Co-Utile Ridesharing

David Sánchez, Sergio Martínez, and Josep Domingo-Ferrer

Abstract Ridesharing has the potential to bring a wealth of benefits both to the actors directly involved in the shared trip (e.g., shared travel costs or access to high-occupancy vehicle facilities) and also to the society in general (e.g., reduced traffic congestion and CO2 emissions). However, even though ridesharing is based on a win-win collaboration and modern mobile communication technologies have significantly eased discovering and managing ride matches, the adoption of ridesharing has paradoxically decreased during the last years. In this respect, recent studies have highlighted how privacy concerns and the lack of trust among peers are crucial issues that hamper the success of ridesharing. In this chapter, we tackle both of these issues by means of i) a fully decentralized P2P ridesharing management network that avoids centralized ride-matching agencies (and hence private data compilation by such agencies); and ii) an also decentralized reputation management protocol that brings trust among peers, even when they have not previously interacted. Our proposal rests on co-utility, which ensures that rational (even purely selfish) peers will find no incentives to deviate from the prescribed protocols. We have tested our system by using data gathered from real mobility traces of cabs in the San Francisco Bay area, and according to several metrics that quantify the degree of adoption of ridesharing and the ensuing individual and societal benefits.

David Sánchez
Universitat Rovira i Virgili, Dept. of Computer Science and Mathematics, UNESCO Chair in Data Privacy, Av. Països Catalans 26, E-43007 Tarragona, Catalonia, e-mail: david.sanchez@urv.cat

Sergio Martínez
Universitat Rovira i Virgili, Dept. of Computer Science and Mathematics, UNESCO Chair in Data Privacy, Av. Països Catalans 26, E-43007 Tarragona, Catalonia, e-mail: sergio.martinezl@urv.cat

Josep Domingo-Ferrer
Universitat Rovira i Virgili, Dept. of Computer Science and Mathematics, UNESCO Chair in Data Privacy, Av. Països Catalans 26, E-43007 Tarragona, Catalonia, e-mail: josep.domingo@urv.cat

1 Introduction

Ridesharing is a mode of transportation in which several travelers share a vehicle (typically a private car) for a trip and split the trip costs. In this manner, they can enjoy the convenience and speed of private car rides without paying much more than if using public transportation [14]. Moreover, ridesharing is often seen as a promising means to reduce CO_2 emissions [6] and use finite oil supplies in a wiser way. For end users (drivers and passengers), ridesharing mainly saves travel costs, but it may also reduce travel time in case high-occupancy vehicle (HOV) facilities are available for vehicles carrying two or more people (special lanes, toll booths or parking spaces) and, if widely adopted, by mitigating traffic congestion [7, 16]. In fact, the benefits of congestion mitigation can be very significant and extend far beyond the ridesharers themselves: the annual cost of traffic congestion in the US in terms of wasted time and fuel was estimated at $78 billion in 2007 [23].

Ridesharing naturally defines a win-win scenario whereby the involved parties maximize their outcomes as a result of their collaboration. This mutually beneficial collaboration has been formalized and termed *co-utility* (see [13, 12] and the first chapter of this book). Ridesharing is co-utile because, in terms of cost and time, it is the best option even for purely selfish agents [22].

Ridesharing has been used on a regular basis since the 1970s by means of *carpools*. Carpooling is usually understood as an organized and regular ridesharing service (e.g., employees taking turns to drive each other to work). With the advent of the Internet and the enormous adoption of mobile communication technologies, modern ridesharing is also characterized by a more dynamic (or even real-time) scheduling of rides [7]. By dynamic ridesharing we refer to an automated system that matches drivers and travelers on very short notice or even en route [1]. Current ridesharing platforms are Internet-based agencies (e.g., Carma, BlaBlaCar, CarpoolWorld, etc.) that match drivers' availabilities with passengers' needs in a centralized way [14]; in this scenario, the main challenges are to find optimal matches and immediately satisfy on-demand requests to form ridesharing instantaneously [1].

Recent studies [19], however, report a significant and continuous decline of ridesharing (i.e., ridesharing accounted for 19.7% of work trips in 1980 whereas it fell down to 10% in 2009). Why the immediacy and ubiquity of Internet-enabled mobile communications have not shifted people's choices of transportation has been recently studied in [1, 14]. Furuhata et al. [14] identify several difficulties that prevent ridesharing from being widely adopted. Building trust among travelers that do not know each other stands out as a crucial issue. It is well-known that most people are reluctant to travel with complete strangers (e.g., according to the survey in [8], only 7% of respondents would accept rides from strangers, whereas 98% and 69% would accept rides from a friend and the friend of a friend, respectively). The user's experience is usually the most reliable foundation of trust, and it can be spread through the network with a reputation system [11] by which users (both drivers and passengers) provide feedback on each other. Even though some matching agencies (such as Carma or Carpool World) already implement reputation systems, there is another key issue that severely hampers the ridesharing model offered by these

agencies: users' privacy [14]. Indeed, a significant loss of privacy occurs as a result of matching agencies systematically collecting travel data and reputations of their customers [2], an information that may be sold or used for other purposes, such as marketing products or personal profiling [26].

To neutralize the above trust-related disincentives to ridesharing without running into privacy-related disincentives, we propose a fully decentralized P2P ridesharing system with the following features [22]:

- Being fully decentralized, it circumvents central matching agencies that may compile, aggregate and exploit individuals' data (location, travel habits, reputations) or that may bias ride matches because of commercial interests. Moreover, decentralization eliminates a central point of failure that might be the target of external attacks.
- For the sake of user privacy, we maintain information disclosure at a minimum, so that only the driver and the passenger(s) whose trips match learn each other's identity, desired trip and reputation.
- To tackle the reluctance of users to share trips with strangers, we implement a fully distributed reputation management mechanism whereby i) drivers and passengers can rate each other after a shared ride, and ii) peers can learn the reputation of others (resulting from aggregation of ratings of past rides) in a trustworthy way before deciding whether to share or not a future trip. Even though decentralized reputation may look more complex and less reliable than centralized reputation maintained by a trusted party, our reputation mechanism is designed in such a way that rational agents are interested in honestly cooperating to maintain it [11]. As a result, it is robust against a number of tampering attacks (e.g., fake reputation reporting, creation of multiple accounts, etc.), because agents cannot derive any benefit from such attacks.
- Our system rests on the theoretical foundations of co-utility [13, 12] to characterize users' collaboration and ensure that rational users (even purely selfish ones) will follow the proposed protocols.

The technical emphasis of our work is thus on the decentralized, privacy-preserving and efficient management and matching of drivers' offers and passengers' requests, as well as on the distributed calculation of peers' reputations. By designing the protocols in our system to be co-utile, we make sure the system will work as intended even if there is no central entity controlling its operation.

2 Decentralized P2P ridesharing

Our system follows the *organized ridesharing* paradigm, by which ride matching can be prearranged between participants (unlike hailing a cab or hitchhiking, which are *ad hoc*). However, it does not necessarily assume previous involvements between the participants [9] (i.e., passengers and drivers do not need to know each other).

Organized ridesharing has been operated by agencies that match passengers and drivers according to a specific matching algorithm [14]. Ridesharing matches are two-sided, since they are performed according to the offers and requests received from drivers and passengers, respectively. Our system provides the same type of ridesharing matching service, but without relying on a central matching agency. Specifically, we offer a fully decentralized P2P network in which passengers and drivers act as peers that dynamically enter the network to offer or request rides. Ride matches are managed by the peers themselves in such a way that only those peers whose ride offers/requests match disclose their personal details (i.e., specific locations and identities) to each other. This feature overcomes the privacy concerns that have hampered ridesharing services thus far [14], because users are no longer forced to disclose all their ride details to a centralized agency.

