Björn Richerzhagen, Nils Richerzhagen, Sophie Schönherr, Rhaban Hark and Ralf Steinmetz. *Stateless Gateways – Reducing Cellular Traffic for Event Distribution in Mobile Social Applications*. In Proc. 25th International Conference on Computer Communication and Networks (ICCCN), IEEE, 2016. ISBN: 978-1-5090-2279-3

# Stateless Gateways – Reducing Cellular Traffic for Event Distribution in Mobile Social Applications

Björn Richerzhagen, Nils Richerzhagen, Sophie Schönherr, Rhaban Hark and Ralf Steinmetz
Multimedia Communications Lab (KOM), Technische Universität Darmstadt, Germany
E-Mail: {bjoern.richerzhagen|nils.richerzhagen|sophie.schoenherr|rhaban.hark|ralf.steinmetz}@kom.tu-darmstadt.de

***Abstract*—The increasing capabilities of modern smartphones lead to the design of mobile social applications focusing on direct interaction between users. Ranging from mobile social networks to fully-fledged augmented reality games, these applications usually operate on contextual information to identify relevant content – most notably, the user's physical location. The resulting locality properties of the interaction between users are not well reflected by the cloud-based, centralized infrastructure utilized in today's mobile applications. Thus, data that is relevant to a group of nearby users is downloaded multiple times via the cellular network. Due to capped and/or expensive data tariffs, this can have severe impact on the user acceptance of such applications. To address this issue, we propose the concept of stateless gateways to augment cloud-based mobile social applications. A stateless gateway is chosen by the cloud to distribute information to nearby interested parties, without requiring any additional state information on the gateway itself. We integrate the concept into a location-based publish/subscribe system and show the resulting performance and cost characteristics through extensive evaluations. Our results show that the stateless concept enables frequent gateway switches, lowering the load on the cellular network by 70 % for the scenario of a mobile augmented reality game. At the same time, our system achieves better fairness characteristics among participants due to a more efficient utilization of gateway nodes compared to a less flexible assignment of gateways.**

## I. Introduction

Smartphones are becoming the premier gateway to applications and services on the Internet. Being equipped with a plethora of sensors and increasingly powerful computational resources, they enable all new kinds of applications. The most prominent ones belong to the category of mobile social applications, which allow users to directly interact with each other. Examples for mobile social applications are simple messaging applications, but also more complex pervasive applications such as augmented reality games. For a large fraction of mobile social applications, the user's current context – most notably, his position – is used to filter relevant information [2]. In case of the augmented reality game, for example, only actions of players within close proximity are of interest. For filtering, applications usually rely on a cloud-based service that matches incoming events to a set of interested users. To this end, publish/subscribe – with extensions to support location-based filtering [8] and client mobility [4], [27] – is a commonly used messaging abstraction, where the cloud acts as broker.

However, as the broker is centralized, the locality properties of the interaction between users are not reflected in the communication system. Even if an event is relevant to a handful of users standing right next to each other, it is distributed individually to each of the users. This leads to an asymmetric utilization of the cellular connection: each event is uploaded to the cloud once, but downloaded multiple times even when devices are in close proximity to each other. Considering capped data tariffs offering only limited cellular bandwidth, each individual transmission has a negative influence on user acceptance. However, relying solely on decentralized ad hoc communication is not viable if the application operates on a global state managed by the cloud. In this work, we present an offloading approach using stateless gateways for event distribution in mobile social applications. We extend location-based publish/subscribe systems with a simple protocol addition to incorporate mobile devices as short-lived relay stations for event distribution. By selecting gateways based on information that is nonetheless available at the publish/subscribe broker, our protocol does not impose additional overhead. We do not require any state to be managed on the chosen gateway node, allowing frequent and lightweight gateway switches. This leads to interesting properties when compared to a global and fairly long-term assignment of gateways as done in existing offloading approaches [18]. Most notably, it allows us to select gateways on a per-notification basis, such that information is always relevant to the gateway itself, leading to a higher offloading ratio and better fairness characteristics.

We implement our concept as an extension to a location-based publish/subscribe system and conduct an extensive evaluation study, focusing on the impact of (i) different gateway selection strategies, (ii) frequent gateway switches, down to the per-notification level, and (iii) a broadcast- vs. unicast-based distribution of events on the achieved performance and induced cost. Our results show that stateless gateways maintain the same quality of service, while significantly reducing the load on the cellular connection. For the real-world use case of a mobile augmented reality game, the load on the cellular connection is reduced by 70 % on average. By utilizing the information available at the broker, our scheme only adds about one percent overhead in terms of traffic for small application payloads. At the same time, fairness properties of the system that would influence user acceptance in a real-world deployment benefit from frequent gateway switches as enabled by our stateless approach.

The remainder of this paper is structured as follows. We introduce mobile social applications in Section II, including a brief description of the involved location-based publish/subscribe system. In Section III we present our extension to a context-based pub/sub system, including a modular approach for gateway selection and clustering. We evaluate a prototype of our proposed system and present the results in Section IV, followed by a discussion of relevant related work in Section V. Section VI concludes the paper.

## II. Scenario: Mobile Social Applications

Before going into details of the proposed approach, we briefly discuss the characteristics of mobile social applications as target scenario. As introduced in the previous section, such applications enable direct interaction between humans via mobile devices, e.g. smartphones, usually including context information such as the physical location of a user. In this work, we focus on the position of a user as the most prominent example for context used in mobile social applications. The generic model of such an application is illustrated in Figure 1. The application consumes events that are related to the user's current location. Such events are generated (i) by other mobile users through interaction with the application, or (ii) by the cloud-based service itself. Depending on the type and purpose of the application, events are generated at different rates, based on user interaction with the application, or location changes.

