarily address data dissemination with the aim

of reaching as many peers as possible, while we emphasize on-demand data dissemination (i.e., a query-based approach). Furthermore, top- $k$  queries are not addressed in [40, 42].

## 2.3 Mobile system applications and payment schemes

MoB [6] is an open market collaborative wide-area wireless data services architecture, which can be used by mobile users for opportunistically trading services with each other. MoB also handles incentive management, user reputation management and accounting services. Incidentally, P2P replication suitable for mobile environments has been incorporated in systems such as ROAM [35], Clique [36] and Rumor [17]. However, these systems do not incorporate economic incentive schemes and top- $k$  queries.

The proposal in [26] discusses the placement of roadside units (RSUs) in participatory sensing using vehicular networks. In particular, it proposes a multi-objective optimization evolutionary algorithm with heuristics to minimize the number of RSUs, while performing such placement. Furthermore, the work in [33] discusses effective data downloading for real-time applications, where user queries are prioritized by the delivery deadline. In particular, it proposes a cooperative downloading algorithm for maximizing the amount of data packets downloaded from the RSU, while minimizing an average delivery delay of each user query. The work in [10] discusses issues regarding the Quality-of-Experience (QoE) of users in mobile social networks, and proposes future research directions for mobile social networks from the perspective of QoE.

Several non-repudiation [25, 37] systems, which can be incorporated to control the deceiving behaviour of peers, have been developed. A bootstrap kind of mechanism can also be used in many applications [9]. Symella is a Gnutella file-sharing client for Symbian smartphones. It expects that illegal acts occur, such as the manipulation of the distribution history to get incentives. Hence, the distribution history attached to the e-coupon [7] is enciphered with a public-key cryptographic system so that users cannot peruse the distribution history. Notably, these secure payment schemes are complementary to our proposal, but they can be used in conjunction with our proposal.

## 3 Architecture of E-Top

The architecture of E-Top consists of MPs that can assume one of the four following roles: *query-issuer*, *broker*, *ranker* and *relay*. Notably, these roles are interchangeable e.g., a given MP can be a broker for a top- $k$  query  $Q_1$ , but a ranker for another top- $k$  query  $Q_2$ .

We assume that an MP is an autonomous entity, which may join or leave the M-P2P network at any point of time. Moreover, we also assume that the MPs are interested in maximizing their revenues by earning more currency (incentives) by providing valuable services as a relay peer or a ranker in the M-P2P network. Furthermore, we assume that the underlying network takes care of the delivery of messages.

*Query-issuer*  $QI$  issues queries of the form  $(k, L, \tau_Q, \rho)$ , where  $k$  is the number of data items that are requested in the location-based *range-constrained* top- $k$  query.  $L$  represents the query location, and is of the form of  $\{(x, y), rad\}$ . Here,  $(x, y)$  represents the spatial coordinates associated with a given query  $Q$ , while  $rad$  represents the radius. For example,  $QI$  may want to find restaurants within 1 km of its current location  $L$ .  $\tau_Q$  is the deadline time of  $Q$ .  $\rho$  is the query price that  $QI$  will pay to obtain the top- $k$  query result<sup>1</sup>. An MP decides the query price based on his/her information. Moreover, brokers periodically broadcast price ranges for data from different domains such as restaurants, travel and so on. MPs can also subscribe for such information, and brokers can inform them from time to time. *Broker*  $B$  acts as a mediator, which facilitates efficient top- $k$  query processing in lieu of a commission. As we shall see in Section 4,  $B$  also performs economic incentive functions i.e., distribution of payoffs. Notably, the payoffs are distributed to the individual rankers only for the **completed queries**. A given query is deemed to be completed if the broker receives at least  $k$  items from the individual rankers (or group leaders in case of ETG) within  $x\%$  of  $\tau_Q$  in the system. Observe that the broker must receive the items from the individual rankers within  $x\%$  of the deadline time  $\tau_Q$  so that it can use the remaining time to the deadline for performing the computations associated with collating the items from the rankers and determining the top- $k$  result. Based on the results of our preliminary experiments, we found  $x \approx 70$  to be a suitable value for our experiments.

*Rankers* are MPs, which provide data items for answering the top- $k$  query. Rankers are rewarded if their items contribute to the top- $k$  result, otherwise they are penalized. Relay MPs forward messages in multi-hop M-P2P networks in lieu of a small constant commission. Notably, payments to rankers are typically higher than that of broker commissions in order to better incentivize MPs to provide data. This is because MPs providing data generally contribute significantly more to data availability than brokers. Furthermore, relay commission is lower than that of broker commission to better incentivize brokerage functions as compared to relay functions.

During the network configuration phase in the beginning, the broker will be pre-defined, but can also be elected based

on the resources. We assume that an MP with relatively more resources may want to become a broker for a given query  $Q$ , as it can provide better services, while in case of low resources, an MP should play a role of relay peer, as it requires very few resources for relay service. Moreover, an MP, which has an answer to a given query  $Q$ , may be more interested to be a ranker for that query to earn rewards. Hence, our system does not assign specific roles to the MPs, thereby providing them with the flexibility to decide their respective roles for a given query. However, the role assignment for a broker is done in a pre-defined manner.

Notably, we divide the region of interest into square cells of equal area in a grid. Since MPs may not be uniformly distributed across the cells, the density can vary across cells. Observe that the density  $d$  of a broker's cell is an important consideration for E-Top because a broker connected with more MPs (or groups in case of ETG) is preferable over one connected with less MPs (or groups). In E-Top, a broker estimates  $d$  for its cell by examining the average number of unique MPs, which had connected to it, during the past  $N$  time periods. Observe that this is a moving average. (We divide time into equal intervals called periods, the size of a period being application-dependent.) The results of our preliminary experiments showed that  $N = 5$  is a reasonable value for our application scenarios. We have defined  $d$  as follows:

$$d = \frac{1}{N} \sum_{i=1}^N \left( np_i / \sum_{j=1}^R tp_{ij} \right) \quad (1)$$

where  $np_i$  is the number of unique MPs, which had connected to the broker during the  $i^{th}$  time period, while  $tp_{ij}$  is the total number of MPs in the  $j^{th}$  region of interest and  $R$  is the number of regions that broker passed through during the  $i^{th}$  time period. Whenever the brokers come within communication range of each other, they exchange information about  $tp_{ij}$ . Brokers periodically broadcast the value of  $tp_{ij}$  in their respective region so that all the MPs are aware of the value of  $tp_{ij}$ . Since  $np_i < \sum_{j=1}^R tp_{ij}$ , therefore  $0 \leq d \leq 1$ .

### 3.0.1 Query processing in E-Top

Figure 1 illustrates query processing in E-Top.

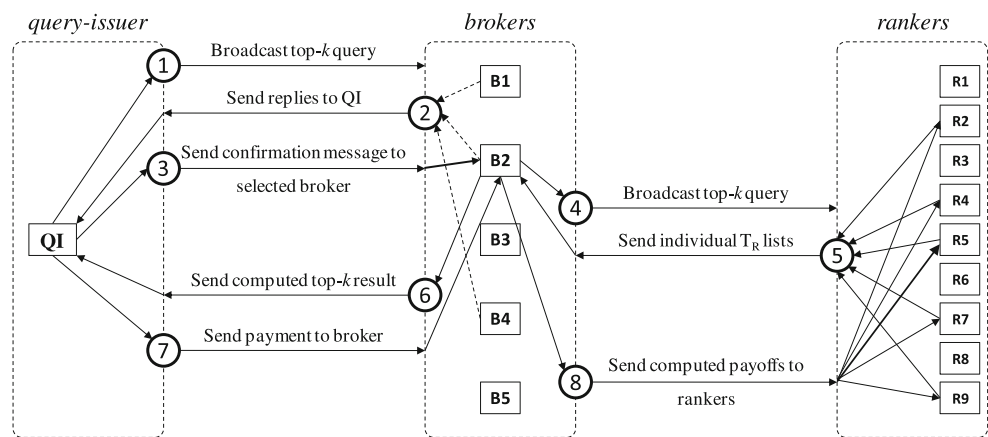
Query-issuer  $QI$  broadcasts a top- $k$  query  $Q$ , and waits for  $W$  time units to get replies from the potential brokers.  $W$  is computed as below:

$$W = (1 - d) \times \tau_Q \quad (2)$$

where  $d$  is the density of the query issuer's region (i.e., square cell), and it is computed using Eq. 1.  $\tau_Q$  is the query deadline time of  $Q$ . Notably,  $QI$  estimates the value of  $np_i$  in Eq. 1 as the average number of unique MPs, which

<sup>1</sup>Query results received by  $QI$  after the deadline entail no payments.

**Fig. 1** Illustrative example of query processing in E-Top



connected to it during recent time periods. As Eq. 2 indicates,  $QI$  is willing to wait longer for replies from potential brokers if the density of its region is low.

Each *broker* replies to  $QI$  with information about its remaining energy  $En$ , bid price  $\rho_{bid}$ , current currency  $Curr$ , distance  $Dist$  from  $QI$  and density  $d$  of its current location.  $QI$  computes the average location density  $d_{avg}$  as  $\frac{1}{n} \sum_{i=1}^n d_i$ , where  $d_i$  is the density for the  $i^{th}$  broker, and  $n$  is the total number of brokers that replied to  $QI$ . Now, as candidates,  $QI$  will only consider brokers, whose value of  $d$  exceeds  $d_{avg}$  because brokers in higher-density locations are likely to provide better service due to their proximity to an increased number of potential rankers. Thus, for each broker, whose density exceeds  $d_{avg}$ ,  $QI$  computes a score  $\eta$  and selects the broker with the highest value of  $\eta$  for processing  $Q$ .  $\eta$  is computed below:

$$\eta = (w_1 \times En) + (w_2 / \rho_{bid}) + (w_3 / Curr) + (w_4 / Dist) \quad (3)$$

where  $w_1$  to  $w_4$  are weight coefficients such that  $0 < w_1, w_2, w_3, w_4 \leq 1$  and  $\sum_{i=1}^4 w_i = 1$ . Thus, E-Top prefers relatively high-energy brokers because they are less likely to run out of energy, while processing the query. Lower values of bid prices are preferred by  $QI$  since it wants to obtain the query result at lower cost. Brokers with less currency are given higher preference to facilitate revenue-balancing across brokers. This prevents low-currency brokers from starvation, which may result in decreased number of brokers in the network.  $QI$  prefers relatively nearby brokers to obtain the query result in a timely manner.

Now the broker broadcasts  $Q$  with time-to-live (TTL) of  $n$  hops. (Results of our preliminary experiments showed that  $n = 6$  is a reasonable value for our application scenarios.) The high value of TTL leads to the longer query path, hence it increases both the query latency and the communication overhead. But very low value of TTL also has negative impacts such as decreasing in peer participation, thereby reducing the data accuracy and the success rate. Hence, considering the impacts of very high or very low

values of TTL, we considered to keep the value of TTL reasonable, which is dependent on the application scenario and the density of the region. Here, low-density region may need high TTL and vice versa.