2.1 Ridesharing types

Ridesharing matching depends on the spatial and temporal features of ride offers and requests. In our system, each driver D has a trip route $R(D) = \{(l,t)_0^D, \ldots, (l,t)_n^D\}$ that is defined by an origin location and time $(l,t)_0^D$, a destination location and time $(l,t)_n^D$ and, optionally, an ordered set of the intermediate positions and associated (estimated) times $(l,t)_i^D$. Even if intermediate positions are optional, the more detailed the route (i.e., the more numerous and precise the intermediate points), the higher the chance to successfully match drivers' routes and passengers' requests, as it will be discussed below. Each passenger P, on the other hand, states his ride request by just specifying an origin location and desired pick-up time, and a destination, that is, $R(P) = \{(l,t)_0^P, (l)_1^P\}$. We assume passengers have no specific route constraints between their origins and destinations (the specific path followed to reach destination does not matter to them).

Similar to the usual practice by matching agencies, we do not require ride offers and requests to perfectly match (w.r.t. locations and times), because this would be too rigid and would severally restrict feasible matches. Instead, each passenger P defines spatial and temporal slacks that specify the maximum distance δ^P he accepts to walk (from his origin location l_0^P to the pick-up location by a driver, say l_i^D, and from the driver's drop-off location, say l_j^D, to his destination l_1^P), and the maximum time τ^P he accepts to wait before being picked up (with respect to his initial preference t_0^P). Our system supports the usual types of ridesharing implemented by state-of-the-art matching agencies [14, 16]:

- *Identical ridesharing*: the origins and destinations of driver and passenger match both spatially and temporally (within the passenger's space and time slacks); that is, $|l_0^D - l_0^P| \leq \delta^P \wedge |l_n^D - l_1^P| \leq \delta^P \wedge |t_0^D - t_0^P| \leq \tau^P$.
- *Inclusive ridesharing*: the origin and destination of the passenger are included in the route $R(D)$ of the driver (within the passenger's space and time slacks); that is, $\exists (l,t)_i^D \in R(D) \mid |l_i^D - l_0^P| \leq \delta^P \wedge |t_i^D - t_0^P| \leq \tau^P \wedge \exists l_j^D \in R(D) \mid |l_j^D - l_1^P| \leq \delta^P$.

With this ridesharing type, the driver does not need to deviate from her route, but she must stop at the pick-up and drop-off locations agreed with the passenger (l_i^D and l_j^D, respectively). As stated above, the more numerous and precise the intermediate positions/times of the driver's route, the higher the chance of finding an inclusive match with a passenger.

The above ridesharing types also support multiple passengers per vehicle (up to the maximum capacity c^D specified by each driver D for her vehicle). In inclusive ridesharing, the only difference between carrying one or several passengers is that in the latter case the driver must stop several times for pick-ups and drop-offs.

The literature also considers more complex ridesharing types. For example, in *partial ridesharing* the shared ride covers only a part of the passengers's trip, so that additional shared rides or transportation means need to be concatenated to complete the trip; in *detour ridesharing*, the driver may deviate from her route to accommodate the passengers' requests, at the expense of increased travel costs. Because these complex ridesharing types hinder instantaneous decision making by drivers and passengers and/or may require adding compensations for detouring, they are not currently offered by matching agencies [14] and we leave them as future work.

2.2 The P2P ride management network

Rather than using a central matching agency, our system consists of a fully decentralized P2P network of drivers and passengers. Since decentralized P2P networks lack a trusted authority and a common "legal" framework, our challenge here is to design a co-utile protocol for ridesharing (which in turn will rely on a co-utile protocol for reputation management), that is, a ridesharing protocol that is followed by rational peers because of the benefits it brings to them.

There are several alternative communications technologies to implement decentralized P2P applications (e.g., CAN [10], Chord [24], Pastry [5], etc.). Among these, Pastry [5] is especially interesting because of its proximity-based routing. It consists of a self-organizing overlay network that provides a fault-tolerant and load-balanced distributed hash table (DHT) for large-scale P2P applications. Each node (i.e., a P2P user) is identified by a unique, randomly generated *nodeId* from a 128-bit identifier space, which acts as a pseudonym that hides the real identity (IP address) of the user. The DHT is internally used to map an existing *nodeId* to its associated IP address. In this manner, a message can be anonymously routed to any network user (via anycast) by forwarding it a bounded number of hops, concretely, $< \log_{2^b} N$, where N is the number of nodes and b typically has the value of 4. Moreover, each node requires locally storing only $O(\log_{2^b} N)$ entries of the hash table. Proximity-based message routing is performed by sending each message to the numerically closest *nodeId* that the sender has in its local hash table. If the receiver is not the message addressee, it forwards the message to its closest *nodeId*, and the process is repeated until the message reaches its intended addressee.

2.3 Distributed ride-matching protocol

To match their rides, drivers and passengers need to advertise their offers and requests, thereby incurring some information disclosure. As discussed above, privacy concerns about that disclosure may deter users from participating in the ride-matching protocol [14]. Thus, to ensure that the protocol remains co-utile (and thus self-enforcing for rational agents), we need to keep the disclosure of the peers' spatio-temporal features as low as possible.

Within a decentralized network, a straightforward way to look for ride matches is to broadcast to all active peers a message with the passengers' requests and drivers' offers. However, this incurs a large information disclosure, and it may be even worse for privacy than using a centralized matching agency. Moreover, broadcast is very inefficient because it unnecessarily floods the network with messages that are useless to most peers. Instead, we propose a mechanism predicated on a topic-based decentralized subscription system implemented by the network by which i) users only communicate with those peers with *similar* travel needs/offers and ii) their concrete spatio-temporal data are only disclosed if a potential match is found. The idea is that drivers who offer rides *subscribe* to *topics* according to the routes they offer (topics are locally stored by the peers), and passengers *publish* their ride needs to the appropriate topic in the subscription system. Being decentralized, an event notification system for topic-based subscribe-publish applications is in charge of communicating via multicast only those peers whose rides match (i.e., drivers subscribed to a topic describing a route for which a passenger has published a message). Since the subscription information is local for each peer, topic subscriptions by drivers can only be known by publishing a message in a matching topic. Thanks to the proximity-based message routing and the topic-oriented multicast, the notification system is capable of scaling to a potentially large number of subscribers per topic and to a large number of different topics. Technically, the subscription infrastructure is built upon the decentralized Pastry P2P network to manage topic creation, topic subscription and dissemination of messages published in each topic.

Topics can be initially created (and subscribed to) by any peer (drivers, in our case) by defining a *topicId*. In our system, *topicId*s are defined according to the detail of the routes being offered by drivers, that is, all the possible rides consisting of a pick-up location/time and a drop-off location that can be defined from each route (that is, all subroutes of each route). Drivers offering rides equal to an existing *topicId* can register to the same topic to become subscribers of it. On the other hand, only peers knowing a *topicId* can publish messages in a certain topic. In our case, passengers publish messages in *topicIds* that match their ride needs (desired pick-up location/time and drop-off location). Thus, a passenger's message is disseminated only to the drivers subscribed to the matching topic, that is, to a ride offer that matches the passenger's needs; only in this case, peers may exchange their personal details (e.g., identities).