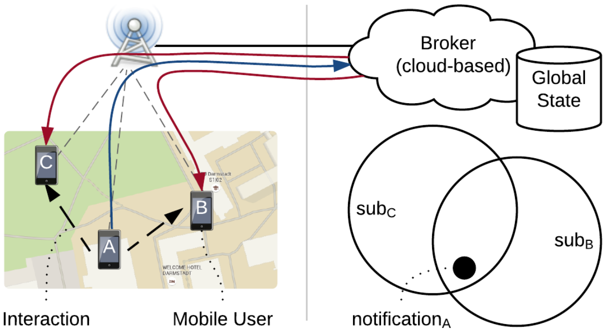


Fig. 1. Overview of the scenario: mobile nodes issue events that are relevant to a set of nearby subscribers. Brokering is done in the cloud, leading to redundant message transmissions back to mobile clients.

To filter events that are relevant to a specific user, the publish/subscribe messaging paradigm is deployed in the cloud, usually relying on extensions that support consumer and producer mobility, as well as location-based subscriptions (illustrated as circular areas in Figure 1). The application running on a mobile device subscribes to relevant information by including a context variable in the subscription and stating a so-called *area of interest*. A simple example would be a circular area of interest specified by $s = (\$loc_i, r)$, where $\$loc_i$ is a placeholder for the location of the user $i$ and $r$ is the radius. Once the subscription is stored on the cloud-based broker, the value for $\$loc_i$ has to be updated. This is done via a separate protocol that updates context variables periodically or based on changes detected at the client (e.g., after movement occurred), or an update for the context variable is included within events that are generated by the application. The latter is usually the case in interactive mobile social applications, as it provides a way to keep context variables updated at low overhead. In such a system, an event generated by a mobile device would, in addition to the application payload, carry an updated context variable $\$loc_i :=$ curr_position that is processed by the broker.

In this work, we focus on applications that exhibit strong locality properties in the interaction between users. These applications become increasingly popular, with one prominent example being Google's augmented reality multiplayer game Ingress. Other emerging concepts are in situ videostreaming applications, where local devices offer different camera perspectives that are then composed into one videostream and distributed to interested viewers [6], [12]. Here, the locality of interaction between users is not reflected in the utilized communication system. Instead, locally relevant information is uploaded to the cloud and then has to be sent to all clients, leading to an asymmetric utilization of the cellular bandwidth, as illustrated in Figure 1. Even in cases where the server delays information to group relevant events into one message, this asymmetric behavior is still reflected in the packet sizes. Sending all events to the central cloud-based entity is required to maintain state and ensure world-wide connectivity among users of the application. However, we can exploit these locality characteristics for the distribution of events to the set of interested clients, as discussed in the following section.

## III. Event Distribution with Stateless Gateways

The concept of stateless gateways is illustrated in Figure 2. As with all gateway-based solutions, a message from the cloud is sent solely to the gateway nodes, which are then responsible to further distribute it to the interested subscribers. However, there exist some key differences between our approach and the gateway concepts known from mobile ad hoc networks. First of all, our system does not require any multi-hop or end-to-end routing scheme in-between mobile nodes. Instead, gateways are selected such that they only serve their one-hop neighborhood. Second, all upload traffic from the mobile users to the cloud goes directly via the cellular network and does not pass through a gateway. This ensures that state updates always reach the central entity (assuming that the cellular connection is reliable). Gateways are only utilized for the distribution of incoming events, as illustrated. To this end, the intended receivers' addresses are simply piggybacked to the notification. The protocol modifications required to allow such functionality are detailed in Section III-C.

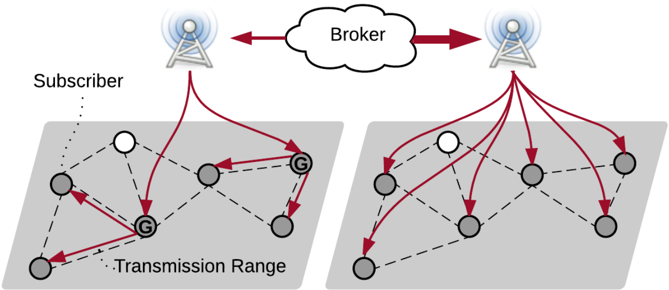


Fig. 2. Concept of stateless gateways (left) compared to the default event delivery scheme relying on the cellular network (right). Duplicate transmissions are reduced. Coordination by the cloud ensures that each notification is delivered only once, even if a node is within range of multiple gateways.

### A. Gateway Selection Algorithms

In general, selecting a gateway node for a region involves (i) clustering the nodes and (ii) ranking potential gateways according to a utility function. Both steps interfere with each other: depending on the result of the initial clustering, some potentially high-ranked gateway nodes are no longer of interest. Therefore, we design our system such that the steps can be executed in any order to later evaluate their impact.

Regarding clustering, we utilize the well-known density-based clustering algorithm DBScan [7], as well as a quadtree-based clustering scheme that ensures clusters are only formed with a configured maximum distance between nodes and gateways, as well as a maximum density of nodes per cluster. With respect to the ranking of potential gateway nodes, a plethora of ranking algorithms can be utilized. Most of these ranking schemes depend on the availability of additional information, such as the current battery power of a node. This information is then used to either ensure a fair load distribution or prevent depletion of less powerful nodes. Our design supports arbitrary ranking functions, given that the required data is available at the broker. Besides two basic schemes (random selection and Lowest-ID), we integrate the Weighted Clustering Algorithm (WCA) by Chatterhee et al. [3] and the Flexible Weighted Clustering Algorithm based on Battery Power (FWCABP) by Hussein et al. [11] into our design.