Each ranker  $R$  has an *individual* item ranking list  $T_{fR}$ , each data item of which is associated with an item rank  $r$  and a selection probability  $\mu$ . Notably, the value of  $r$  is subjective because it is autonomously assigned to an item by a given ranker. The implication is that the same item may be ranked differently at different rankers. As we shall see in Section 4,  $\mu$  facilitates the adjustment of item selection probability based on recent payoffs assigned to a given item. Using the values of  $\mu$  and  $r$ , each ranker  $R$  computes a score  $\gamma$  and selects items with relatively higher values of  $\gamma$  to send to the broker.  $\gamma$  is computed below:

$$\forall i \in T_{fR} : \gamma_i = (w_1 \times (N_{T_{fR}} - r_i) / N_{T_{fR}}) + (w_2 \times \mu_i) \quad (4)$$

where  $r_i$  and  $\mu_i$  are the rank and the selection probability of item  $i$  respectively.  $N_{T_{fR}}$  is the total number of items in  $T_{fR}$ . Here,  $w_1$  and  $w_2$  are weight coefficients such that  $0 < w_1, w_2 \leq 1$  and  $w_1 + w_2 = 1$ . E-Top stipulates that  $w_2 > w_1$  to give higher weightage to the item selection probability than to the rank of the item. As we shall see in Section 4, this is consistent with the overall objective of E-Top i.e., linking item re-ranking with payoffs. Moreover, these weight coefficients are application-dependant i.e., according to application's requirement, weight coefficients are set to any values in-between 0 and 1. There is no restriction on whether to choose  $w_1 > w_2$  or  $w_2 > w_1$ , but to prioritize the item's selection probability, we have chosen  $w_2 > w_1$  for our proposed application scenarios. In this work, based on our experimental results, we set  $w_1 = 0.2$  and  $w_2 = 0.8$  for all the MPs. Furthermore, each ranker is associated with a risk profile  $\delta$ , where  $0 < \delta \leq 1$ . Only items, whose respective values of  $\gamma$  exceed  $\delta$ , are consolidated by the ranker in a list  $T_R$  and sent to the broker. Thus,  $T_R$  is a sorted item ranking list, which is sent by an individual MP in response to a query. Hence,  $T_R \subseteq T_{fR}$ . Observe

that as the value of  $\delta$  increases, the risk of the ranker in incurring a penalty decreases.

E-Top considers that each broker has a global ranking list, which we shall henceforth designate as  $T_G$ . Here,  $T_G$  is a global standard (e.g., Michelin guide to restaurants) across the system for considering guideline for the items' ranks. This approach is adopted to incorporate the global rank views (such as Internet or feedback-based) about the items along with the local rankings.  $T_G$  is periodically exchanged among nearby brokers. Upon receiving the individual  $T_R$  lists from possibly multiple rankers, the broker  $B$  collates and compares them with  $T_G$ .  $B$  parses  $T_G$  in a top-down fashion as follows. If an item  $i$  in  $T_G$  occurs in at least one of the individual  $T_R$  lists, it is added to the top- $k$  result set  $T_A$  along with the unique identifiers of the rankers that sent  $i$ . (In case  $i$  does not occur in any of the individual  $T_R$  lists,  $B$  simply traverses downwards to the next item in  $T_G$ .)  $B$  continues parsing  $T_G$  in the above manner until the result set  $T_A$  contains  $k$  items. Then  $B$  sends  $T_A$  to  $QI$ . Notably, if  $T_A$  contains less than  $k$  items, the result set is deemed to be *incomplete*, and it is not sent to  $QI$ .

Upon receiving  $T_A$ ,  $QI$  pays  $B$ , which deducts its own commission before distributing the payoffs to rankers and commissions to relay MPs. (We shall discuss ranker payoffs, and broker and relay commissions in Section 4.) Then each ranker  $R$  re-evaluates the selection probability  $\mu$  of each item in its own  $T_R$  based on received payoffs, and then re-computes the values of  $\gamma$  for these items.

In this work, we do not address the formation of the global list  $T_G$  because this is application-dependent. Moreover, we do not consider updates to  $T_G$  because it may exist for a long time. Furthermore, any update to  $T_G$  must be propagated to all the relevant brokers, which also increases the communication overhead.

#### 4 Economic incentive schemes in E-Top: ETK and ETK+

This section discusses the ETK and ETK+ economic incentive schemes used by E-Top. We define an item  $i$  to be **relevant** to a top- $k$  query  $Q$  if it occurs in the top- $k$  query result set  $T_A$ . We define a **successful** ranker w.r.t. its (sent) data item  $i$  if  $i$  is relevant to  $Q$ , otherwise the ranker is deemed to be *unsuccessful*. Thus, a ranker may be successful w.r.t. item  $i$ , but unsuccessful w.r.t. another item  $j$ . Notably, in ETK and ETK+, a ranker can only participate for query  $Q$  if it hosts at least  $k$  relevant items to  $Q$ .

Incidentally, a given ranker  $R$  has no way of knowing if its sent-result would finally occur in  $T_A$ .  $R$  may maintain historical data concerning items that have occurred previously in  $T_A$  in response to similar queries. However, if a new

query comes to  $R$ , no such historical data would be available at  $R$ . In such cases,  $R$  would send its individual top- $k$  ranking list without considering the historical data.

In both ETK and ETK+, the total payment  $\rho_R$  to be distributed to the successful rankers is computed as follows:

$$\rho_R = \rho - \rho_B - \rho_{RL} \quad (5)$$

where  $\rho$  is the query price paid by  $QI$  to the broker,  $\rho_B$  is the broker commission and  $\rho_{RL}$  is the total amount of relay commission that the broker will pay to the relay MPs in the respective successful query paths. Notably, the value of  $\rho_B$  is application-dependent. For both ETK and ETK+, we defined  $\rho_B$  as 10 % of the query price  $\rho$ . Although our schemes can be intuitively generalized to work with other values of  $\rho_B$ , results of our preliminary experiments showed that our schemes perform best when  $\rho_B$  is in the range of 5 % to 15 % of  $\rho$ . This is also consistent with our overall objective of providing better incentives to rankers than to brokers. For both ETK and ETK+, we define the relay commission  $\rho_{RL}$  as 1 % of the query price  $\rho$ , thereby incentivizing brokers more than relay peers.

As we shall see shortly, the rewards to be assigned to the successful rankers are computed based on the value of  $\rho_R$ . Similarly, the penalties to be assigned to the unsuccessful rankers are also computed based on the value of  $\rho_R$ . The broker receives the penalty payments from the unsuccessful rankers, and sends the total amount of penalty payments back to  $QI$ . Thus, it is possible for the effective payment made by  $QI$  to the broker to be less than  $\rho$ .

Notably, as is common with currency-based approaches, there is a bootstrapping problem. That is, an MP must first earn currency by providing services, but at the beginning, no MP can request for those services because no MP has any currency yet. To address the bootstrapping problem, the system will provide some initial currency to every MP at the beginning.

##### 4.1 ETK

In ETK,  $\rho_R$  is equally divided among all the relevant items. Then each ranker, which successfully sent item  $i$ , receives a reward  $P_i$  that is equal to the total reward for item  $i$  divided by the total number  $f_i$  of successful rankers w.r.t. item  $i$ . Given that the top- $k$  result set is  $T_A$ ,  $P_i$  is computed as follows:

$$\forall i \in T_A : P_i = \frac{1}{f_i} \left( \frac{\rho_R}{k} \right) \quad (6)$$

The reward  $REW_{Rj}$  assigned to a given ranker  $Rj$  is the total amount that it obtains for each of its relevant items i.e., those that occur in the  $T_A \cap T_{Rj}$ , where  $T_{Rj}$  is the

individual rank list of  $R_j$ . Given the set  $S_{Ranker}$  of rankers, the computation of  $REW_{R_j}$  follows:

$$\forall j \in S_{Ranker} : REW_{R_j} = \sum_{i \in (T_A \cap T_{R_j})} P_i \quad (7)$$

ETK defines penalties based on the notion of *opportunity cost*. This is because for all successful items, which were not sent by ranker  $R_j$ ,  $R_j$  would have earned currency if it had sent those items. Hence, the penalty  $PEN_{R_j}$  assigned to  $R_j$  equals  $\sum P_i$ , where  $i$  represents items that occur in  $T_A - T_{R_j}$ . The computation of  $PEN_{R_j}$  follows:

$$\forall j \in S_{Ranker} : PEN_{R_j} = \psi \times \left[ \sum_{i \in (T_A - T_{R_j})} P_i \right] \quad (8)$$

where  $\psi$  is the factor that represents the trade-off between communication overhead and peer participation. If the value of  $\psi$  is high, communication overhead would reduce because peers would be wary of sending data to the broker due to the higher penalties assigned to unsuccessful rankers. However, this would also reduce peer participation. On the other hand, if the value of  $\psi$  is low, peer participation would increase albeit at the cost of increased communication overhead due to lower disincentives for sending items that do not contribute to the top- $k$  result. In this work, we set the value of  $\psi$  to 1.3, which implies that the penalties for sending unsuccessful items is 30 % more than the reward for sending successful items. This creates disincentives for sending out unsuccessful items, while keeping the peer participation at a reasonable level. We leave the determination of an optimal value for  $\psi$  to future work.

The net payment  $NET_{R_j}$  received by  $R_j$  is the difference between its total reward and its total penalty.  $NET_{R_j}$  is computed as follows:

$$\forall j \in S_{Ranker} : NET_{R_j} = REW_{R_j} - PEN_{R_j} \quad (9)$$

Now, based on the payoffs received,  $R_j$  will re-evaluate the selection probability of all the items in its individual  $T_{R_j}$ . ETK performs rank-weighted increase/decrease in  $\mu$  for each item, depending on whether the item is rewarded or penalized. For each item  $i$  in  $T_{R_j}$ , the value of  $\mu_{ij}$  is computed as follows:

$$\forall j \in S_{Ranker}, \forall i \in T_{R_j} :$$

$$\mu_{ij} = \begin{cases} \min(\mu_{ij} + \alpha_{up} \left( \frac{|T_{R_j}| - r_{ij}}{|T_{R_j}|} \right), 1), & \text{if } i \text{ is rewarded} \\ \max(\mu_{ij} - \alpha_{down} \left( \frac{|T_{R_j}| - r_{ij}}{|T_{R_j}|} \right), 0), & \text{if } i \text{ is penalized} \end{cases} \quad (10)$$

where  $r_{ij}$  is the rank of item  $i$  in  $T_{R_j}$ . Observe that,  $\mu_{ij}$  increases slightly for higher-rank items that received rewards but decreases significantly in case of a penalty. Similarly,  $\mu_{ij}$  increases significantly for lower-rank items that received rewards but decreases relatively slightly in case of

a penalty. Here,  $\alpha_{up}$  and  $\alpha_{down}$  represent the weight coefficients for assigning rewards and penalties respectively. ETK stipulates that  $0 < \alpha_{up}, \alpha_{down} \leq 1$  and  $\alpha_{up} < \alpha_{down}$  to ensure that penalties exceed rewards, thereby creating disincentives for rankers in terms of sending out items that are not relevant. In this work, we set the values of  $\alpha_{up}$  and  $\alpha_{down}$  to 0.1 and 0.3 respectively. We leave the determination of optimal values of  $\alpha_{up}$  and  $\alpha_{down}$  to future work.

## 4.2 ETK+

In ETK+,  $\rho_R$  is divided among all the items in the top- $k$  result  $T_A$  based on their respective rank-weights i.e., each item  $i$  with its associated rank  $r_i$  has weight  $w_i = (k - r_i)$ , where highest to lowest rank counts are from 0 to  $(k - 1)$ . Furthermore, total number  $W$  of weights of all items in  $T_A$  is computed as  $W = \sum_{i=1}^k w_i = k(k + 1)/2$ . Similar to ETK, each ranker, which successfully sent item  $i$ , receives a reward  $P_i$  that is equal to the total reward for item  $i$  divided by the total number  $f_i$  of successful rankers w.r.t. item  $i$ . Thus, in ETK+,  $P_i$  is computed as follows:

$$\forall i \in T_A : P_i = \frac{1}{f_i} \left( \frac{w_i}{W} \times \rho_R \right) \quad (11)$$

Consequently, rewards and penalties assigned to each ranker  $R_j$  are computed as in Eqs. 7 and 8 respectively, using the value of  $P_i$  from Eq. 11. Hence, the net payment received by  $R_j$  is computed by Eq. 9.