While the subscribe-publish mechanism routes the messages only to the peers whose offers/requests match and the anonymity of peers is preserved unless they identify themselves to engage in a shared ride, topic subscription and publishing

via detailed routes still discloses the spatio-temporal features of the users in the *topicIds*. Moreover, the number of different *topicIds* that can be defined according to *exact* locations and times may be unmanageably large. Also, too much detail may be counterproductive for the flexible matching described in Section 2.1: passengers ought to publish as many topics as locations and times included in their space and time slacks.

We tackle these issues by defining topics as *generalizations* of spatio-temporal features, rather than exact values. Specifically, exact locations l_i (both of passengers and riders) are generalized to zones within the area A (e.g. a city) where the ridesharing system will be deployed. Zones are fixed by a predefined partition $Z(A)$ of the area (e.g., districts or neighborhoods in a city, uniform partitions in a map, etc.). Figure 1 shows an example of such partition for the city of San Francisco, where each $Z_i \in Z(A)$ corresponds to the name of a neighborhood. Likewise, concrete times t_i associated to positions l_i are generalized to fixed time intervals I (e.g., 10 minutes, a quarter of an hour, etc.). Specifically, since we measure t_i as the Epoch time (i.e., seconds since January 1st, 1970), the corresponding interval I is calculated as the integer part of the quotient between t_i and the size of the time interval (e.g., $\lfloor t_i/900 \rfloor$ for 15 minutes). Once generalized, each *topicId* is defined by the string resulting from concatenating the pick-up zone Z_o and time interval I_o with the drop-off zone Z_e. The system designer may configure beforehand the granularity of the generalization of both dimensions in order to control the level of incurred disclosure/matching accuracy. To optimize privacy, zones and time intervals should be taken to be as large as allowed by the spatio-temporal flexibility of passengers.

Formally, the ride-matching protocol (depicted in Figure 2) is as follows:

1. Each driver D advertises the route $R(D)$ she would like to share by subscribing to as many topics as possible rides can be defined in her route. To do so, first, D generalizes positions and times $(l,t)_i^D \in R(D)$ to zones and time intervals $(Z,I)_i^D$, respectively; we refer to the *generalized route* as $\bar{R}(D) = \{(Z,I)_0^D, \ldots, (Z,I)_n^D\}$. Then, D subscribes to all topics $S(D)$ whose *topicId*s are triples (Z_o, I_o, Z_e), where $(Z_o, I_o) \in \bar{R}(D)$ is a generalized pick-up location-time pair and $Z_e \neq Z_o$ is a generalized drop-off location is such that $(Z_e, I_e) \in \bar{R}(D)$ for some generalized time interval I_e. Note that, by considering all the possible rides within the driver's route, we support the *inclusive ridesharing* described in Section 2.1.
2. Each passenger P looking for a ride (with desired route $R(P)$) publishes his request in a list of topics $S(P)$ that includes *topicID*s obtained in a way analogous to the *topicIds* subscribed by a driver (see above). Specifically, since $R(P) = \{(l,t)_0^P, (l)_1^P\}$, then $S(P) \supseteq \{(Z_o, I_o, Z_e)\}$ where Z_o and Z_e are the zones corresponding to the generalization of l_0^P and l_1^P, respectively, and I_o is the generalized time interval of t_0^P. If the passenger has flexible spatio-temporal requirements, $S(P)$ contains as many topics as zones and time intervals can be obtained by generalizing the locations and times resulting from adding/subtracting the passenger's space slack δ^P to locations in $R(P)$ and the passenger's time slack τ^P to the time in $R(P)$. As a result of the publication on these topics, only those drivers that have subscribed to the same topics will receive the passenger's request.

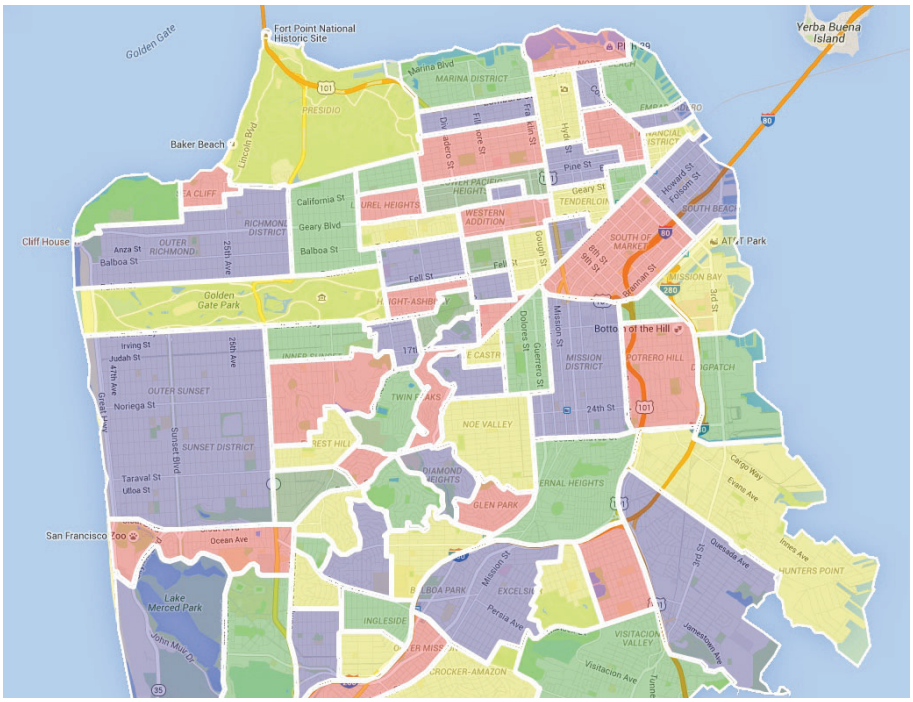


Fig. 1 Example of zone partitioning for the city of San Francisco: zones correspond to neighborhoods.

3. The drivers that receive a passenger's request (say q different drivers) respond with a direct message that discloses the precise pick-up and drop-off locations and pick-up time they generalized into the topic, that is, the specific $(l,t)_i^D$ and l_j^D.
4. Passenger P collects all the answers of the q matching drivers and selects the ride (and the respective driver) that specifically matches his request considering also his spatio-temporal flexibility δ^P and τ^P, as detailed in Section 2.1. If several rides (drivers) match his request, P selects the closest one (in normalized average space and time to his origin and destination) and sends the driver $\hat{D}$ offering that closest ride an "accept" message that reveals P's identity.
5. The driver $\hat{D}$ receiving the "accept" sends an "acknowledgment" to passenger P also revealing $\hat{D}$'s identity.
6. Finally, passenger P sends to the remaining $q-1$ drivers collected at Step 4 a "decline" message. If the best-matching driver $\hat{D}$ does not reply to passenger P for some reason (e.g., it already received an "accept" from another passenger and reached the maximum occupancy c^D of her vehicle), P returns to Step 4 to select the next best alternative.