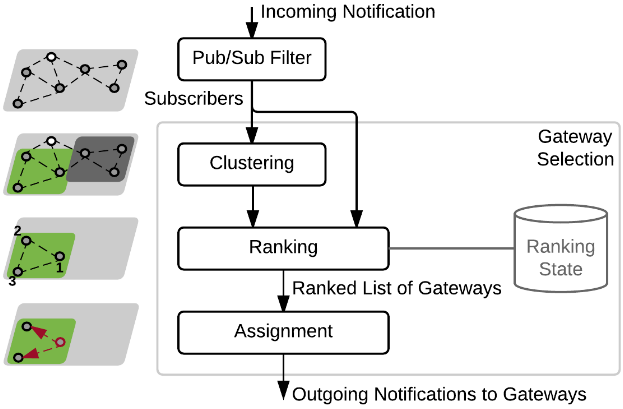


Fig. 3. The process for the per-notification selection of gateways that utilizes the result-set of the publish/subscribe system. Only subscribers are fed into the clustering and ranking functions. The ranking functions might maintain global state for nodes to achieve fair overall utilization.

WCA takes into account the size of a gateway's neighborhood, the signal transmission range, the position, mobility, and the energy level of a mobile node. Additionally, the time a node has already served as a gateway is taken into consideration. With the exception of the current energy level, all required information for WCA can be calculated based on the positions of the individual nodes, which are available at the cloud-based broker. The signal transmission range used by WCA can be set to a pessimistic, expected value for the utilized technology. The best ranked node is selected as a gateway and all nodes within its transmission range are assigned as its leafs and are removed from the list of potential gateways. This process is repeated until all nodes are assigned as either a gateway or a leaf. FWCABP is an extension to WCA, further elaborating on the neighborhood properties of a potential gateway node. For a detailed explanation of the WCA and FWCABP ranking schemes, the reader is referred to [3] and [11], respectively.

In contrast to existing gateway selection schemes for Mobile Ad Hoc Networks (MANETs), we explicitly want to utilize the central, cloud-based broker and its view on the network state for the selection of gateways to avoid additional protocol overhead and state maintenance on mobile clients. As all required information is just piggybacked to the messages sent to gateways by the cloud, there is no need for state maintenance at the gateway nodes. Therefore, switching to another gateway node does not require state transfer in-between the affected nodes, enabling a notification-based assignment of gateway nodes, as discussed in the following section.

### B. Notification-based Gateway Selection

As frequent gateway switches do not introduce any communication overhead, gateways can even be selected on a per-notification basis. Instead of determining a set of suitable gateways for a physical region once, the gateways are selected individually for each notification, allowing for better coverage of the interested subscribers. This process is illustrated in Figure 3. Instead of considering all available nodes as potential gateways, gateway nodes are determined out of the set of subscribers. The set of subscribers is fed into the clustering and ranking functions and is later used for the assignment procedure, where each subscriber is assigned to exactly one gateway node. The notification as well as the required forwarding information is then sent to the respective gateway nodes. Compared to the network-wide selection of gateways, this scheme ensures that the minimal number of gateways is selected for the given notification. Consequently, a higher offloading ratio is achieved, as evaluated in Section IV-C. At the same time, all gateways selected with the notification-based scheme are also subscribers to the given notification, further increasing the efficiency of the scheme. As the set of subscribers is already provided by the publish/subscribe system, retrieving the list of potential gateway candidates does not require additional computation. However, computing a ranking function and the corresponding clusters for each notification – even if they only have to be computed on a smaller subset of all nodes – can pose significant overhead at the cloud-based broker. At the same time, it is not guaranteed that the fairness properties of the given ranking function still persist if only a subset of the nodes is involved in each round. Here, we can trade-off complexity at the cloud against efficiency with respect to the utilization of clients' resources.

### C. Required Protocol Modifications

We propose our scheme as an extension to existing location-based publish/subscribe systems. To deal with clients that are not willing to (or able to) use a second communication interface for direct ad hoc communication, each client has to report its capabilities in terms of local communication. This information is piggybacked as a flag to each outgoing notification and subscription, in order to capture state changes at the broker. Such changes can occur in cases where a client manually turns off an interface, e.g., Wi-Fi. Clients that report they do not support local connectivity are excluded from the gateway selection process and obtain their notifications via the cellular interface. In addition to the flag piggybacked to each message, clients have to report the IP address used on the local interface. This information is only piggybacked to (less frequent) subscriptions to reduce overhead. Ad hoc communication takes place on a configured, well-known port.

If the cloud determines a gateway for a notification, it piggybacks the IP addresses of the nodes that are to be notified by that gateway to the outgoing message. Upon reception of such a message, a node simply relays it using its local communication interface. In our current scheme, we support delivery via UDP unicast or broadcast. Both transmission methods do not provide delivery guarantees, but our experiments show that the overall completeness of notification delivery is not affected (c.f. SectionIV-D). The publish/subscribe system or the application is forced to provide means to deal with packet loss, if the application requires reliable event delivery. As all clients send their own events via the cellular network, the global state is still maintained correctly at the cloud and can then be reported correctly with the next event update. This is common practice in online multiplayer games and similar applications, where a low-latency delivery of events is a key requirement.

The global state at the cloud allows for duty cycling: if a user is not in proximity of other users, the broker could signal that the user may turn off his Wi-Fi interface to save energy, assuming that no other protocol is currently making use of it.

## IV. Evaluation

We conduct an extensive evaluation study of our proposed system based on a prototype implemented within the SIMONSTRATOR-Platform [19]. The platform was previously used to study the impact of direct ad hoc dissemination of events in a MANET for the use-case of an augmented reality game [21], both in simulations and prototypically. Therefore, models for node movement, as well as communication models for the direct ad hoc network and the cellular network based on measurements on the augmented reality game prototype already exist. We rely on simulations due to the scale of the scenario in terms of the number of mobile devices involved. The simulation setup is further discussed in Section IV-A.