Now, each ranker  $R_j$  will re-evaluate the score (effectively the selection probability  $\mu$ ) of each item  $i$  in its top- $k$  rank list  $T_{R_j}$  on the basis of its received payoffs. The effective change in the selection probability of an item depends upon two factors: (a) the notion of item selection potential w.r.t. the risk profile ( $\delta$ ) (b) earning potential of the ranker  $R_j$ . Item selection potential increases as the *difference* between  $\mu$  and  $\delta$  increases. Average selection potential for rewarded and penalized items for each ranker  $R_j$  are computed as  $s_j$  and  $s'_j$  respectively. The computations of  $s_j$  and  $s'_j$  are shown below:

$$\forall j \in S_{Ranker} :$$

$$s_j = \frac{1}{|T_{R_j} \cap T_A|} \left[ \sum_{i \in (T_{R_j} \cap T_A)} (\mu_{ij} - \delta_j) \right] \quad (12)$$

$$s'_j = \frac{1}{|T_{R_j} - T_A|} \left[ \sum_{i \in (T_{R_j} - T_A)} (\mu_{ij} - \delta_j) \right] \quad (13)$$

where  $T_{R_j}$  is the top- $k$  rank list of  $R_j$ ,  $T_A$  is the top- $k$  result of a query  $Q$ ,  $\mu_{ij}$  is the selection probability of item  $i$  in  $T_{R_j}$  and  $\delta_j$  is the risk profile of  $R_j$ .



Earning potential  $e_j$  of each ranker  $R_j$  is a measure of its item selection efficiency.  $e_j = |(REW_{R_j} - PEN_{R_j}) / (REW_{R_j} + PEN_{R_j})|$ . Based on the payoff of each item  $i$  in  $T_{R_j}$ , the new (re-evaluated) value of  $\mu_{ij}$  is computed as follows:

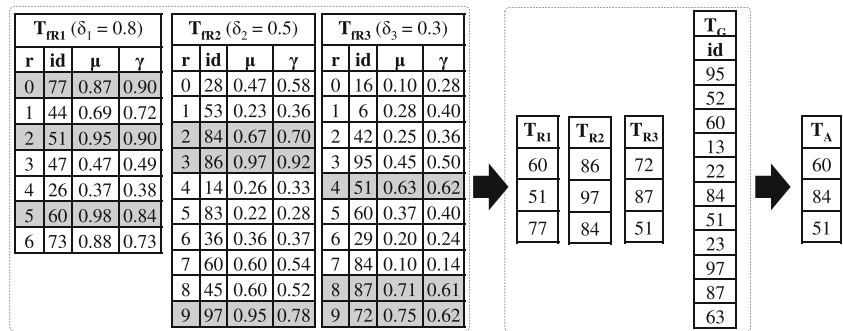
$$\forall j \in S_{Ranker}, \forall i \in T_{R_j} : \mu_{ij} = \begin{cases} \min(\mu_{ij} + \alpha_{up} \left(\frac{s_j + e_j}{2}\right), 1), & \text{if } i \text{ is rewarded} \\ \max(\mu_{ij} - \alpha_{down} \left(\frac{s_j + e_j}{2}\right), 0), & \text{if } i \text{ is penalized} \end{cases} \quad (14)$$

where  $\alpha_{up}$  and  $\alpha_{down}$  are the weight coefficients discussed in Eq. 10.

**Fig. 2** Illustrative example for ETK and ETK+

### 4.3 Illustrative example for ETK and ETK+

Figure 2 illustrates the computations in ETK and ETK+. In Figure 2a, observe how each ranker  $R$  computes the value of  $\gamma$  using Eq. 4 with  $w_1 = 0.2$  and  $w_2 = 0.8$ . For ranker  $R_1$ , the elements of  $T_{R_1}$  are shaded in grey i.e.,  $T_{R_1} = \{60, 51, 77\}$  because their respective values of  $\gamma$  exceed 0.8 ( $\delta_1 = 0.8$ ). Figure 2b depicts the payoff computations with  $\psi = 1.3$ . Observe that ETK+ assigns higher penalties (than ETK) to rankers for sending irrelevant items e.g., ETK+ assigned 97.50 to  $R_3$  as compared to 78.00 in



(a) Compilation of top- $k$  result  $T_A$  ( $k = 3$ )

$T_A$	$P_j$ (ETK)			$P_j$ (ETK+)				
	$\rho_R / k$	R1	R2	R3	$\rho_R / k$	R1	R2	R3
60	30	30	-	-	45	45	-	-
84	30	-	30	-	30	-	30	-
51	30	15	-	15	15	7.5	-	7.5
	<b>REW<sub>Ri</sub></b>	45	30	15	<b>REW<sub>Ri</sub></b>	52.5	30	7.5

Ranker	$T_A - T_{Ri}$	$PEN_{Ri}$ (ETK)	$PEN_{Ri}$ (ETK+)
R1	{84}	$1.3 \times 30 = 39.00$	$1.3 \times 30 = 39.00$
R2	{60, 51}	$1.3 \times (30+15) = 58.50$	$1.3 \times (45+7.5) = 68.25$
R3	{60, 84}	$1.3 \times (30+30) = 78.00$	$1.3 \times (45+30) = 97.50$

(b) Computation of rewards and penalties

R1 ( $\delta_1 = 0.8$ )				R2 ( $\delta_2 = 0.5$ )				R3 ( $\delta_3 = 0.3$ )			
id	$\mu$	$\mu_{ETK}$	$\mu_{ETK+}$	id	$\mu$	$\mu_{ETK}$	$\mu_{ETK+}$	id	$\mu$	$\mu_{ETK}$	$\mu_{ETK+}$
77	0.87	0.57	0.82	28	0.47	0.47	0.47	16	0.10	0.10	0.10
44	0.69	0.69	0.69	53	0.23	0.23	0.23	6	0.28	0.28	0.28
51	0.95	1.00	0.96	84	0.67	0.75	0.71	42	0.25	0.25	0.25
47	0.47	0.47	0.47	86	0.97	0.76	0.89	95	0.45	0.45	0.45
26	0.37	0.37	0.37	14	0.26	0.26	0.26	51	0.63	0.69	0.69
60	0.98	1.00	0.99	83	0.22	0.22	0.22	60	0.37	0.37	0.37
73	0.88	0.88	0.88	36	0.36	0.36	0.36	29	0.20	0.20	0.20
				60	0.60	0.60	0.60	84	0.10	0.10	0.10
				45	0.60	0.60	0.60	87	0.71	0.65	0.53
				97	0.95	0.92	0.87	72	0.75	0.72	0.57

(c) Updates in the selection probabilities

ETK. Figure 2c depicts the re-evaluation of the selection probability  $\mu$  using Eq. 14 with  $\alpha_{up} = 0.1$  and  $\alpha_{down} = 0.3$ .

## 5 ETG: A peer group-based economic incentive scheme in E-Top

This section discusses the group-based ETG scheme.

### 5.1 Peer groups in ETG

We define a **peer group** as a set of MPs, which collaborate in answering a given top- $k$  query. Recall that in our application scenarios, a query-issuing MP  $QI$  may try to find top- $k$  restaurants with “happy hours” nearby itself. MPs that are moving nearby  $QI$  form *ad hoc groups* for answering this query. Thus, groups are formed based on region. The universe is initially divided into rectangular cells of equal area, and all the MPs moving within a particular cell constitute a group. In case there are not sufficient members in a region at a given point of time, the region can be enlarged based on some minimum spatial density threshold. Conversely, group region can be shrunk based on a maximum density threshold. This work does not specifically focus on how groups are formed, but existing works [16] can be used in conjunction with our work for group formation purposes. Notably, ETG stipulates that each MP can belong to any one group at a given point of time, thereby ensuring that any MP obtains its payoff from not more than one group leader for a given top- $k$  query.

In ETG, each group has a group leader, which facilitates top- $k$  query processing within the group. A group leader should be an MP with relatively high energy, bandwidth and processing capacity. Its mobility is typically limited and it stays within the region. In our application scenarios, some of the restaurant managers in the vicinity of the query location can be the group leaders. The group leader receives a percentage of the group rewards as a commission, thereby incentivizing it to participate. In this work, we set the group leader’s commission to 5 % of the group reward.

Query processing in ETG proceeds via group members sending their individual list of top- $k$  items to the group leader. The group leader selects the top- $k$  items to be sent to the broker based on relative frequencies of items in these individual top- $k$  lists by sorting the items in descending order of frequency. Then the group leader sends the  $k$  items with the highest frequencies to the broker. Ties in item frequencies are resolved arbitrarily by the group leader.

ETG uses either ETK or ETK+ for performing the following two economic functions in the top- $k$  query processing. First, brokers assign payoffs to the groups based on either ETK or ETK+. (These payoffs are allocated by the group leader among the group members, as we shall

describe shortly.) Second, upon receiving the payoffs, group members modify their item selection probabilities as in either ETK or ETK+. Thus, ETG works in conjunction with either of these schemes. In our performance study, we have first shown the performance of ETG in conjunction with both ETK and ETK+, and then presented the remaining results corresponding to ETG in conjunction with ETK+.

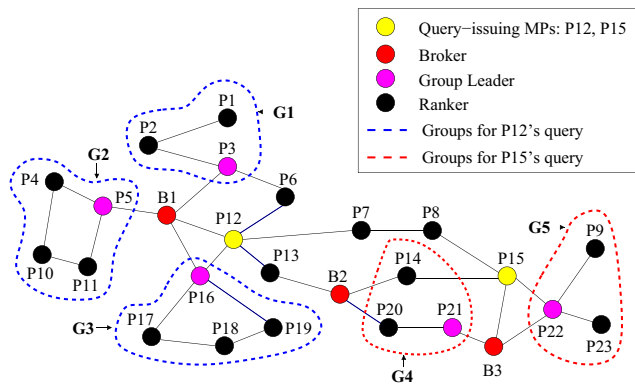
Recall that in ETK and ETK+, any given ranker can only participate for a top- $k$  query  $Q$  if it hosts at least  $k$  items related to  $Q$ . In case of ETG, this criterion is relaxed because even if a ranker does not host  $k$  items related to  $Q$ , it can still participate in the top- $k$  query processing as long as the group hosts at least  $k$  items related to  $Q$ . Thus, ETG increases the opportunities for rankers to contribute to the top- $k$  query processing, thereby providing increased opportunities for rankers to earn currency and also providing additional incentives towards ranker collaboration.

Group-based collaboration provides better incentives for MPs to answer top- $k$  queries. When an MP  $M$  acts individually in answering top- $k$  queries, it can incur significant penalties due to sending irrelevant items to the broker. This may discourage  $M$  from answering queries. As we shall see shortly, when an MP participates in a group, both rewards as well as penalties are distributed among the group members. In effect, this encourages MPs to provide answers to top- $k$  queries because in case its answer turns out to be irrelevant, it risks a lower amount of individual penalties due to sharing of penalties among group members. Group-based collaboration also increases the probability of obtaining rewards because collective top- $k$  answers from the members of the group are likely to be of higher quality (i.e., more relevant and accurate) than individual answers. As we shall see shortly, this is made possible by the ‘filtering’ performed by the group leader on the individual top- $k$  lists sent by the group members. In essence, group-based collaboration leads to better economy of scale and better results than MPs acting individually.