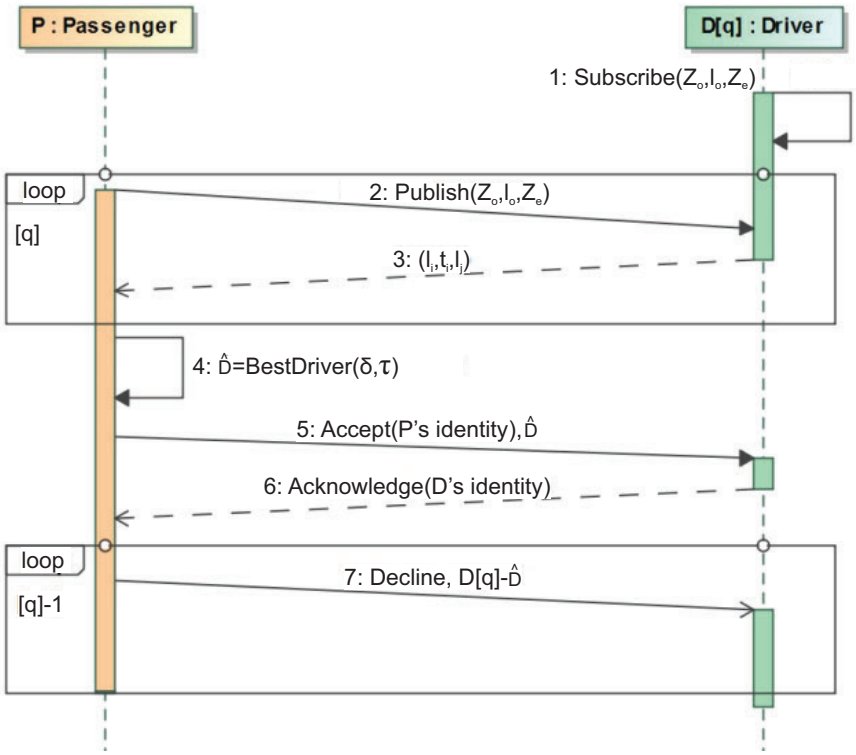


Fig. 2 Sequence diagram of the ride-matching protocol based on a subscribe-publish mechanism

To illustrate the above-described protocol, assume a driver D wants to share the route $R(D)$ across the north of San Francisco shown in Figure 3. $R(D)$ starts in Richmond, ends in Nob Hill, and consists of seven points listed in Table 1.

Table 1 Points (locations coordinates and Epoch times) of the example route $R(D)$

$R(D)$	Location coordinates (l)	Epoch time (t)
$(l,t)^D_0$	37.783094, -122.462326	1455534795
$(l,t)^D_1$	37.784009, -122.457863	1455534882
$(l,t)^D_2$	37.784891, -122.448550	1455535206
$(l,t)^D_3$	37.786892, -122.434302	1455535487
$(l,t)^D_4$	37.788419, -122.422887	1455535851
$(l,t)^D_5$	37.790759, -122.419025	1455536448
$(l,t)^D_6$	37.791166, -122.417394	1455536552

Route $R(D)$ is first generalized to $\bar{R}(D)$ (as shown in Table 2) according to the zone partition $Z(A)$ of San Francisco shown in Figure 1 and time intervals of 10 minutes (i.e., $\lfloor t_i/600 \rfloor$).



Fig. 3 Example driver D's route, $R(D)$, with 7 points, and (flexible) origin and destination for a passenger P

Table 2 Generalized route $\bar{R}(D)$ using San Francisco neighborhoods as zones and time intervals of 10 minutes

$\bar{R}(D)$	Location zones (Z)	Time intervals (I)
$(Z,I)_0^D$	Richmond	2425891
$(Z,I)_1^D$	Lauren Heights	2425891
$(Z,I)_2^D$	Lauren Heights	2425892
$(Z,I)_3^D$	Pacific Heights	2425892
$(Z,I)_4^D$	Pacific Heights	2425893
$(Z,I)_5^D$	Nob Hill	2425894
$(Z,I)_6^D$	Nob Hill	2425894

To advertise the route, driver D subscribes to as many topics $S(D)$ as unique ordered combinations of pick-up/drop-off points can be derived from his generalized route $\bar{R}(D)$. Table 3 lists the triples (Z_o, I_o, Z_e) that constitute the *topicId*s D subscribes to.

Table 3 Set $S(D)$ of *topicId*s derived from the generalized route $\bar{R}(D)$ that driver D subscribes to

Z_o	I_o	Z_e
Richmond	2425891	Lauren Heights
Richmond	2425891	Pacific Heights
Richmond	2425891	Nob Hill
Lauren Heights	2425891	Lauren Heights
Lauren Heights	2425891	Pacific Heights
Lauren Heights	2425891	Nob Hill
Lauren Heights	2425892	Pacific Heights
Lauren Heights	2425892	Nob Hill
Pacific Heights	2425892	Pacific Heights
Pacific Heights	2425892	Nob Hill
Pacific Heights	2425893	Nob Hill
Nob Hill	2425894	Nob Hill

Finally, let us consider that passenger P is looking for a ride with the following features

$$R(P) = \{(37.785146; -122.450288), 1455535117, (37.787215; -122.423938)\},$$

each one respectively corresponding to the points $(l,t)_0^P$ and $(l)_1^P$ shown in Figure 3. Then, the passenger publishes a message in the *topicId* corresponding to the generalized points of his request, that is,

$$S(P) = (LaurenHeights, 2425891, PacificHeights).$$

There is a match between $S(D)$ and the 5th *topicId* subscribed to by D (Table 3). Therefore, the driver answers with the concrete details of that part of her ride, that is,

$$(l_2^D, t_2^D, l_4^D) = ((37.784891, -122448550), 1455535206, (37.788419, -122.422887)).$$

Since the driver's locations and time are within the passenger's spatio-temporal flexibility range given by δ^P and τ^P, the match is successful.

Note that all the steps of the protocol can be automatically managed without the intervention of the users. In Step 1, once drivers have defined their routes, subscription to the appropriate topics is deterministic and straightforward. Moreover, even if drivers just define the starting location and time and the destination of their rides, the system may also estimate the intermediate points and times of the route using a standard route planning algorithm. In Step 2, multicast messages are also automatically managed by the underlying network. In Step 3, response messages can be automatically created by de-generalizing zones and times and in Steps 4 to 6, the best match can be straightforwardly agreed upon according to the passenger restrictions.

A driver D can also unsubscribe from topics when, for some reason, her route changes or when the maximum capacity c^D of her vehicle is reached. Likewise, subscriptions are automatically removed from the system when the ride's time interval I is reached.

2.4 Co-utility analysis of the ride-matching protocol

When travel time and cost are the only utilities relevant to drivers and passengers, ridesharing always results in mutual benefits provided that travel costs (tolls, gas, etc.) are split among the driver and the passenger(s) in a fair way, that is, so that the split cost is less than the cost of traveling alone for both the driver and the passenger(s).

Additionally, if high occupancy vehicles, carrying two or more passengers, are favored by the administration, then not only the cost may be further reduced for the driver and the passenger(s) (e.g. thanks to reduced highway tolls and/or parking

spaces at reduced fees), but the travel time is likely to be shorter with respect to traveling alone (e.g. thanks to HOV lanes).

Since both drivers and passengers obtain benefits by sharing a ride (assuming their time/location/route constraints match), it is in the best interest of the agents to collaborate (i.e., to follow the ride-matching protocol). Hence, ridesharing is *co-utile*. In fact, because our matching patterns do not consider detours by drivers from their routes, co-utility in ridesharing only depends on the availability of agents with appropriate types, that is, compatible ride offers and requests. Therefore, provided that a large and balanced enough (with regard to ride offers/requests) community of drivers/passengers exists, ridesharing will be adopted by *rational* agents in a self-enforcing way.

As discussed in the introduction, (co-utile) ridesharing also boosts societal welfare in the medium term, because it reduces traffic congestion (thereby shortening travel time) and CO_2 emissions. Even though these societal benefits may further motivate some people to adopt ridesharing, the immediate individual cost and time benefits discussed above are arguably the most powerful incentives to collaboration.