Goal of the evaluation is an in-depth assessment of the concept of stateless gateways as introduced in Section III. We first evaluate the selection and clustering strategies under varying scenario configurations and load patterns with respect to their performance and their fairness characteristics. Based on these findings, we further analyze (i) the impact of global vs. notification-based gateway selection (c.f. Section IV-C), (ii) the impact of physical layer broadcasts (c.f. Section IV-D), and (iii) the dependency of our results on the behavior model for clients (c.f. Section IV-E).

### A. Simulation Scenario and Metrics

We model the inner city of Darmstadt based on Open Street Map (OSM) data, including a set of attraction points derived from data openly provided by Google[1] for the augmented reality game Ingress. Nodes are randomly placed on the map and start moving to their nearest attraction point, as illustrated in Figure 4. Once a node reaches an attraction point, it pauses for a uniformly distributed random time between one and five minutes, mimicking interaction with the application. After the pause time, the node select a new attraction point based on a modified version of the SLAW algorithm [15], [25] and starts moving towards it. Node movement is constrained to pedestrian walkways at a speed between 0.5 and 1.5 m/s.

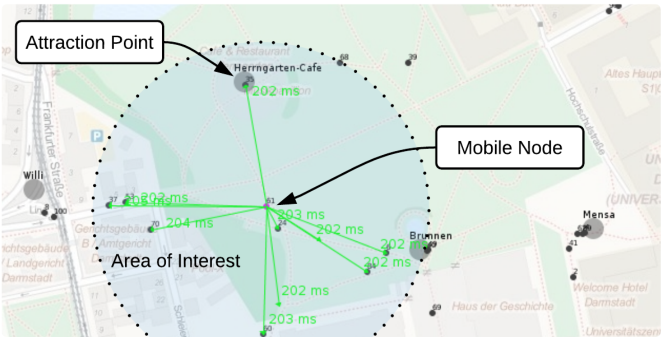


Fig. 4. Region of the scenario with area of interest for one mobile node.

With respect to the application workload, each mobile node publishes one event per second. Nodes subscribe to a circular *Area of Interest* around their current position, with a default radius of 150 meters. Communication with the cloud-based central broker takes place over a reliable, cellular connection with a latency of around 200 ms. Note, that a more elaborate model of the cellular network is not necessary, as we solely assess the offloading effect and not the performance of the cellular network itself. All parameters are summarized in Table I for later reference. Underlined values denote defaults used in the simulations if another parameter is varied. All simulations are repeated ten times with different random seeds and all plots report the 95 % confidence intervals over these repetitions, unless otherwise noted.

TABLE I. Scenario Parameters

| | |
|---|---|
| Simulated Area | City center of Darmstadt, 1.3 km × 1.3 km |
| Simulated Time | One hour |
| Cellular Network | Reliable, 200 ms latency ±100 ms |
| Ad Hoc Connectivity | NS-3 802.11g model [23], no routing<br>Expected communication range: 80 m |
| Movement Model | OSM, Gauss, RWP |
| Pause Times | $0-1$ min, $\underline{1-5\,\text{min}}$ |
| Movement Speed | Pedestrian, $0.5-1.5$ m/s |
| Number of Nodes | 50, $\underline{100}$, 200, 400 |
| Radius of Interest | Circular, $r = 50, \underline{150}, 250$ m |
| Gateway Selection | global, per-notification |
| Event Generation | 1 event per second per node, 128 byte payload |

The following metrics are captured to assess the performance and cost characteristics of the evaluated gateway selection strategies and system configurations. The choice of metrics is based on a recent survey on data offloading techniques by Rebecchi et al. [18].

*a) Completeness:* The completeness denotes the fraction of events that were delivered correctly to all subscribers and is ideally equal to one. Note, that the data transmission between gateways and associated leafs is unreliable. Consequently, the selection of gateways can have an impact on the achieved completeness, if gateways are selected such that not all clients are reachable by direct one-hop ad hoc transmission.

*b) Offloading Ratio:* The offloading ratio describes the fraction of the overall traffic that is offloaded from the cellular network to local distribution by gateway nodes. Assuming a completeness of one, strategies with an offloading ratio closer to one require less gateways for the distribution of events to all subscribers and are, thus, more efficient. An offloading ratio of exactly one is not possible in our system, as each notification needs to be sent to at least one gateway via the cellular network.

[1] Locations of *Portals* (attraction points within the game) can be accessed from www.ingress.com/intel. An account for the game is required.

*c) Gateway Selection Fairness:* Being elected as a gateway, a node has to contribute resources in terms of uploading bandwidth to the overall system. For a mobile, energy-constrained device, this can have a significant impact on the battery lifetime. Therefore, we also assess the selection schemes with respect to the fairness among participants in the system. We use the fairness index $J(x)$ for a metric $x$ among $N$ participants as introduced by Jain et al. [13]:

$$J(x) = \frac{\left(\sum_{i=1}^{N} x_i\right)^2}{N \sum_{i=1}^{N} x_i^2} \qquad (1)$$

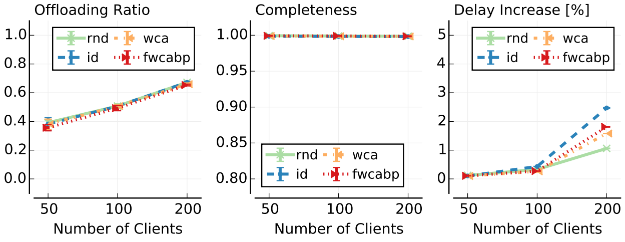
A system with higher $J(x)$ distributes the load across participants in a more balanced way. We assess (i) how often a node was selected as a gateway as *Selection Fairness* and (ii) how many clients the gateway had to serve as *Load Fairness*.