### 5.2 Illustrative example of peer groups in ETG

Figure 3 depicts an illustrative example of an instance of network topology in ETG. Now we shall use Fig. 3 to illustrate the concept of groups as well as the steps involved in top- $k$  query processing under the ETG scheme. In Fig. 3,  $P12$  and  $P15$  are the query-issuers,  $P1$  to  $P23$  (except  $P12$  and  $P15$ ) represent the rankers, and  $B1$  to  $B3$  indicate the brokers. The groups corresponding to the queries of  $P12$  and  $P15$  are  $\{G1, G2, G3\}$  and  $\{G4, G5\}$  respectively. The group leaders of  $G1$  to  $G5$  are  $P3, P5, P16, P21$  and  $P22$  respectively.

In Fig. 3, observe that multiple brokers exist. However, given a query  $Q$ , only one of them act as the broker for  $Q$ .



**Fig. 3** Illustrative example of peer groups in ETG

Broker selection for  $Q$  is based on the value of  $\eta$  that is computed from Eq. 3, as discussed earlier in Section 3. For example, in case of  $P12$ 's query, the candidate brokers are  $B1$  and  $B2$ . For simplicity, suppose the energy, bid price and currency of  $B1$  and  $B2$  are equal. In this case,  $B1$  will be selected as the broker because it is nearer to the query-issuer  $P12$ . Similarly, in case of  $P15$ 's query, holding all other factors constant,  $B3$  will be selected as the broker because it is nearer (than  $B2$ ) to  $P15$ .

Now let us examine the processing of  $P12$ 's query.  $P12$  first sends out a broadcast query to list candidate brokers in its vicinity. Based on the respective values of the broker scores  $\eta$ , suppose  $P12$  selects broker  $B1$  as the broker for processing its query. Then  $B1$  sends out the query to groups  $G1$  to  $G3$ , which are nearby the query location. The group leaders (i.e.,  $P3$ ,  $P5$ ,  $P16$ ) in these groups consolidate the top- $k$  results from their respective groups and send the results back to the broker. Upon receiving the results from the group leaders,  $B1$  compares with its global top- $k$  list to generate the final top- $k$  list, which it sends to the query-issuer  $P12$ . At this stage,  $B1$  also assigns payoffs to the groups. Notably, the broker's assignment of payoffs to the groups is done based on either ETK or ETK+. Then  $B1$  sends the top- $k$  results to  $P12$  and obtains payment from  $P12$ . Finally,  $B1$  sends the respective payments to the group leaders according to its assigned payoffs.

### 5.3 Allocation of payoffs among group members in ETG

Given a top- $k$  query  $Q$ , the resulting payoff for the group has to be allocated among the group members such that they are incentivized towards group-based collaboration.

We define a group member as a **participant** in query  $Q$  if at least one of the top- $k$  items that it sent to the group leader is selected by the group leader in the top- $k$  list that the group leader propagates to the broker. On the other hand, we define a group member as a **contributor** to  $Q$  if at least one of the top- $k$  items that it sent to the group leader occurs

in the final top- $k$  list that is selected by the broker. Thus, the set of contributors is a subset of the set of participants for a given top- $k$  query. Based on the notions of contributors and participants, we propose three schemes for payoff allocation among group members. In all these proposed schemes, penalties are divided among all the participants for a query  $Q$ , while rewards are distributed only among the contributors. Thus, the three approaches differ in the way in which group rewards are allocated.

Let  $n_P$  be the number of participants for  $Q$ . Let  $PEN_G$  represent the penalty for the group corresponding to  $Q$ . The penalty  $PEN_j$  for participant  $j$  is computed as follows:

$$PEN_j = \frac{PEN_G}{n_P} \quad (15)$$

Notably, all our three proposed approaches for payoff allocation compute the penalty  $PEN_j$  incurred by participant  $j$  by means of Eq. 15 above.

Observe how ETG incentivizes MP participation in groups (as opposed to MPs acting individually) by reducing potential penalties for group members in two ways. First, the 'filtering' of top- $k$  items performed by the group leader implies that even if a group member  $P$  had sent one or more items (to the group leader), which do not occur in the final top- $k$  result selected by the broker,  $P$  incurs no penalty for such irrelevant items as long as the group leader does not send them to the broker. In effect, being part of a group shields the MP from incurring penalties to a certain extent. Moreover, since the group leader receives top- $k$  items from multiple group members, it has a broader (and more collective) view of the likely top- $k$  results than individual group members. This increases the likelihood of the group leader's 'filtering' process being more effective in predicting the final top- $k$  results than if the top- $k$  predictions were done by individual MPs.

Second, the sharing of penalties across participants reduces the penalties incurred by those members, which sent out irrelevant items, which were selected by the group leader and which did not occur in the final top- $k$  results. This does not incentivize group members to frivolously send out irrelevant items to the group leader because the items should have at least some chance of occurring in the final top- $k$  result for the group leader to have selected them.

Incidentally, the equal sharing of penalties across all participants may result in increased penalties for some of the participants, especially for the contributors. For example, even if a contributor had not sent out any irrelevant items, it still has to pay the penalty due to irrelevant items being sent by some of the other participants. However, the cost of such possible additional penalties is offset by the benefit obtained by the contributor(s) in terms of avoiding potential penalties due to the group leader's filtering process. This

explains the rationale for dividing penalties equally among all the participants for a given query  $Q$ .

The rationale for distributing rewards only among the contributors is two-fold. First, it discourages free-riding within the group since a peer has to contribute to the final top- $k$  query result in order to qualify for obtaining a share of the group reward. Observe that if the group reward were to be distributed across all the participants, it would act as a disincentive for the contributors since they would earn lower amounts of currency. Second, it recognizes the contribution of the contributors to the group revenue, thereby incentivizing peer contributions to the group.

Now we shall discuss the three approaches that ETG deploys for allocation of group rewards among contributors for a given top- $k$  query  $Q$ .

### 5.3.1 Equal allocation of payoff (EQ)

In EQ, each contributor obtains an equal share of the group reward  $REW_G$  for  $Q$ . Given that  $n_C$  is the number of contributors for  $Q$ , the reward  $REW_j$  for contributor  $j$  is computed as follows:

$$REW_j = \frac{REW_G}{n_C} \tag{16}$$

Notably, a major drawback of the EQ approach is that the allocation of group reward is not based on the contribution of individual contributors since it does not consider the number of items contributed by each of them.

### 5.3.2 Item contribution-based allocation of payoff (ICON)

To address the drawback of EQ, we propose ICON. In ICON, each contributor obtains a share of the group reward  $REW_G$  based on the number of items that it contributed

to the final top- $k$  query result. ICON computes the reward  $REW_j$  for contributor  $j$  as follows:

$$REW_j = \frac{|C_j|}{\sum_{g=1}^{n_C} |C_g|} \times REW_G \tag{17}$$

where  $C_j$  is the set of items that MP  $j$  has contributed to the final top- $k$  result, and  $n_C$  represents the total number of contributors corresponding to  $Q$ .

ICON suffers from the drawback that the allocation of group rewards is not based on the actual revenue earned from the item (i.e., the reward that is assigned to the item). For example, suppose contributor  $P1$  has contributed three items, while contributor  $P2$  has contributed only one item. However, the revenue earned by the group from the one item sent by  $P2$  could be higher than that of the total revenue earned from the three items contributed by  $P1$ .

### 5.3.3 Revenue contribution-based allocation of payoff (RCON)

To address ICON’s drawback, we propose RCON. In RCON, each contributor obtains a share of the group reward  $REW_G$  based on the revenue earned from the items that it contributed to the final top- $k$  result. RCON computes reward  $REW_j$  for contributor  $j$  as follows:

$$REW_j = \frac{\sum_{i \in C_j} \lambda_i}{\sum_{g=1}^{n_C} (\sum_{i \in C_g} \lambda_i)} \times REW_G \tag{18}$$

where  $C_j$  is the set of items that MP  $j$  has contributed to the final top- $k$  result,  $\lambda_i$  represents the revenue earned for a given item  $i$ , and  $n_C$  represents the total number of contributors for  $Q$ .

### 5.3.4 Illustrative example of group reward allocation among contributors

Table 1 depicts an illustrative example of allocation of rewards among contributors.

**Table 1** Illustrative example of group reward allocation among contributors in ETG

Item ID (i)	Item Reward ( $\lambda_i$ )	Contributor (j)	Relevant Item Set ( $C_j$ )	Contributor $j$ 's Reward ( $REW_j$ )			
				EQ	ICON	RCON	
1	60	$P1$	{1, 3}	90/3 = 30	(2/5) × 90 = 36	60 + 10 = 70	(70/160) × 90 = 39.375
2	20	$P2$	{2, 3}	90/3 = 30	(2/5) × 90 = 36	20 + 10 = 30	(30/160) × 90 = 16.875
3	10	$P3$	{1}	90/3 = 30	(1/5) × 90 = 18	60	(60/160) × 90 = 33.750
	$\sum \lambda_i =$ $REW_j = 90$		$\sum  C_j  = 5$				<i>Nett</i> = 160

Consider a top-3 query, for which the result set comprises items {1, 2, 3}. Suppose the relevant items sent by contributors  $P_1$ ,  $P_2$  and  $P_3$  are {1, 3}, {2, 3} and {1} respectively. As Table 1 indicates, the rewards for items {1, 2, 3} are {60, 20, 10} currency units respectively. Hence, the total group reward  $REW_G$  is the sum of these individual item rewards i.e., 90 currency units. (For simplicity, we ignore the group leader's commission for this example.)

For EQ, the reward is distributed equally among the contributors, hence  $P_1$ ,  $P_2$  and  $P_3$  would each obtain a reward of 30 currency units. For ICON, the number of relevant items sent by { $P_1$ ,  $P_2$ ,  $P_3$ } are {2, 2, 1} respectively. Hence, the reward for  $P_1$  is  $(2/(2 + 2 + 1)) * 90$  i.e., 36 currency units. In case of RCON, the rewards for  $P_1$ 's relevant (sent) items {1, 3} are {60, 10} currency units respectively, thereby resulting in a total of 70. Similarly, the corresponding totals for  $P_2$  and  $P_3$  are 30 and 60 respectively. Thus, RCON computes the reward of  $P_1$  based on the weighted average of item revenues earned. Hence,  $P_1$  obtains  $(70/(70 + 30 + 60)) * 90$  i.e., a reward of 39.375 currency units.

Observe the difference in rewards obtained by  $P_2$  and  $P_3$  under ICON and RCON. In case of ICON,  $P_2$  obtains double the reward of  $P_3$ , even though  $P_3$  contributed more revenue to the group. This highlights the drawback of ICON. Observe how RCON alleviates this drawback by assigning  $P_3$  a higher amount of reward than  $P_2$ .

## 6 Performance evaluation

This section reports our performance evaluation by means of simulation in OMNeT++ [2]. To the best of our knowledge, there is no publicly available real dataset for our application scenarios. Furthermore, observe that it is not practically feasible to study the performance of our proposed approaches in a real-world setting using real mobile devices at any reasonably large scale. Additionally, the challenges to doing such a real-world performance evaluation are exacerbated by the fact that peers are autonomous and their behaviours are generally unpredictable. Hence, we have simulated the whole scenario by generating the queries with M-P2P architecture on OMNeT++ platform.