However, at a closer look, travel time and cost are not the only utilities involved in ridesharing. If they were, ridesharing would be far more popular than it is nowadays. As discussed above, privacy concerns related to disclosing personal spatio-temporal information to network peers [8] or to matching agencies [2] may partly or totally offset the time/cost utilities, thereby thwarting co-utility and hence ridesharing. In our system, we minimize by design these "negative" privacy-related utilities because: i) we do not rely on any central agency that systematically collects private data; ii) users remain anonymous behind a randomly assigned *nodeId* and do not disclose their identities until a successful match is found; iii) exact travel data are only exchanged between those peers whose ride offers/requests match (via subscription-driven multicast); and iv) drivers' subscriptions in the P2P network only provide generalized information about their routes that is learned by publishing a request in a matching topic. One may certainly conceive systematic attacks whereby malicious users, behaving as passengers (resp. drivers), publish in (resp. subscribe to) topics with fake offers (resp. requests). However, since this attack should be systematic in order to obtain a significant amount of detailed data (i.e., complete drivers' routes, lists of passengers), peers can easily detect the attack pattern and decline communication with suspect peers.

As discussed in previous studies [14, 11], reluctance of drivers and passengers to share a trip with strangers is an even greater hurdle than privacy to the adoption of ridesharing. A natural way to mitigate users' mistrust when they lack direct experiences with their peers is to use a *reputation system*. Reputation, which captures the opinion of the community on each agent has at least two positive effects [11]:

- Reputation allows agents to build *trust*, which can neutralize the negative utilities related to mistrust (e.g., fear or reluctance in front of strangers). The higher an agent's reputation, the more trusted she is by other agents.
- Reputation makes agents accountable for their behavior: if an agent misbehaves (e.g., during the shared ride or by abusing/attacking the ride-matching system as described above), his reputation worsens and the other agents mistrust him more

and more and are less and less interested to interact with him. In this manner, malicious agents (who may try to subvert the system, even irrationally) may be identified (via a low reputation) and penalized (e.g., through limitation or denial of service).

In the next section, we describe our proposal to incorporate a distributed reputation management protocol into our decentralized P2P network. The protocol we use is carefully designed so that it is itself co-utile, and hence even purely selfish agents are interested in following it. In this way, this protocol can be seamlessly used as a mechanism to enforce co-utility in protocols in which negative utilities would otherwise rule it out [11].

3 Distributed reputation management

In ridesharing, reputation is understood as the mutual feedback by drivers and passengers on each other. Such feedback may consist of the aggregation of several objective and subjective outcomes (e.g., how pleasant the trip was or whether there were any unnecessary delays or overcosts). This reputation model is inherited from commercial platforms (e.g. eBay), that successfully support buyers and sellers in building trust to unknown peers and eliciting honest behaviors [21].

Even though some ridesharing agencies (e.g., Carma, Carpool World, Golco) have incorporated reputation systems, they always manage reputations in a centralized way. Centralized reputation is straightforward to implement, but it adds to the privacy loss caused by centralized ridesharing: agencies not only learn where users go, but also what their reputations are.

To overcome this problem in a way that is coherent with the decentralized nature of our ridesharing management system, we propose a fully decentralized reputation management protocol. Moreover, our protocol should ensure that it is in the best interest of rational agents to follow it and report truthful reputation information (this is in fact the most critical issue in reputation systems [15]). In other words, *the reputation management protocol should be itself co-utile*. This aspect is crucial, if we want to use reputation management as a catalyst to turn ridesharing co-utile by neutralizing negative utilities (i.e., concerns about strangers).

The theoretical notions about reputation management and distributed reputation calculation can be found in [11] and also in the second chapter of this book.

3.1 Incorporating reputations into the ride-matching protocol

In ridesharing interactions, the lack of trust between driver and passenger is mutual and may hamper collaboration (i.e., drivers may be reluctant to share their cars with strangers, and passengers may be concerned with sharing a ride with a driver they do not know). With our reputation calculation mechanism, once global reputation

values have been computed, it is immediate for any agent $\mathscr{P}_i$ (either driver or passenger) to learn the reputation value g_j of $\mathscr{P}_j$ (passenger or driver, respectively) in order to decide whether to ride with $\mathscr{P}_j$ or not. Agent $\mathscr{P}_i$ can query the M score managers of $\mathscr{P}_j$ for the latter's reputation; the resulting M values obtained from the M score managers should be the same, because the inputs of the score managers are the same. However, if some values differ (e.g., if some score managers or agents involved in the calculation have altered the computation for some -non-rational/random- reason), $\mathscr{P}_i$ can take as g_j the most common value among the ones sent by the score managers.

In addition to the global reputation of $\mathscr{P}_j$, agent $\mathscr{P}_i$ may also rely on his direct experiences with $\mathscr{P}_j$, if any, which are reflected in his local normalized value c_{ij}. In some cases, local and global reputation values may not be coherent because the latter are the aggregated version of the former. Thus, for a more robust trust enforcement, it is better for $\mathscr{P}_i$ to consider both local reputations and global reputations and take the lowest value in order to make decisions about collaboration. In this manner, agents will be discouraged from selectively behaving well with some agents while misbehaving with others.

Additionally, individual agents may have different levels of reluctance vs. strangers and, thus, may require different reputation levels to build trust and agree to collaborate with other agents. To model this notion, we may allow each agent $\mathscr{P}_i$ to specify a *minimum reputation* threshold ρ_i that any $\mathscr{P}_j$ should have in order for $\mathscr{P}_i$ to ride with $\mathscr{P}_j$. Since ridesharing is predicated on reciprocal trust, it will only happen if both the driver's reputation g_D and the passenger's reputation g_P are above the other party's reputation threshold (i.e., $(g_D \geq \rho_P) \wedge (g_P \geq \rho_D)$).

In summary, the ride-matching protocol detailed in Section 2.3, can be easily extended to enforce trust among the agents by considering the peers' reputations in Steps 3 and 4. In Step 3, the q drivers receiving a passenger P's request may evaluate P's reputation and, only if it is above their desired respective reputation thresholds, the drivers will answer with their respective ride details. Likewise, in Step 4, passenger P will select the best ride (driver $\hat{D}$) only from those drivers with a reputation above his desired reputation threshold ρ^P. In terms of information disclosure, we can also see that with this reputation-aware protocol a passenger and a driver reveal their identities to each other (in Steps 4 and 5, respectively) *only if they both trust each other*. Finally, as discussed in Section 2.4, peers' reputations can also be used to punish agents that are suspected of abusing/attacking the system. All these enhancements brought by the reputation mechanism make following the ride-matching protocol the only rational choice and, thus, they enforce co-utility even if peers do not know each other.

4 Empirical study

This section reports the results of an empirical study that simulates realistic rides in our P2P network. Data about rides have been extracted from the cab mobility traces

provided by the Exploratorium museum within the cabspotting project[1]. This data set contains the traces of the trips of approximately 500 cabs during May 2008 in San Francisco Bay Area [20]. Each trace is defined as the GPS coordinates and the absolute times of its origin, destination and intermediate positions measured every 10 seconds (on average).