*d) Traffic and Dissemination Delay:* In addition to the aforementioned performance metrics, we capture the overall traffic and the load on the local ad hoc network resulting from the different configurations. If messages are relayed by a gateway, they are slightly delayed compared to a direct delivery via the cellular network. We assess the relative increase in the dissemination delay of events when gateway schemes are used.

### B. *Performance of Gateway Selection Algorithms*

Initially, we compare the performance and cost characteristics of different configurations of the gateway selection algorithms composed out of a cluster algorithm and a ranking function. Available cluster algorithms are DBScan (`dbs`) and the quadtree-based scheme (`qt`) as described in Section III-A. Rating of potential gateway nodes is done randomly (`rnd`), based on the node ID (`id`), according to the WCA algorithm (`wca`), or based on the FWCABP algorithm (`fwcabp`). The resulting configurations either execute the cluster algorithm first and the ranking function afterwards on each of the clusters (e.g., `qt-wca` for quadtree-based clustering and later WCA-based ranking), or they pick gateways one after the other based on the ranking function and assign all nodes within communication range to the respective gateway. In the second case, no additional cluster algorithm is executed and the respective configurations are simply termed `rnd`, `id`, `wca`, and `fwcabp` in all plots.

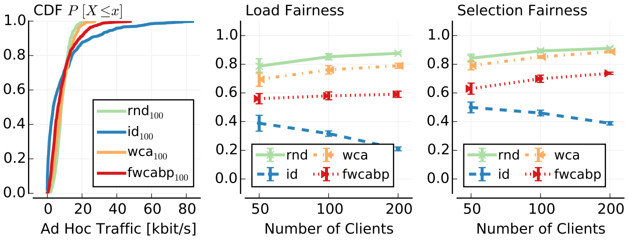
Figure 5 shows the achieved offloading performance for the individual ranking functions without prior clustering. The offloading ratio (c.f. Figure 5a) is comparably high for all four ranking functions. Depending on the number of clients in the scenario, between 40 % and 70 % of all notifications are offloaded and forwarded by a gateway node using local ad hoc communication. The achieved completeness in the system is not affected by the gateways, as it remains constant at one even for larger scenarios, as shown in Figure 5b. However, as messages have to be forwarded by the gateways, the end-to-end dissemination delay increases slightly compared to a purely cloud-based delivery. According to Figure 5c, this becomes more evident for larger scenarios, as messages are queued at a gateway. This is due to the fact that nodes have to compete for access to the wireless medium. The delay increases only slightly (on average 2 % for the largest scenario), making the gateway-based event distribution suitable for interactive applications. Schemes that prefer a gateway for a longer period of time (such as the simple id-based scheme) exhibit worse performance in terms of the dissemination delay, as a consequence of message queues building up. Overall, an increasing density of nodes in the real world is utilized by our system to offload more traffic to gateways, while still maintaining full completeness.



(a) Offloading Ratio (b) Completeness (c) Delay Increase

Fig. 5. Stateless gateways enable cellular offloading of up to 70 % of the traffic especially in dense networks, while maintaining a completeness of one. The dissemination delay increases slightly for larger networks.

The fairness characteristics of the proposed system as well as the resulting ad hoc traffic between gateways and their clients are shown in Figure 6. The average ad hoc traffic caused by the gateway distribution schemes is largely independent of the chosen ranking function, as the total number of gateways remains roughly equal. However, traffic distribution among clients varies significantly, as shown for 100 nodes in the CDF in Figure 6a. The traffic increases with the number of nodes, as notifications need to be delivered to a larger number of subscribers on average. Due to the unevent distribution of traffic, the functions differ in their fairness characteristics, as shown in Figures 6b and 6c. As expected, the id-based gateway selection is unfair as it always prefers nodes with a lower identifier, leading to a skewed distribution of load. The random assignment strategy as a baseline exhibits strong fairness characteristics, as long as only traffic is concerned. As already discussed, a fair load distribution leads to a lower increase in the overall delivery delay (c.f. Figure 5c).

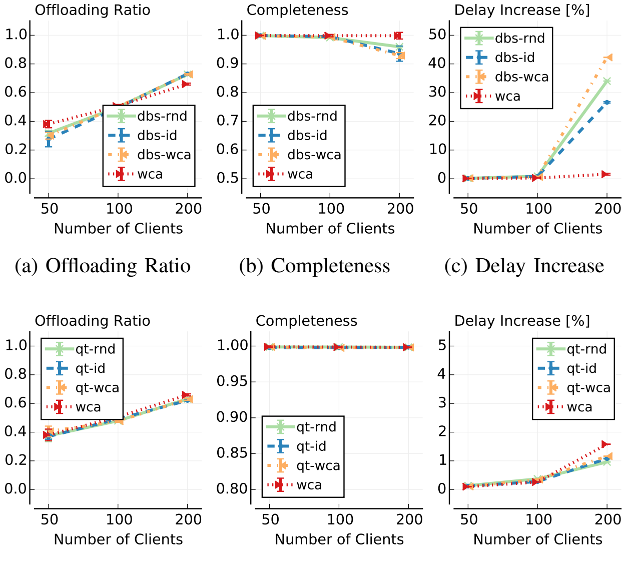


(a) Local Traffic (b) Load Fairness (c) Selection Fairness

Fig. 6. Fairness and cost of different ranking functions used in the gateway selection process. As gateways are already limited to the set of subscribers to a notification, random or round-based ranking functions achieve and maintain a high fairness with respect to the gateway load.

While we do not currently measure the energy consumed on a node, it is a key input parameter for the WCA ranking function. In our current design, we only utilize information that is readily available on the broker node. However, the performance and especially the fairness characteristics of the individual ranking schemes are expected to benefit from additional input data. Gathering such data in a mobile, dynamic setting in a cost-efficient way is an ongoing research effort [22].