In our OMNeT++ simulations, we have considered reasonable assumptions about peers' behaviors such as peers' movement, energy, bandwidth, memory etc. We have looked into the literature [20, 21] to understand the different parameters. Based on our understanding of our application environment, we have selected these parameters. In essence, we believe that we have made our best efforts to capture/simulate the peers' behavior and the environment as closely as possible w.r.t. real-world scenarios. Moreover, we

have indicated how our proposed approaches perform under those simulated settings.

OMNeT++ platform provides the crucial functionality such as creation of heterogeneous devices/nodes (i.e., mobile peers) such as PDA/laptop/smart-phones etc., their range of communication, wireless message exchanges, multicast/broadcast messaging, nodes' location, nodes' database, etc. Hence, the simulation has locations of MPs at any point of time, which are available during the course of our experiments.

In our simulations, MPs move according to the *Random Waypoint Model* [3] within a region of area 1000 metres  $\times$  1000 metres. The *Random Waypoint Model* is appropriate for our application scenarios, which generally involve random movement of peers. For example, people looking for top- $k$  restaurants generally move randomly i.e., they do not follow any specific mobility pattern. Our experiments use a total of 100 MPs. In our experiments, we consider the following type of example scenario. Each data item pertains to information about a given restaurant i.e., the unique identifier of the restaurant and some text content (e.g., telephone number, address etc.) about the restaurant. The MPs store such data items i.e., information about restaurants. Our simulations consider a total of 100 restaurants. Thus, each data item pertains to information about one of these 100 restaurants. Here, each MP contains 20 to 25 data items i.e., information about any 20 to 25 randomly selected restaurants from the total set of 100 restaurants. Each top- $k$  query aims at retrieving the top- $k$  list of restaurants from among these 100 restaurants. The default communication range of all MPs is a circle of 120 metre radius.

Table 2 summarizes the parameters used in our performance evaluation. Query-issuers are selected randomly from among all the MPs in the network. The OMNeT++ simulation provides the location data of the query-issuing peer, which is available during the course of our experiments. The number of such top- $k$  queries issued in the network per time unit is 10, the query deadline  $\tau_Q$  being varied randomly between 3 to 5 time units. Query price  $\rho$  is chosen randomly in the range of 100 to 500 currency units. Broker commission  $\rho_B$  and relay commission  $\rho_{RL}$  are respectively set to 10 % and 1 % of  $\rho$ . For ETG, group leader's commission is set to 5 % of the group reward for a given query. Initial energy of an MP is selected to be randomly in the range of 90000 to 100000 energy units. Sending and receiving a message require 1.5 and 1 energy units respectively.

Recall that each ranker is associated with a risk profile  $\delta$ . The number of MPs with the values of  $\delta$  as 0.3 (high-risk), 0.5 (medium-risk) and 0.8 (low-risk) are 27, 43 and 30 respectively. For all our experiments, the economic parameters are set as follows: (a) the values of weight coefficients  $w_1$  to  $w_4$  for computing the broker score  $\eta$  in Eq. 3 are each

**Table 2** Parameters of our performance study

Parameter	Default Value	Variations
$k$	8	4, 12, 16, 20, 24
Number of MPs ( $N_{MP}$ )	100	20, 40, 60, 80
Percentage of brokers ( $P_B$ )	20 %	10 %, 30 %, 40 %, 50 %
Queries/time unit		10
Communication Range (CR)	120m	40m, 80m, 160m, 200m
Percentage of MP failures ( $P_F$ )	20 %	10 %, 30 %, 40 %, 50 %
Group size (quantified by $S_G$ )	30 %	10 %, 20 %, 40 %, 50 %
Bandwidth between MPs	28 Kbps to 100 Kbps	
Initial energy of an MP	90000 to 100000 energy units	
Memory space of each MP	8 MB to 10 MB	
Speed of an MP	1 meter/s to 10 meters/s	
Size of a data item	50 Kb to 350Kb	

set to 0.25 (b) the values of weight coefficients  $w_1$  and  $w_2$  for computing the item score  $\gamma$  in Eq. 4 are set to 0.2 and 0.8 respectively (c) the penalty factor  $\psi$  (see Eq. 8) is set to 1.3 (d) the values of  $\alpha_{up}$  and  $\alpha_{down}$  for item selection probability re-evaluation (see Eqs. 10 and 14) are set to 0.1 and 0.3 respectively.

In our simulation, we consider a pre-defined global top- $k$  rank list  $T_G$  (of all data items), which is same across all the broker MPs.  $T_G$  was created based on a random ranking of these 100 restaurants. Notably, the ranking of restaurants in  $T_G$  is static, and it does not change over the course of our experiments. This is consistent with practice e.g.,  $T_G$  could be determined using a standard ranking list for restaurants such as Michelin's guide to restaurants. Such guides are published only periodically (quarterly, semi-annually, annually etc.). Notably, such periods far exceed the period over which our simulations were conducted.

In our simulation, suppose a given peer has  $x$  data items i.e., information about 20 different restaurants. The data items assigned to a given peer are determined randomly from the data items corresponding to the 100 restaurants. The selection probability value for each data item at a given peer is initially set randomly on a scale of 0 to 1. These selection probabilities determine the ranking of the data items at a given peer. Notably, the ranking of the data items in  $T_A$  will change (due to temporal dynamism) based on item re-ranking as discussed previously for our proposed schemes. This explains how the numbers are generated for  $T_A$  in our simulations.

Query database is created initially in the simulation. This database is used as synthetic query dataset for all our experiments. Query database has a table, which stores top- $k$  query's information such as unique ID,  $k$  value, price, start-time, mobile peer's ID and deadline. Similarly, we

have also created a database of all MPs, which consists of the information such as MP's unique ID, type (i.e., broker/relay/ranker) number of send/receive messages by MP, MP's top- $k$  data item list (i.e.,  $T_R$ ) with its relevant information (e.g., item id, rank, selection probability etc.), energy. In our simulation, we kept broker MPs as pre-defined, while other MPs may change role to relay/ranker depending upon the query.

Finally, the result set for all the queries were stored as result database in the simulation. A table related to top- $k$  queries' result consists of information such as query ID, answer list  $T_A$  (i.e., top- $k$  data item list as query result), response time (i.e., query's end-time – query's start-time), precision, completed/failed and list of MPs who participated in that query. We compute precision by comparing query result's top- $k$  list with  $T_G$ 's top- $k$  list.

Performance metrics are average response time (ART), precision rate (PREC), query completeness rate (QCR) and communication traffic (MSG). We define a query as **completed** if the broker receives at least  $k$  items from individual rankers (or group leaders in case of ETG) within 70 % of the query deadline time  $\tau_Q$ . Notably, a broker may fail to receive at least  $k$  items due to reasons such as ranker unavailability and network partitioning. (Queries that are not completed are deemed to be query failures.) We compute ART only for the *completed* queries.  $ART = \frac{1}{N_C} \sum_{q=1}^{N_C} (t_f - t_0)$ , where  $t_0$  is the query-issuing time,  $t_f$  is the time of the query result reaching the query-issuer, and  $N_C$  is the total number of *completed* queries. We compute ART in simulation **time units (t.u.)**.

PREC is the average precision rate over all the queries. Suppose  $T_{A_q}$  is the top- $k$  query result and  $T_{G_q}$  is the global top- $k$  rank list of the respective broker for a query  $q$ . To obtain PREC for  $q$ , we measure the number of items in  $T_{A_q}$  which also occur in  $T_{G_q}$ ; then

we divide by the number of items in  $T_{G_q}$ . Notably,  $PREC$  is computed only for *completed* queries. Thus,

$$PREC = \frac{1}{N_C} \sum_{q=1}^{N_C} \left( \frac{|T_{G_q} - T_{A_q}|}{|T_{G_q}|} \right) \times 100.$$

QCR is the ratio of total number  $N_C$  of completed queries to the total number  $N_Q$  of queries.  $QCR = (N_C/N_Q) \times 100$ . We define  $MSG$  as the total number of messages incurred for query processing during the course of the experiment. Thus,  $MSG = \sum_{q=1}^{N_Q} M_q$ , where  $M_q$  is the number of messages incurred for the  $q^{th}$  query.

To the best of our knowledge, none of the existing top- $k$  query processing schemes in M-P2P environment uses incentives. Hence, for purposes of meaningful comparison, we adapt an existing non-incentive-based top- $k$  processing scheme for MANETs. We designate this scheme as NETK (Non-Economic Top- $K$ ), proposed in [20]. Although NETK does not provide incentives to the MPs, it is closest to our top- $k$  query processing scheme.

In NETK, each MP that receives a query message sends back a fixed number,  $R$ , of its holding data items with the  $R$  highest scores. If each MP finds that the total number of data items received from all its successor MPs and its own data items with the  $R$  highest scores becomes larger than  $k$  (i.e., top- $k$  data items), it only sends  $k$  data items with the highest scores among those data items to its predecessor. Notably, NETK suffers from the serious drawback of not being able to encourage peer participation in top- $k$  query processing since it does not provide incentives. To strengthen NETK, we adapted NETK to our scenario with  $R = \lceil k/50 \rceil$  (i.e., 50 % of top- $k$  values are allowed to send towards contributing into top- $k$  query) because at this value of  $R$ , NETK has above-average peer participation (based on the results of our preliminary experimental observations), thereby making NETK a fairly efficient approach in itself. Furthermore, NETK does not incorporate the notion of item re-ranking as no feedback has been sent back to the MPs, who participated into top- $k$  query processing.

## 6.1 Effect of peer groups with ETK and ETK+

Recall that ETG uses either ETK or ETK+ for performing some of the economic functions (e.g., assignment of payoffs from broker to group leader) during top- $k$  query processing. We designate these variations as ETG(K) and ETG(K+) corresponding to ETK and ETK+ respectively.

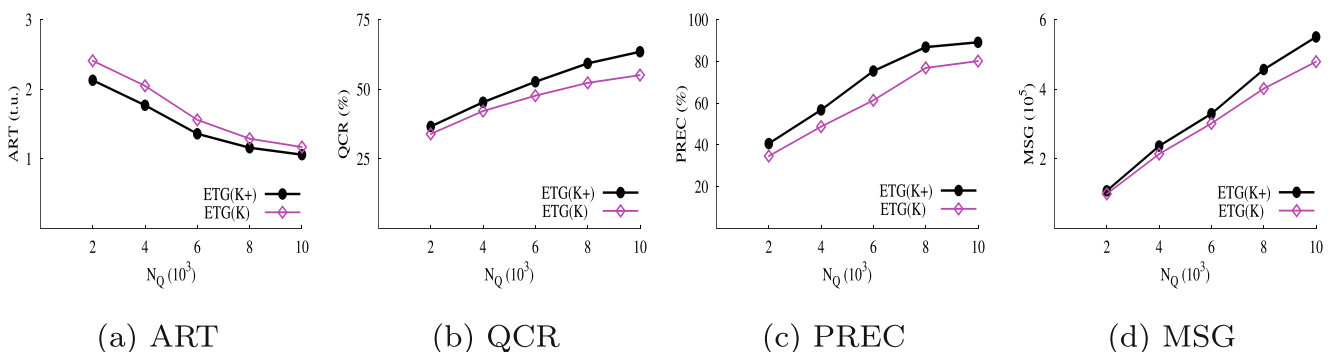
Figure 4 depicts the results. ETG(K+) outperforms ETG(K) due to two reasons. First, ETK+'s rank-weighted payoff strategy provides better incentivization than the uniform incentivization provided by ETK. Second, ETK+ provides more effective re-evaluation of the item selection probability  $\mu$  by tying  $\mu$  to payoffs associated with rankers' items. In contrast, ETK does not directly link  $\mu$  to payoffs. However, ETG(K+) incurs more  $MSG$  due to group communication overhead. For the remainder of this section, we show the performance of ETG in conjunction with ETK+.