4.1 Configuration of the experiments

In our experiments we used the mobility traces of a whole week to simulate ride offers (by drivers) and requests (by passengers). We considered only those traces in which the cab was occupied by a customer, because these are the ones corresponding to realistic routes with meaningful and precise origins and destinations. In contrast, we omitted cabs wandering in search of customers, because this may result in seemingly random routes. We also omitted very short routes of less than 500 meters (for which sharing a ride is not worth while) and/or those with less than 4 position measurements. The resulting set contains 94,070 traces. Figure 4 (left) shows that the traces are uniformly distributed through the week (with a small dominance of Saturdays), whereas Figure 4 (middle) shows that they are concentrated during working hours ([8h-18h]) and the leisure time ([19h-1h]). Likewise, Figure 4 (right) shows that the length of most rides is 3 km or less.

We simulated three well-differentiated scenarios by randomly assigning a percentage of the traces to drivers (ride offers) and passengers (ride requests), as follows:

- *Balanced scenario*: 50% of the traces (47,035) are assigned to drivers' offers and 50% (47,035) to passengers' requests.
- *Driver-dominated scenario*: 70% (65,849) of the traces are assigned to drivers' offers and 30% (28,221) to passengers' requests. This is a favorable setting for passengers, since they can choose from a wide range of offers.
- *Passenger-dominated scenario*: 30% (28,221) of the traces correspond to drivers' offers, whereas 70% (65,849) correspond to passengers' requests. This favors drivers, since they can choose from more passengers. Since there are less drivers than passengers, ridesharing may involve sharing the same trip with several passengers, which also contributes to reducing travel expenses for each ridesharer.

For each driver D, we used all the measurements of her mobility trace (positions and times) in the data set to define her route $R(D)$. For each passenger P, we used only the initial position and time of his trace to define l_0^P and t_0^P, respectively, and the final position to define l_1^P. We also set different levels of passenger flexibility regarding the pick-up/drop-off locations and times: the maximum distance δ^P that a passenger P will be willing to walk (with respect to the pick-up and drop-off locations) has been set to $\delta^P = \{250, 500, 750, 1000\}$ meters, whereas the maximum

[1] `http://www.exploratorium.edu/id/cab.html`

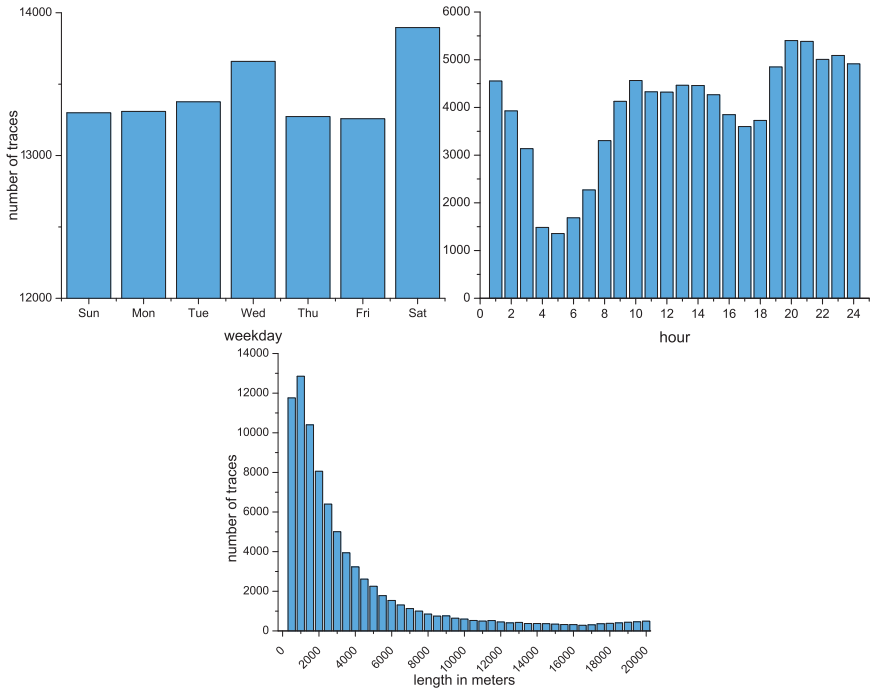


Fig. 4 Distribution of the mobility traces used in our experiments: by days of the week (top left), by hours of the day (top right) and by length (bottom).

waiting time τ^P for pick-up has been set to $\tau^P = \{5, 10, 15, 20\}$ minutes. The maximum values of both dimensions are coherent with recent surveys on carpooling [4], which show that ride partners are usually found within one km radius (roughly 20 minutes walk time) of their current locations.

The management and matching of rides have been implemented as detailed in Section 2.2: drivers define their ride offers in a privacy-preserving way by subscribing to the set of topics defined by the city zones and generalized time intervals corresponding to their routes, and receive requests from potential passengers whose (generalized) ride requirements match. City zones for San Francisco Bay have been taken to be neighborhoods and generalized time intervals have been taken as 10-minute intervals. Because the sub-routes defined by the intermediate positions of the rides are also considered (during the subscription and ride matching), both the *identical* and the *inclusive* ride matching patterns detailed in Section 2.1 have been implemented. Finally, we set a maximum occupancy per vehicle of $c^D = 4$ passengers (plus the driver).

The results of the simulations for the different scenarios have been evaluated according to the following metrics:

- Percentage of passengers that found a matching driver's offer. This metric reflects the functional success of ridesharing for passengers because, if passengers cannot

find drivers, they will be forced to look for an alternative conveyance (e.g., public transportation, cab or their own private vehicle).
- Average occupancy (in number of passengers other than the driver) of the vehicles. This represents the economic success of ridesharing, both for the driver (the higher the occupancy, the more people she can share costs with) and also for the passengers (who will each bear a lower cost). For simplicity, we measured the occupancy as the number of different passengers that share (a part of) the ride through the entire driver's route.
- Percentage of kms saved by sharing rides compared to all agents (both passengers and drivers) traveling separately (for example, with their own private vehicles). This metric measures the boost of social welfare resulting from ridesharing, since saved travel kms imply a reduction of CO_2 emissions.

4.2 Results

Figure 5 shows the results for the three scenarios introduced above according to the three evaluation metrics. Regarding the percentage of passengers that found a matching ride, results are proportional to i) the ratio of passengers/drivers (i.e., the more drivers are available, the higher the chance of finding a match) and ii) the passengers' flexibility regarding time and locations. Flexibility turns out to have the greatest influence on the results, as the percentage of matches increases by 6-9 times between the least flexible and the most flexible requirements (i.e., 15% vs. 87%, 12% vs. 78% and 8% vs. 70% for the driver-dominated, balanced and passenger-dominated scenarios, respectively). Since most ridesharing solutions available in the literature [14] assume that passengers and drivers are matched by proximity rather than by exact position/times, we can see that (flexible) ridesharing has a high potential.

Regarding vehicle occupancy, results are opposite: the more passengers available, the more occupied are vehicles, which results in more saving for everyone. Differences are more significant when comparing the different scenarios: an average 1.623 passengers/vehicle is achieved in the passenger-dominated scenario vs. an average 0.355 in the driver-dominated one. However, even the higher average occupancy is still far from the maximum occupancy of 4 passengers/vehicle. Like matchings, occupancy increases very much with the passengers' flexibility, specifically by a factor 6-9 between the least flexible and the most flexible requirements.

The percentage of kms saved as a result of sharing rides is perfectly proportional to the number of passenger matches (the more matches, the more passengers avoid resorting to their own private cars). For the driver-dominated scenario with the maximum passenger flexibility, as much as 65% kms are saved (130,588 kms in absolute terms). In terms of social welfare and extrapolating the weekly data we considered in our experiments to a year period, saving 65% kms in this scenario represents an annual saving of around 739 thousand gas liters (assuming a consumption of 0.1089

l/km [17]) and a reduction of 1.73 tons of CO_2 emissions (assuming that 255.38 grams of CO_2 are emitted per kilometer [18]).