The performance characteristics for the cluster-based schemes are shown in Figure 7. For comparison, the WCA ranking function without clustering is also contained in the plots. Notably, the DBScan clustering algorithm does not respect the communication range of nodes, leading to decreasing completeness with a higher number of nodes, as shown in Figure 7b. This is not the case for the quadtree-based clustering algorithm, as it limits the size of a cluster to an expected, pre-configured communication range. Here, the completeness remains one for all scenario configurations. Compared to the ranking-based gateway selection schemes, the quadtree-based clustering scheme leads to slightly better dissemination delays even for large scenarios, as shown in Figure 7f. This is due to the fact that the quadtree-based approach selects more gateways on average, leading to reduced distances between clients and gateways. This, in turn, leads to higher transmission rates on Wi-Fi when using unicast delivery.



(a) Offloading Ratio (b) Completeness (c) Delay Increase

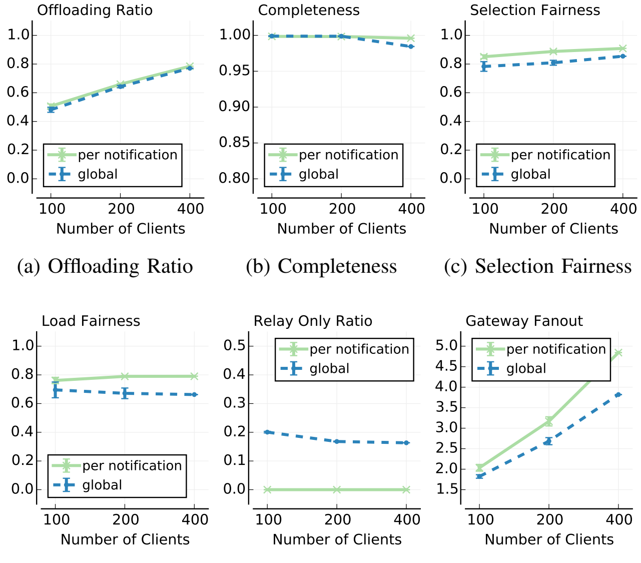
(d) Offloading Ratio (e) Completeness (f) Delay Increase

Fig. 7. Performance of DBScan (a-c) and Quadtree-based (d-f) selection schemes. DBScan leads to suboptimal completeness, as the formed clusters do not guarantee one-hop connectivity. Compared to pure ranking, clustering only results in slightly decreased dissemination delays, as potential gateway nodes are already limited by the set of subscribers of a notification.

As the publish/subscribe system already limits the set of potential gateways to the subscribers of a given notification, a separate clustering step does not lead to a significant increase in performance (c.f. Figures 7a and 7d). Consequently, we omit the cluster-based schemes during the remainder of this evaluation and focus on the WCA-based ranking function to assess other aspects of the proposed system, such as the implications of a per-notification gateway selection.

### C. Global vs. Per-Notification Selection of Gateways

As presented in Section III-B, our system is designed such that it supports frequent gateway switches down to the per-notification level. We compare the performance and cost of a per-notification selection of gateways against a round-based, global assignment of gateways. In the global scheme, gateways are selected out of all available nodes every five seconds. Out of the global set of gateways, we pick the set of suitable gateways for each notification. Consequently, each of the picked gateways is either a subscriber to the event, or it is responsible for a leaf node that is subscribed to the event.



(a) Offloading Ratio (b) Completeness (c) Selection Fairness

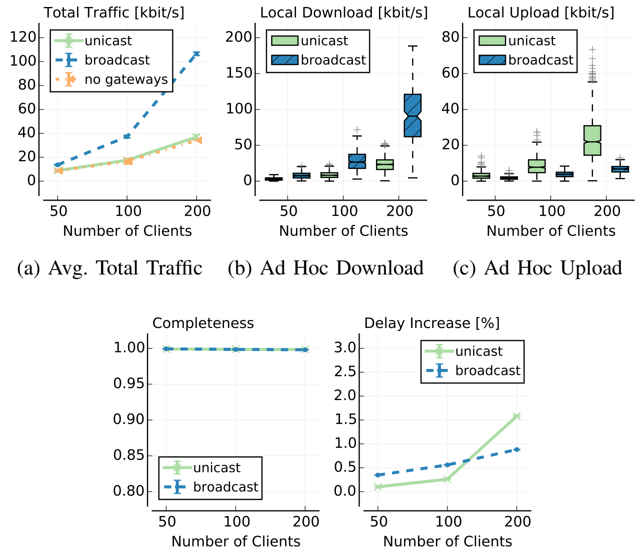
(d) Load Fairness (e) Relay-only Ratio (f) Gateway Fanout

Fig. 8. Global vs. per-notification selection of gateways. Selecting gateways globally leads to less fanout as more gateways are required for each notification due to suboptimal placement. Additionally, 20 % of gateways act only as relays and are not themselves interested in the notification.

The global selection performs as good as a per-notification selection with respect to the achieved completeness and the offloading ratio, as shown in Figures 8a and 8b. The fairness of the selection procedure is reduced slightly for the global case (c.f. Figure 8c). This is due to the fact that we obtain sub-optimal gateways for a given notification, as discussed in Section III-B. This is also reflected in the number of clients served by a gateway on average, as shown in Figure 8f. The distribution of load among gateways (in terms of clients that are served by a single gateway) is also affected by a global selection, as shown in Figure 8d. In the global selection case, a significant portion of notifications forwarded by a gateway are not relevant for the gateway itself. Figure 8e shows the ratio of gateway usages where the gateway solely relayed information without actually benefiting from it. For both global schemes, around 20 % of notifications are relayed and not consumed by the gateway itself, reducing the efficiency of the gateway-based distribution. This, in turn, could lead to decreased user acceptance, as users would need to contribute resources in terms of cellular connectivity as well as battery power without actually benefiting. The results motivate a notification-based selection of gateways for better resource utilization. However, there exists a tradeoff between the computational complexity at the broker and the achieved efficiency of the gateway selection scheme. Especially in case of clustering-based approaches it might not be feasible to execute a clustering algorithm for each notification. Our results indicate that additional clustering prior to executing a ranking function does not lead to significant performance gains. Therefore, we propose to utilize the per-notification selection of gateways combined with application specific ranking functions for a good performance vs. cost tradeoff.