## 6.2 Effect of variations in the percentage of brokers

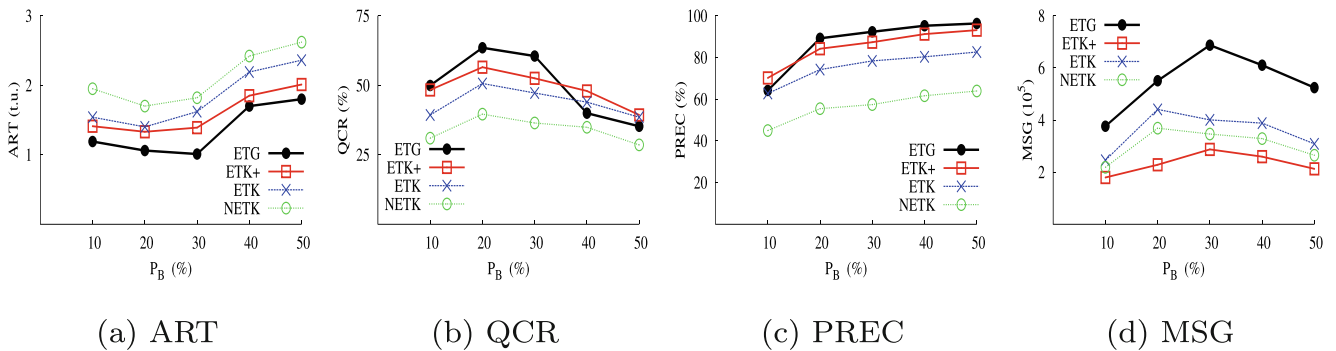
We performed an experiment to determine the percentage  $P_B$  of brokers in the network. Figure 5 depicts the results.

As  $P_B$  is increased from 10 % to 20 %, ART decreases and QCR increases for all the schemes. This is because the involvement of more brokers increases the probability that a given query is processed by at least one of the brokers. Notably, the sum total of the number of brokers and the number of rankers is fixed. Hence, when  $P_B$  is increased beyond 30 %, the number of rankers reduces, thereby reducing QCR and increasing ART (due to more hop-counts required to reach the rankers). Interestingly, beyond  $P_B = 40$  %, ETG performs slightly worse QCR than both ETK and ETK+ due to a significantly decreased number of rankers, which make group formation difficult.

As  $P_B$  increases,  $PREC$  increases due to the involvement of more brokers for all the schemes. However,  $PREC$  exhibits a saturation effect beyond  $P_B = 30$  % due to



**Fig. 4** Effect of peer groups with ETK and ETK+



**Fig. 5** Effect of variations in the percentage of brokers

reduced number of rankers. As  $P_B$  is increased till 30 %, MSG increases for all the schemes due to the involvement of more brokers. However, beyond  $P_B = 30$  %, MSG decreases due to reduced number of rankers. Based on the results, we set the percentage of brokers to 20 % so that we can obtain good performance of E-Top with reasonable communication traffic.

### 6.3 Performance of ETK and ETK+

We conducted an experiment using the default values of the parameters in Table 2.

Figure 6 depicts the results. As more queries are processed, performance improves for ETK, ETK+ and ETG due to incentives and effective item re-ranking. However, the performance eventually plateaus due to network partitioning and unavailability of some of the rankers. ETK+ outperforms ETK because it provides better incentivization and more effective re-evaluation of the item selection probability, as explained for the results in Fig. 4. ETG outperforms both ETK and ETK+ due to better incentives for group-based collaboration and effective payoff sharing among group members.

NETK performs worse than that of ETK in terms of ART, QCR and PREC due to less ranker participation (owing to the lack of incentives), which may cause inadequate items

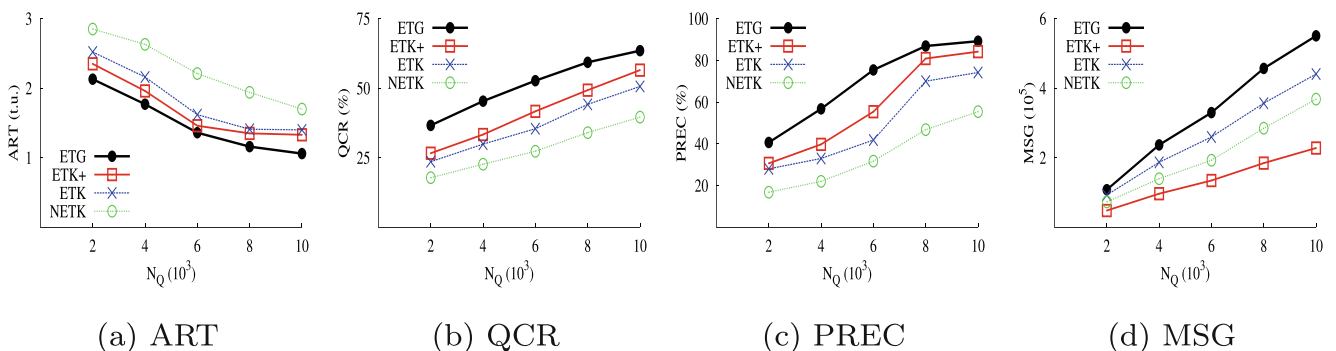
to generate the final top- $k$  result. Since the broker does not always receive at least  $k$  items from rankers, NETK results in a significant number of incomplete query results.

MSG increases over time for all the schemes as more queries are being processed since MSG is a cumulative metric. Interestingly, NETK incurs lower MSG than the other schemes due to lower levels of ranker participation in the absence of item re-ranking. ETK+ incurs lower MSG than ETK because ETK+ assigns higher amount of penalties (as compared to ETK) to rankers that send irrelevant items, hence fewer rankers reply to the broker in case of ETK+. ETG incurs higher MSG than both ETK and ETK+ due to periodic communication between group members for exchanging their own individual top- $k$  lists and for sharing payoffs.

### 6.4 Effect of variations in $k$

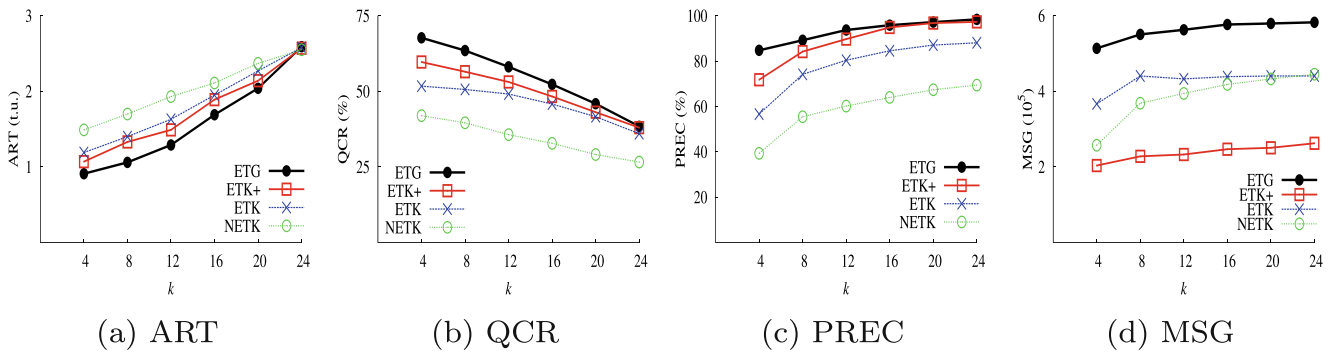
Figure 7 depicts the effect of variations in  $k$ .

As  $k$  increases, QCR decreases for all the schemes because relatively fewer rankers would be capable of participating in the top- $k$  query processing. This is because a ranker is allowed to send its top- $k$  result only if it hosts at least  $k$  items pertaining to the query. ART increases due to longer query paths as nearby rankers are unable to provide enough relevant items. The performance gap (in



**Fig. 6** Performance of ETK & ETK+





**Fig. 7** Effect of variations in  $k$

terms of ART and QCR) between ETK, ETK+ and ETG keeps decreasing with increase in  $k$  due to decreased ranker participation.

As  $k$  increases, PREC increases for all the schemes due to increase in the probability of the items (sent by the rankers) being relevant to the top- $k$  result. For example, if  $k = 4$ , an item will contribute to the top- $k$  if it matches one of the four items in the broker's global top- $k$  list  $T_G$ . However, if  $k = 24$ ,  $T_G$  has 24 items, hence the ranker-sent item has a better chance of a 'match' with any one of the items in  $T_G$ . ETG and ETK+ exhibit comparable PREC beyond  $k = 12$  because their incentives result in the same rankers sending the top- $k$  results at these higher values of  $k$ . PREC also eventually plateaus for all the schemes after  $k = 12$  due to peer mobility, frequent network partitioning and unavailability of some of the rankers.

As  $k$  increases, MSG increases for all the schemes due to longer query paths arising from less ranker participation. However, MSG eventually plateaus at  $k = 12$  because the increased number of hops required to reach the relevant rankers is offset by the decreased number of relevant rankers.

### 6.5 Effect of variations in the number of MPs

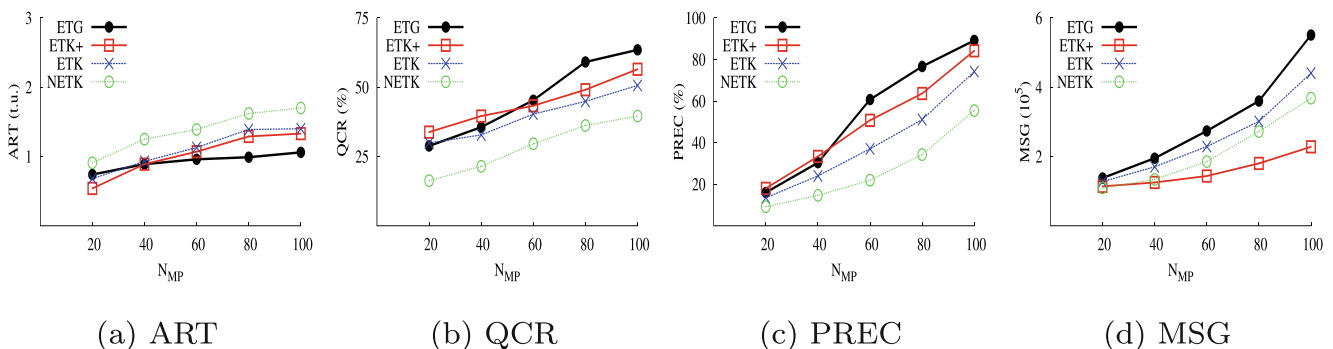
We conducted an experiment to examine the scalability of our proposed schemes. Figure 8 depicts the results. As the

number  $N_{MP}$  of MPs increases, ART increases for all the schemes due to larger network size. As  $N_{MP}$  increases, QCR and PREC increase because larger network implies more rankers. Observe that the performance of NETK is worse than that of ETK, ETK+ and ETG due to lower levels of ranker participation in the absence of item re-ranking, as explained for the results in Fig. 6.