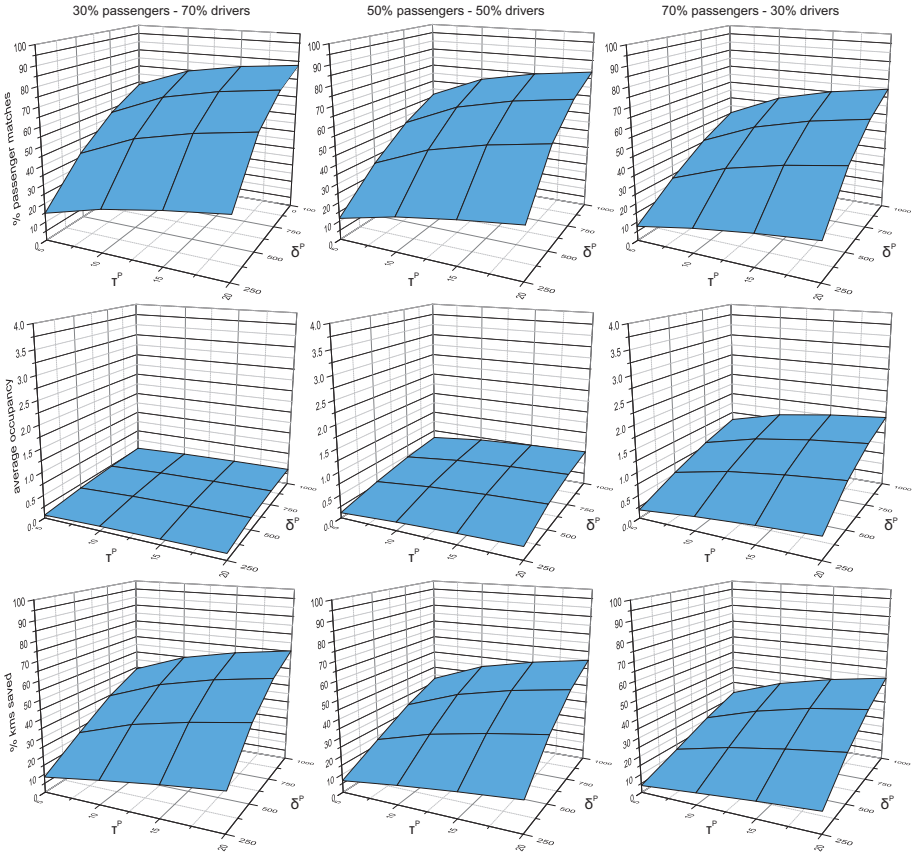


Fig. 5 Percentage of passengers that found a matching ride (top), average occupancy of vehicles (middle) and percentage of kms saved (bottom) as a result of ridesharing for the three scenarios (left: driver-dominated; center: balanced; right: passenger-dominated).

4.3 Lack of trust and reputation management

In the previous simulations, we have assumed that peers do not mind sharing rides with strangers. In this section, we drop this assumption and evaluate the effectiveness of the reputation management mechanism proposed in Section 3 to cope with mistrustful peers.

As a baseline, we focus on the most favorable scenario for passengers, the driver-dominated scenario with maximum passenger flexibility. In the second row of Table 4 we report both the percentage of passengers that found a matching ride (very high: 82.74%) and the average vehicle occupancy in the ideal *Trusted scenario* in which all agents trust each other (or at least do not mind sharing rides with strangers). However, as discussed in [8] only 7% of the respondents seem to be willing to accept rides from strangers. The third row of Table 4 simulates a completely *Untrusted scenario* in which only that 7% of passengers will accept rides with drivers and in which the effectiveness of ride-matching is severely hampered (only 5.79% of matches). Finally, the remaining rows simulate *Untrusted scenarios* in which agents rely only on reputations to decide whether to share or not a matching ride. In all cases, each agent $\mathscr{P}_i$'s global reputation g_i has been randomly chosen in the range $[0,1]$, which corresponds to the normalized reputation range of the calculation mechanism depicted in Section 3. Then, different scenarios have been defined by setting a minimum and equal reputation requirement ρ for all agents in the system, $\rho = \{0.1, 0.2, 0.4, 0.6, 0.8\}$; that is, if we set $\rho \geq 0.4$, by following the reputation-based protocol depicted in Section 3.1, a matching ride would only be accepted if both the driver and the passenger have a global reputation value $g \geq 0.4$. Because of the randomly assigned reputations, for consistency, all the reported results are the average of 5 runs. We can see that in this *Untrusted scenario*, the use of reputations builds trust among the agents and increases the effectiveness of ridesharing in a way that is inversely proportional to the reputation requirements of the agents.

Table 4 Percentage of passengers that found and accepted a matching ride, and vehicle occupancy for different scenarios of trust and reputation requirements

Scenario	*P*'s matches	*D*'s vehicle occupancy
Trusted	82.74%	0.355
Untrusted (no reputations)	5.79%	0.025
Untrusted ($\rho \geq 0.8$)	11.88%	0.051
Untrusted ($\rho \geq 0.6$)	28.28%	0.121
Untrusted ($\rho \geq 0.4$)	46.14%	0.198
Untrusted ($\rho \geq 0.2$)	64.54%	0.277
Untrusted ($\rho \geq 0.1$)	73.25%	0.314

Finally, we simulated a more realistic and heterogeneous scenario in which each agent $\mathscr{P}_i$ is assigned a random global reputation g_i and also a random reputation requirement ρ_i with respect to the other peers, both uniformly distributed within the reputation value range [0..1]. Then, we measured how the global reputation values of peers affected the success of passengers in finding a ride and the success of drivers in finding passengers. Moreover, for passengers we also measured the average distance they needed to walk from their origin to their pick-up location (and also from the drop-off location to their destination) and the time they had to wait for pick-up. For drivers we also measured the average vehicle occupancy for those drivers that were able to find a match. For both, this was measured according to their global

reputation values. Results are shown in Table 5, both for passenger and drivers, for (global) reputation quartiles; again, they are the average of 5 simulation runs.

Table 5 Passengers' average waiting time and walked distance, and average driver's vehicle occupancy for matching peers according to their global reputation values

Peers' global reputation	*P*'s wait. time	*P*'s walk. distance	*P*'s matches	vehicle occup. for matched *D*	*D*'s matches
[0.0...0.25)	478.99 s	297.95 m	13.83%	1.09	5.97%
[0.25...0.50)	409.25 s	293.60 m	24.84%	1.24	17.61%
[0.50...0.75)	368.17 s	280.20 m	29.15%	1.41	30.30%
[0.75...1.0]	342.36 s	268.70 m	32.18%	1.65	46.12%

It is clear from Table 5 that, in an scenario with uniformly distributed reputations and reputation requirements, the higher the reputation of the peers, the higher their chance of finding other peers with matching rides that *trust* them. This is also reflected in the higher vehicle occupancy achieved by drivers with high reputations: they are able to find even more than one matching passenger. On the other hand, passengers with high reputations also fare better: they need to walk shorter distances to/from the pick-up/drop-off location and they wait shorter times for pick-up. The explanation is that passengers with higher reputations can select the best matching offer in spatio-temporal terms from a wider range of offers (at Step 4 of the ride-matching protocol detailed in Section 2.3).