## D. Unicast vs. Broadcast Delivery

As gateways need to disseminate an event to multiple clients in proximity, exploiting the broadcast nature of the wireless medium is a natural step. In contrast to the unicast-based distribution of events individually to each client, the gateway only needs to send the broadcast once. However, sending via broadcast has some practical issues in Wi-Fi. First of all, broadcasting does not include any kind of acknowledgement, neither for the data transmission itself, nor for the reservation of the wireless medium (RTS/CTS in 802.11). As a consequence, broadcast transmissions can lead to packet loss in dense networks. Furthermore, a broadcast is always sent using a robust modulation scheme to guarantee high reach. However, this limits the rate at which data can be transmitted. In practice, broadcast transmissions are often fixed to 1 Mbit/s as the most robust modulation scheme is used. When transmitting via unicast, sender and receiver negotiate a suitable modulation scheme depending on the quality of the wireless channel. As a consequence, the resulting transmission takes place at a higher bitrate, especially if nodes are within close proximity to each other. In the end, there exists a tradeoff between queuing delays caused by multiple consecutive unicast transmissions and lower bitrates of broadcast transmission.



(a) Avg. Total Traffic (b) Ad Hoc Download (c) Ad Hoc Upload

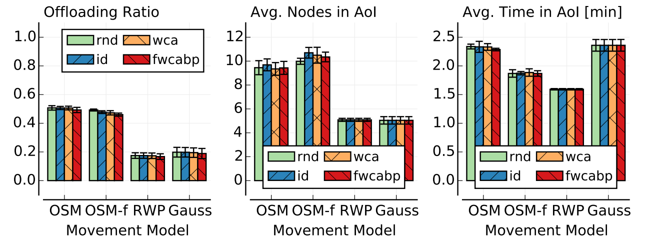
(d) Completeness (e) Delay Increase

Fig. 9. Our gateway scheme adds only negligible overhead by piggybacking information to notifications. When using broadcasts to disseminate events, the download traffic increases significantly due to overheard messages – however, dissemination delay is reduced in high density situations.

This tradeoff can be seen in Figure 9e, as the broadcast-based scheme outperforms the unicast-based delivery for dense networks. This is due to the fact that messages transmitted in the mobile augmented reality game are only 128 byte in size, carrying a location and some identifiers for actions in the game. In terms of completeness, both schemes perform comparably well. However, as broadcasts are received by a potentially large number of clients that are not interested in the notification, the broadcast-based scheme leads to significantly higher download traffic on mobile nodes (c.f. Figure 9b), while at the same time reducing the outgoing traffic of gateways (c.f. Figure 9c). Figure 9a shows the overall traffic profile of the system, including both, cellular and ad hoc transmissions. Compared to delivery of events without gateway functionality, our system adds only negligible overhead if unicasts are used, even for small payload sizes of only 128 byte.

## E. Impact of Client Behavior

The performance of our gateway-based event distribution for mobile social applications heavily depends on the respective scenario characteristics. These are mostly determined by the user behavior in terms of movement and interaction patterns. To this end, we evaluated our system using sophisticated models for mobility and attraction points within the scenario. To evaluate the impact of the scenario characteristics on the performance of the system we compare the results obtained with our OSM-based movement and attraction model against two simple movement models: the Random Waypoint Model (RWP) and the Gaussian mobility model.

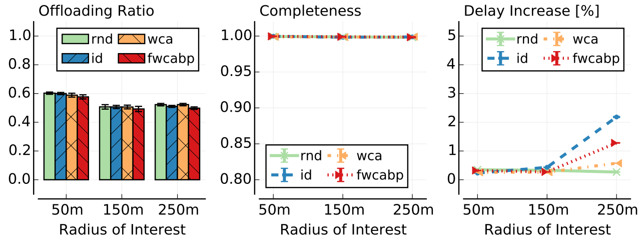


(a) Offloading Ratio (b) Nodes in AoI (c) Time in AoI

Fig. 10. Offloading performance and scenario characteristics for different mobility models. OSM and OSM-f are based on pedestrian mobility, while RWP and Gauss are synthetic models that are often used as a baseline.

The results for varying movement models are shown in Figure 10. OSM-f is a modified version of the map-based mobility model with reduced pause times. Within OSM-f, nodes do only pause for a maximum time of one minute when reaching an attraction point before continuing to the next attraction point. As a result, the average time a node spends within the area of interest of another node is reduced from about 2.4 to 1.9 minutes, as shown in Figure 10c. Still, the offloading ratio decreases only slightly to about 50 %. For the random waypoint movement and the Gaussian movement model, the offloading ratio is significantly worse. This is due to the fact, that in these models there are on average only five nodes within an area of interest (c.f. Figure 10b, compared to about ten for the OSM-based models. Consequently, there is less opportunity for offloading. Our results show the importance of trace-based mobility models that accurately capture relevant characteristics of the scenario that is to be evaluated. In our case, the mobility and interaction models are based on measurements conducted with a real-world prototype of an augmented reality game [20].