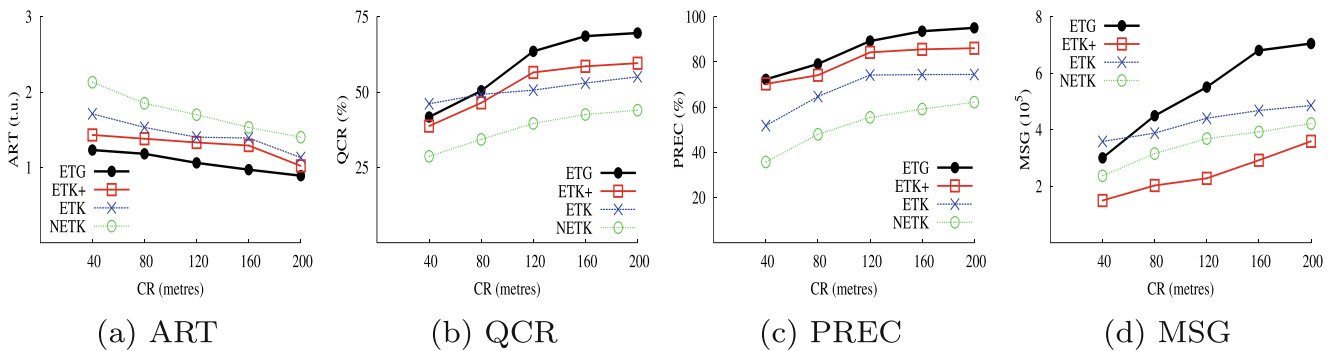
ETG outperforms ETK and ETK+ due to more pronounced effect of peer group collaboration when  $N_{MP}$  exceeds 40. However, below  $N_{MP} = 40$ , ETG performs slightly worse than ETK+ due to limited effect of group collaboration. MSG increases for all the schemes due to larger network size.

### 6.6 Effect of variations in the communication range

The results in Fig. 9 depict the effect of variations in the communication range CR of the MPs. Increase in CR has the effect of bringing the MPs 'nearer' to each other. Hence, performance improves for all the schemes due to data items becoming 'nearer' and more accessible to query-issuers. Thus, relatively fewer queries fail due to the maximum TTL criteria of 6 hops as more MPs come within range to answer queries. Observe that the rate of decrease in ART is not necessarily uniform because of deviations arising from bandwidth differences at MPs. QCR and PREC exhibit a saturation effect for all the schemes



**Fig. 8** Effect of variations in the number of MPs



**Fig. 9** Effect of variations in the communication range

beyond  $CR = 160$  metres due to unavailability of some of the rankers.

As  $CR$  increases,  $MSG$  increases for all the schemes because the increased reachability of the MPs increases communication among them. With increase in  $CR$ , a lower number of messages are required to reach a given MP, thereby decreasing  $MSG$ . However, more MPs become involved in the processing of a given query, thereby increasing  $MSG$ . These two opposing effects somewhat offset each other at higher values of  $CR$ , thereby explaining the reason why  $MSG$  eventually plateaus. Interestingly,  $ETG$  incurs lower  $MSG$  than  $ETK$  at  $CR = 40$  metres. This occurs because at such low values of  $CR$ , the MPs are in effect ‘far’ from each other, thereby reducing the effectiveness of group collaboration. Consequently, a lower number of messages are required for group interactions.

### 6.7 Effect of MP failures

MPs can fail due to reasons such as depletion of their limited energy resources. Figure 10 depicts the results of the effect of MP failures.

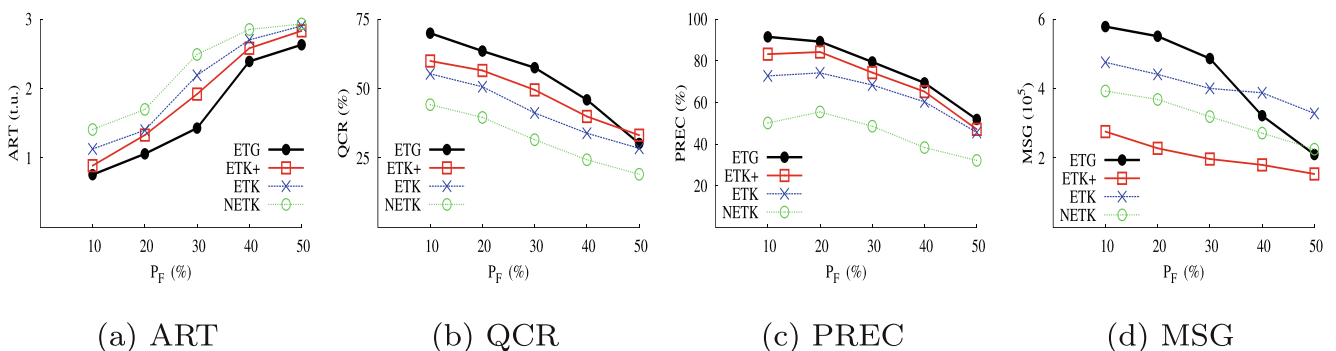
As the percentage  $P_F$  of MP failures increases, MP participation decreases, query paths become longer and fewer potential rankers remain available, thereby degrading

the performance of all the schemes. Interestingly, at  $P_F = 50\%$ ,  $ETK$ ,  $ETK+$  and  $ETG$  exhibit comparable performance due to limited MP participation making the effect of groups and item re-ranking less pronounced. As the results in Fig. 10d indicate,  $MSG$  decreases with increase in  $P_F$  for all the schemes due to reduced communication overhead among a lower number of available MPs. Interestingly, detailed examination of the experimental logs indicated that beyond  $P_F = 35\%$ ,  $ETG$  incurs lower  $MSG$  than  $ETK$  due to difficulties in group formation when relatively fewer MPs are available.

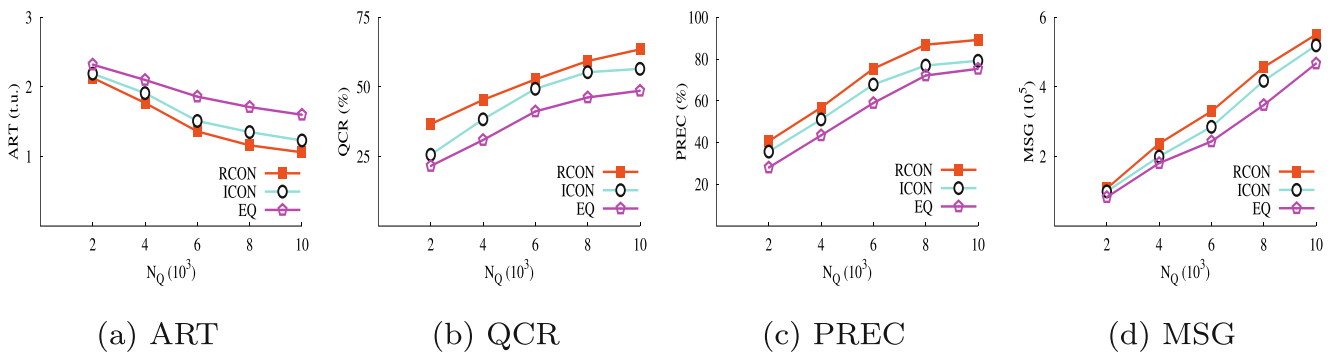
### 6.8 Effect of different payoff allocation approaches in $ETG$

We conducted an experiment to investigate the relative performance of  $ETG$  with the different payoff allocation approaches, namely  $EQ$ ,  $ICON$  and  $RCON$ . Figure 11 depicts the results.

As the number of queries increases, performance improves for all the approaches partly due to better filtering by the group leader and partly because participants lower the item selection probability for penalized items. (Recall that in this work, participants in  $ETG$  decrease the item selection probability in the same manner as in



**Fig. 10** Effect of MP failures



**Fig. 11** Effect of different payoff allocation approaches in ETG

the ETK+ scheme.) In effect, the filtering occurs iteratively to refine the top- $k$  results across an increased number of queries, thereby improving both QCR and PREC.

ICON outperforms EQ because it provides better incentives to contributors. Unlike EQ, it takes into account the number of relevant items sent by contributors for distributing rewards. Moreover, RCON outperforms ICON since it better incentivizes contributors by tying rewards to the revenue earned by the items. QCR and PREC saturate for all the approaches after 8000 queries have been processed due to network partitioning and unavailability of some of the rankers. RCON incurs more MSG than ICON, and ICON incurs more MSG than EQ because better incentives entail more participation towards top- $k$  query processing.

### 6.9 Effect of variations in the group size

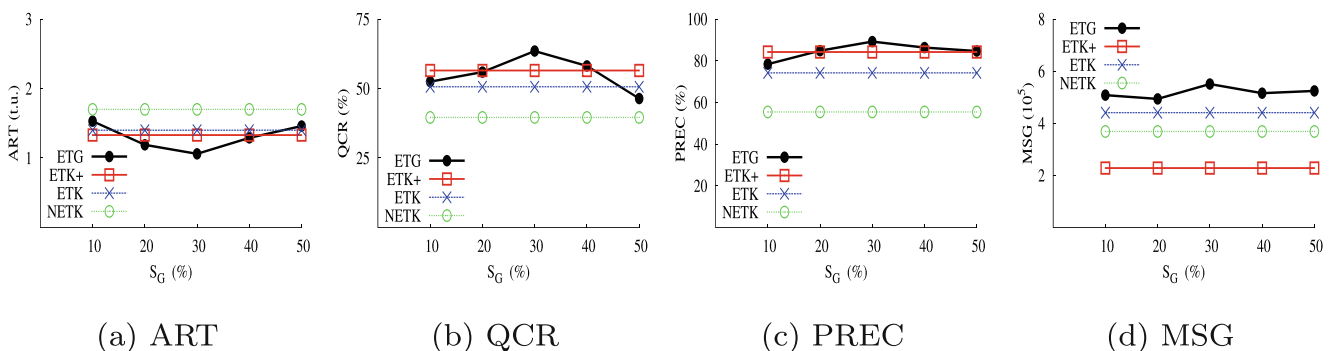
We define the size of a group as the number of MPs in it. We conducted an experiment to investigate the effect of variations in the group size. For simplicity, we divide the region of interest into square cells of equal area in a grid. MPs moving within any given cell constitute a group, hence each cell corresponds to a group. Thus, we vary the group size by adjusting the area of the cells. Hence, if we increase the area of each cell, the group size increases and vice versa.

Notably, although the cells are of equal area, group size may vary across cells because MPs are not uniformly distributed across the region.

We define a parameter  $S_G$  to quantify the side-length of an individual cell as a percentage of the total side-length of the region. Recall that our region is 1000 metres  $\times$  1000 metres. Hence, when  $S_G = 10\%$ , each cell has an area of 100 metres  $\times$  100 metres, hence there will be 100 cells i.e., groups. Similarly, when  $S_G = 30\%$ , each cell has an area of 300 metres  $\times$  300 metres, hence there will be  $11.11 \approx 12$  cells. (Notably, the group corresponding to the last cell is likely to have fewer MPs than the first eleven cells.) Observe how the number of cells (and consequently, groups) decreases drastically with increase in  $S_G$ .

Figure 12 depicts the effect of variations in  $S_G$ . At low values of  $S_G$ , the number of groups is high, but each group is relatively small in size due to fragmentation. Hence, it becomes more difficult for the group leaders to obtain  $k$  relevant items for sending to the broker. Thus, group members behave more like individual rankers, thereby not fully realizing the benefits of group-based collaboration. Hence, ETG exhibits improved performance as  $S_G$  is increased from 10% to 30% due to increase in group size.