This last aspect strengthens reputation as the mechanism to enforce co-utility in ridesharing: if both passengers and drivers are aware that a higher reputation will allow them to find more (and better) matches, it will be in their own interest to behave well with their peers in order to increase their reputations. In other words, *reputation becomes a new, secondary utility that agents wish to optimize*, along with the primary time and cost utilities. In turn, by promoting the good ridesharing behaviors that peers rate positively (e.g., avoid unnecessary delays or costs, or provide a pleasant company), we make peers' collaboration more harmonious and sustainable.

5 Conclusions

Coming up with mechanisms that contribute to the adoption of ridesharing is of great interest, both for end users and for the society at large. Indeed, ridesharing has the potential to bring significant benefits to the involved agents and a more sustainable management of transportation to society.

In this chapter we have tackled two of the main obstacles deterring ridesharing (i.e., lack of trust and privacy concerns with respect to matching agencies) by means of a reputation-enabled privacy-preserving decentralized P2P ridesharing network. Even though decentralization entails additional challenges in comparison with the usual solutions based on central matching agencies, our ride-matching and reputa-

tion management protocols have been carefully designed so that peers are rationally motivated to adhere to them. In addition to being self-enforcing, our protocols are also *co-utile*, because they bring a number of mutual benefits to the involved peers (e.g., reduced travel costs, fair reputation calculation, better ride matches, etc.).

The reputation management protocol is a clear example of how intrinsically co-utile incentive protocols can spark co-utility in other protocols that are not intrinsically co-utile [22, 25]: the reputation mechanism is self-enforcing and it enables co-utility in the ride-matching protocol, that would not be co-utile without reputation, due to the lack of trust. Moreover, as shown by our simulations, reputation can also enhance the benefits of collaboration (i.e., a higher reputation brings more and better matches). This makes reputation an additional utility that agents wish to maximize, which motivates them to behave well and results in smoother collaboration and more social welfare.

Acknowledgements Funding by the Templeton World Charity Foundation (grant TWCF0095/AB60 "CO-UTILITY") is gratefully acknowledged. Also, partial support to this work has been received from the Government of Catalonia (ICREA Acadèmia Prize to J. Domingo-Ferrer and grant 2014 SGR 537), the Spanish Government (projects TIN2014-57364-C2-1-R "SmartGlacis", TIN2015-70054-REDC and TIN2016-80250-R "Sec-MCloud") and the European Commission (projects H2020-644024 "CLARUS" and H2020-700540 "CANVAS"). The authors are with the UNESCO Chair in Data Privacy, but the views in this work are the authors' own and are not necessarily shared by UNESCO or any of the funding bodies.

References

1. Agatz, N. A. H., Erera, A.L., Savelsbergh, M. W. P., Wang, X. (2012) Optimization for dynamic ridesharing: a review. European Journal of Operational Research 223(2):295–303.
2. Amey, A. M. (2011) A proposed methodology for estimating ridesharing viability within an organization, application to the MIT community. In: Transportation Research Board Annual Meeting.
3. BlaBlaCar, `http://www.blablacar.com/`. Accessed: 31 Jan 2017.
4. Buliung, R., Soltys, K., Bui, R., Habel, C., Lanyon, R. (2010) Catching a ride on the inforamtion super-highway: toward an understanding of internet-based carpool formation and use. Transportation 37(6):849–873.
5. Castro, M., Druschel, P., Hu, Y. C., Rowstron, A. (2002) Exploiting network proximity in peer-to-peer overlay networks. Technical report MSR-TR-2002-82.
6. Caulfield, B. (2009) Estimating the environmental benefits of ridesharing: A case study of Dublin. Transportation Research Part D 14:527–531.
7. Chan, N. D., Shaheen, S. A. (2012) Ridesharing in North America: past, present and future. Transport Reviews 32(1):93–112.
8. Chaube, V., Kavanaugh, A. L., Pérez-Quiones, M. A. (2010) Leveraging social networks to embed trust in rideshare programs. In: Proceedings of the Hawaii International Conference on System Sciences (HICSS), pp. 1–8.
9. Dailey, D. J., Loseff, D., Meyers, D. (1999) Seattle smart traveler: dynamic ridematching on the world wide web. Transportation Research Part C 7(1):17–32.
10. Deering, S. E., Cheriton, D. R. (1990) Multicast routing in datagram internetworks and extended LANs. ACM Transactions on Computer Systems 8(2):85–110.

11. Domingo-Ferrer, J., Farràs, O., Martínez, S., Sánchez, D., Soria-Comas, J. (2016) Self-enforcing protocols via co-utile reputation management. Information Sciences 367(C):159–175.
12. Domingo-Ferrer, J., Martínez, S., Sánchez, D. and Soria-Comas, J. (2017) Co-utility: self-enforcing protocols for the mutual benefit of participants. Engineering Applications of Artificial Intelligence 59:148–158.
13. Domingo-Ferrer, J., Sánchez, D., Soria-Comas, J. (2016) Co-utility: self-enforcing collaborative protocols with mutual help. Progress in Artificial Intelligence 5(2):105–110.
14. Furuhata, F., Dessouky, M., Ordóñez, F., Brunet, M. E., Wang, X., Koening, S. (2013) Ridesharing: the state-of-the-art and future directions. Transportation Research Part B 57:28–46.
15. Jurca, R., Faltings, B. (2003) An incentive compatible reputation mechanism. In: Proceedings of the IEEE International Conference on E-Commerce (CEC), pp. 285–292.
16. Morency, C. (2007) The ambivalence of ridesharing. Transportation 34(2):239–253.
17. Office of Highway Policy Information (2013) Annual Vehicle Distance Traveled in Miles and Related Data. `https://www.fhwa.dot.gov/policyinformation/statistics/2013/vm1.cfm`. Accessed: 26 Feb 2016.
18. Office of Transportation and Air Quality (2016) Measuring Greenhouse Gas Emissions from Transportation. `http://www3.epa.gov/otaq/climate/measuring.htm`. Accessed: 26 Feb 2016.
19. Park, H., Gebeloff, R. (2011) Car-pooling declines as driving becomes cheaper. The New York Times, 29 Jan 2011.
20. Piorkowski, M., Sarafijanovoc-Djukic, N., Grossglauser, M. (2009) A parsimonious model of mobile partitioned networks with clustering. In: Proceedings of the International Conference on COMmunication Systems and NETworkS (COMSNETS).
21. Resnick, P., Zeckhauser, R. (2002) Trust among strangers in internet transactions: empirical analysis of eBay's reputation system. The Economics of the Internet and E-Commerce 11:127–157
22. Sánchez, D., Martínez, S., Domingo-Ferrer, J. (2016) Co-utile P2P ridesharing via decentralization and reputation management. Transportation Research Part C 73:147–166.
23. Schrank, D., Lomax, T. (2007) Urban mobility report. Texas Transportation Technical Report.
24. Stoica, I., Morris, R., Karger, D., Frans Kaashoek, M., Balakrishnan, H. (2001) Chord: A scalable peer-to-peer lookup service for internet applications. In Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '01), ACM, pp. 149–160.
25. Turi, A. N., Domingo-Ferrer, J., Sánchez. (2016) Filtering P2P loans on co-utile reputation. In: Proceedings of the 13th International Conference on Applied Computing, AC 2016, pp. 139–146.
26. U.S. Federal Trade Commission (2014) Data Brokers, A Call for Transparency and Accountability.