On the other hand, at high values of  $S_G$ , group size becomes large, hence relatively fewer groups exist. However, when the group size is too large, performance



**Fig. 12** Effect of variations in the group size

degrades. In the extreme case when the group encompasses the whole region, the performance essentially reduces to that of ETK and ETK+ because all the MPs act as part of one group. This explains why the performance of ETG degrades and becomes comparable to that of ETK and ETK+, at  $S_G = 40\%$  and beyond. Observe that ETG performs best at  $S_G = 30\%$  when the group sizes are neither too small nor too large. Interestingly, MSG is highest for ETG at  $S_G = 30\%$  due to more group interactions in processing a larger number of successful queries.

## 7 Conclusion

We have proposed the E-Top system for efficiently processing top- $k$  queries in M-P2P networks. E-Top issues economic rewards to the MPs, which send relevant data items, and penalizes peers otherwise, thereby optimizing the communication traffic. Peers use the payoffs as a means of feedback to re-evaluate the scores of their items for re-ranking purposes.

E-Top uses three economic incentive schemes, namely ETK, ETK+ and ETG. In ETK and ETK+, MPs act individually, the difference being that ETK performs equal distribution of payoffs to the rankers, while ETK+ uses a weighted distribution. ETG extends ETK and ETK+ by considering MP collaboration in groups. Our performance evaluation demonstrates that E-Top is indeed effective in improving the performance of top- $k$  queries in terms of query response times and accuracy at reasonable communication traffic cost.

Recall that in this work, we have considered only one broker towards serving a given top- $k$  query in a given query path. In the near future, we plan to further enhance system scalability by requiring multiple brokers per query for providing better services to the mobile peers, albeit at the cost of increased communication overhead in terms of energy and bandwidth. There is no restriction over considering multiple brokers, but in that case the inter-broker communication further increases communication traffic, thereby degrading the overall performance of the system. This is due to the fact that the nearby brokers periodically exchange the information (such as global ranking list ( $T_G$ ), number of unique MPs that interacted with brokers, etc.) with each other, to maintain consistency in such a dynamic environment. Additionally, we will create a small prototype to provide a proof of concept.

## References

1. Mechanical Turk (Amazon.com). <http://www.mturk.com/mturk/welcome>
2. OMNeT++ (omnet.org). <http://www.omnetpp.org>
3. Broch J, Maltz D, Johnson D, Hu Y, Jetcheva J (1998) A performance comparison of multi-hop wireless ad hoc network routing protocol. Proc. MobiCom
4. Buttyan L, Hubaux J (2001) Nuglets: a virtual currency to stimulate cooperation in self-organized mobile Ad Hoc networks. Technical report DSC/2001/001, swiss federal institute of technology, Lausanne
5. Buttyan L, Hubaux J (2003) Stimulating cooperation in self-organizing mobile Ad Hoc networks. ACM Mob Netw Appl 8(5)
6. Chakravorty R, Agarwal S, Banerjee S, Pratt I (2005) MoB: a mobile bazaar for wide-area wireless services. Proc. MobiCom
7. Chen K, Nahrstedt K (2004) iPass: an incentive compatible auction scheme to enable packet forwarding service in MANET. Proc. ICDCS
8. Crowcroft J, Gibbens R, Kelly F, Ostring S (2003) Modelling incentives for collaboration in mobile Ad Hoc networks. Proc. WiOpt
9. Datta A, Hauswirth M, Aberer K (2003) Beyond web of trust: Enabling P2P E-commerce. Proc. ICEC
10. Dong M, Kimata T, Sugiura K, Zettsu K (2014) Quality-of-experience (QoE) in emerging mobile social networks. IEICE Trans 97-D(10)
11. Ferguson D, Nikolaou C, Yemini Y (1993) An economy for managing replicated data in autonomous decentralized systems. In: Proc. International Symposium in Autonomous Decentralized Systems
12. Ferguson D, Yemini Y, Nikolaou C (1988) Microeconomic algorithms for load balancing in distributed computer systems. Proc. ICDCS
13. Franklin M, Kossmann D, Kraska T, Ramesh S, Xin R (2011) CrowdDB: Answering Queries with Crowdsourcing. Proc. SIGMOD
14. Golle P, Brown K, Mironov I (2001) Incentives for sharing in peer-to-peer networks. Proc. Electronic Commerce
15. Grothoff C (2003) An excess-based economic model for resource allocation in peer-to-peer networks. Proc. Wirtschaftsinformatik
16. Gutierrez-Nolasco S, Venkatasubramanian N, Talcott C, Stehr MO (2011) Tailoring group membership consistency for mobile networks. Proc. CTS
17. Guy R, Reiher P, Ratner D, Gunter M, Ma W, Popek G (1998) Rumor: mobile data access through optimistic peer-to-peer replication. Proc. ER Workshops
18. Hagihara R, Shinohara M, Hara T, Nishio S (2009) A message processing method for top- $k$  query for traffic reduction in Ad Hoc networks. Proc. MDM
19. Ham M, Agha G (2005) ARA: a robust audit to prevent free-riding in P2P networks. Proc. P2P
20. Hara T, Hagihara R, Sasaki Y, Shinohara M, Nishio S (2010) Efficient top- $k$  query processing in mobile ad hoc networks. Next-Generation Comput. 1(2)
21. Hara T, Madria S (2006) Data replication for improving data accessibility in Ad Hoc networks. IEEE TMC 5(11)
22. Hwang S, Chang K (2007) Optimizing top- $k$  queries for middleware access: a unified cost-based approach. ACM TODS 32(1)
23. Jung H, Cho B, Chung Y, Liu L (2010) On processing location based top- $k$  queries in the wireless broadcasting system. Proc. ACM SAC
24. Kamvar S, Schlosser M, Garcia-Molina H (2003) Incentives for combatting free-riding on P2P networks. Proc. Euro-Par
25. Kremer S, Markowitch O, Zhou J (2002) An intensive survey of non-repudiation protocols. Technical Report 473, ULB
26. Kumrai T, Ota K, Dong M, Champrasert P (2014) RSU placement optimization in vehicular participatory sensing networks. Proc. INFOCOM workshops

27. Kurose JF, Simha R (1989) A microeconomic approach to optimal resource allocation in distributed computer systems, vol 38
28. Lam K, Cheng R, Liang B, Chau J (2004) Sensor node selection for execution of continuous probabilistic queries in wireless sensor networks. Proc. ACM VSSN Workshop
29. Lee W, Lee J, Kim Y, Leung C (2009) AnchorWoman: Top-k structured mobile web search engine. Proc. CIKM
30. Liebau N, Darlagiannis V, Heckmann O, Steinmetz R (2005) Asymmetric incentives in peer-to-peer systems. Proc. AMCIS
31. Liu J, Issarny V (2004) Service allocation in selfish mobile Ad Hoc networks using Vickrey auction. Proc. Current trends in database Technology - EDBT Workshops
32. Liu X, Xu J, Lee W (2010) A cross pruning framework for top-k data collection in wireless sensor networks. Proc. MDM
33. Ota K, Dong M, Chang S, Zhu H (2015) MMCD: cooperative downloading for highway VANETs, vol 3, IEEE Trans Emerg Top Comput
34. Ramaswamy L, Liu L (2003) Free riding: A new challenge to P2P file sharing systems. Proc. HICSS
35. Ratner D, Reiher P, Popek G, Kuenning G (2001) Replication requirements in mobile environments, vol 6
36. Richard B, Nioclaïs D, Chalou D (2003) Clique: a transparent, peer-to-peer replicated file system. Proc. MDM
37. Sabater J, Sierra C (2005) Review on computational trust and reputation models. Artif Intell Rev 24(1)
38. Soliman M, Ilyas I, Chang K (2008) Probabilistic top-k and ranking-aggregate queries. ACM TODS 33(3)
39. Srinivasan V, Nugehalli P, Chiasserini C, Rao RR (2003) Cooperation in wireless Ad Hoc networks. Proc. INFOCOM
40. Wolfson O, Xu B, Sistla A (2004) An economic model for resource exchange in mobile peer-to-peer networks. Proc. SSDBM
41. Wu M, Xu J, Tang X, Lee W (2006) Monitoring top-k query in wireless sensor networks. Proc. ICDE
42. Xu B, Wolfson O, Rish N (2006) Benefit and pricing of spatio-temporal information in mobile peer-to-peer networks. Proc. HICSS
43. Xue Y, Li B, Nahrstedt K (2005) Channel-relay price pair: towards arbitrating incentives in wireless Ad Hoc networks. Wireless communications and mobile computing
44. Xue Y, Li B (2005) Optimal resource allocation in wireless Ad Hoc networks: A price-based approach. IEEE TMC
45. Zeinalipour-Yazti D, Vagena Z (2008) Distributed Top-K Query processing in wireless sensor networks. Proc. MDM
46. Zhong S, Chen J, Yang Y (2003) Sprite: A simple: Cheat-proof, credit-based system for mobile Ad-Hoc networks. Proc. INFOCOM



**Nilesh Padhariya** is a Ph. D. scholar at the Indraprastha Institute of Information Technology, Delhi (IIIT-D), INDIA, where he works on mobile data management and economy-based incentive schemes for improving peer participation in mobile-P2P environments. He earned his M.Tech. degree in Computer Applications from Indian Institute of Technology, Delhi (IITD) in 2006. He has also received several research grants.



**Anirban Mondal** is a senior researcher in Xerox research. Earlier, he was an Associate Professor at the Indraprastha Institute of Information Technology, Delhi (IIIT-D), INDIA. Prior to this, he had been working for the University of Tokyo for a long tenure of seven years. He received his Ph.D. in Computer Science from the School of Computing, National University of Singapore (NUS) in 2003. He had completed his Bachelor of Technology (with

Honours) in Computer Science and Engineering at the Indian Institute of Technology (IIT), Kharagpur, INDIA in 1998.

His research interests include Peer-to-Peer environments, mobile environments, mobile-P2P networks, GRID computing, distributed data management, spatial data and Geographic Information Systems, indexing and load-balancing for multidimensional databases, DataBanking and autonomic computing. He had been awarded the prestigious JSPS (Japanese Society for Promotion of Science) Fellowship from September 2003-September 2005. He has published several papers and given invited talks at several key international database conferences/workshops.



**Sanjay Kumar Madria** received his Ph.D. in Computer Science from Indian Institute of Technology, Delhi, India in 1995. He is a Professor, Department of Computer Science, at the Missouri University of Science and Technology (formerly, University of Missouri-Rolla), USA and Site Director, NSF Industry/University Center on Net-centric System Software. Earlier he was Visiting Assistant Professor in the Department of Computer Science,

Purdue University, West Lafayette, USA. He co-authored a book entitled “Web Data Management: A Warehouse Approach” published by Springer-verlag. He guest edited WWW Journal. His research interests are in cloud computing, mobile p2p, security and data management. He served as a general co-chair of IEEE Mobile Data Management conference and IEEE Symposium on Reliability in Distributed Systems. He received faculty excellence award in 2007, 2009, 2011 and 2013, Japanese Society for Promotion of Science invitational fellowship in 2006, and Air Force Research Lab’s visiting faculty fellowship from 2008 to 2015. He is IEEE Senior Member, IEEE Golder Core Member and an ACM Distinguished Visitor program.