In addition to a variation of the number of clients in the scenario, we also vary the radius of interest for the clients. The radius of interest determines the size of the area of interest of a given user and thereby the geographical coverage of the user's subscription. Larger radii lead to more gateways being required to distribute a single notification. The size of the radius of interest is usually determined by the client's interaction with the application, e.g. zooming out and thereby altering a subscription. The resulting performance characteristics are shown in Figure 11. The offloading ratio is slightly decreased for larger radii of interest, as more gateway nodes are required to reach all interested subscribers. The individual ranking schemes do not differ significantly with respect to the offloading ratio and the completeness. However, the message dissemination delay increases with larger radii, especially for the ID-based ranking function. The results indicate that our system is able to scale with the size of the radius of interest, as it simply deploys more gateway. Other approaches that rely on multi-hop dissemination of events quickly suffer from the exponential increase in messages that are transmitted via the local ad hoc network [21].



(a) Offloading Ratio (b) Completeness (c) Delay Increase

Fig. 11. Performance of our system for increased radii of interest. While the completeness is not affected, the larger geographical area leads to more gateways per notification and, thus, a slightly reduced offloading ratio.

In conclusion, our proposed system is able to reduce the load on the cellular infrastructure without imposing additional overhead. This is due to the fact that we utilize the information available within a location-based publish/subscribe system to provide a lightweight and flexible gateway selection approach. Consequently, gateways can even be selected on a per-notification basis, leading to better fairness characteristics and a higher offloading ratio when compared to a global selection of gateway nodes.

## V. Related Work

We briefly discuss related work in the area of (i) cellular offloading, (ii) communication systems for mobile social applications, and (iii) gateway selection in mobile ad hoc networks and highlight the resulting research challenges.

In a recent survey on cellular offloading, Rebecchi et al. [18] highlight key approaches to deal with the exponentially increasing mobile data traffic. The authors group existing approaches into two basic categories: access-point-based offloading as analyzed in [16] and terminal-to-terminal offloading that utilizes direct ad hoc connectivity. They further distinguish between delay-tolerant communication and direct transmissions. Our proposed system utilizes terminal-to-terminal offloading in a non-delayed fashion, as we specifically target the use case of interactive mobile social applications. It is comparable to cooperative data distribution approaches such as [1], [9], [14], [26]. In [14], the authors propose a centrally coordinated scheme, where content is downloaded by a mobile node acting as a proxy (comparable to a gateway). The proxy then distributes the content locally via an ad hoc network. The role of the proxy is shifted periodically to increase fairness. Proxies have to periodically probe their current neighborhood by issuing beacon messages and report their current neighborhood to the coordinating server to facilitate central coordination. The system is targeted towards longer lasting file transfers (e.g., video streaming) with only limited dynamics in client behavior. Our system addresses dynamic scenarios and applications that rely on event-based communication, achieving similar communication savings at significantly less overhead by utilizing state information available at the broker.

Wang et al. [26] propose an offloading scheme that combines the interaction patterns of users in an online social network service with their mobility and contact patterns in real life to estimate a set of suitable relay users. By exploiting the individual access delay that is distinctive to each user within their system, the authors are able to reduce cellular traffic by up to 86.5% while still delivering content in time for each individual user. While the system proposed by Wang et al. is inherently addressing delay-tolerant applications, their correlation study of user behavior in real life and in online social network services further motivates strong locality properties in (mobile) social applications. Barbera et al. [1] further confirm these locality properties by studying the impact of socially important users in an offloading scenario. Han et al. [9] further confirm offloading potential for the use case of a mobile social network, where data is disseminated in an opportunistic fashion. The authors propose three basic algorithms to select a suitable set of gateways for a mobile ad hoc network based on the estimated utility of each mobile node. Instead of minimizing energy consumption while still maintaining fairness [3], [11], the authors solely focus on the minimal set of gateways to deliver a given message to all nodes in the ad hoc network. In our system, we already obtain a limited set of nodes that we have to distribute information to, namely the subscribers to a notification. Furthermore, we do not require multi-hop communication in the ad hoc network, thereby increasing scalability in dense scenarios and limiting energy consumption [24]. In an earlier work [21] we studied the impact of direct ad hoc dissemination of events in the use case of an augmented reality game. Our goal was to reduce the transmission delay for nearby users, while still maintaining global state at a cloud entity. To this end, the cellular connection was always used as a fallback to ensure event delivery. Saving cellular resources was not in the scope of that work. However, the resulting prototype [20] was used to gather data that was used for a refinement of the simulation models used in this work.

In this work, we used WCA [3] and FWCABP [11] as well-known and popular ranking schemes for the gateway selection process. Especially with respect to energy consumption, there exist a range of related works in the area of sensor networks, such as LEACH [10] and its more recent extensions [5]. Evaluating the impact of more sophisticated gateway selection algorithms on the energy consumption of smartphones participating in our system is a direction for future work. However, as stated in [24], we expect only a slight increase in energy consumption due to our scheme, as we do not require extensive MANET routing or multi-hop forwarding in our scheme.

## VI. Conclusion

We present a lightweight extension to existing location-based publish/subscribe systems that facilitates offloading the cellular infrastructure through stateless event distribution gateways. Stateless gateways enable frequent gateway switches – even on a per-notification basis – at no additional overhead.

This allows us to select gateways such that the gateway nodes are always interested in the notification they have to forward, leading to better utilization of resources and increased fairness characteristics of the system. Our system is targeted at mobile social applications, where communication via the application exhibits strong locality properties that can be utilized in the communication system. For the end user, cellular data consumption is reduced at the cost of additional ad hoc communication. Overall, our system achieves an offloading ratio of about 70 % in the real-world scenario of a mobile augmented reality game, with only negligible communication overhead. Even in crowded areas with high node density, the gateway-based offloading scheme adds only around 2 % delay to the overall event dissemination process.

For future work, we plan to evaluate the system in a scaled-down scenario using our prototype of an augmented reality game to assess energy consumption. Besides mobile social applications, other types of location-based services can benefit from our extension of location-based publish/subscribe systems. One example are cloud-based services for the automotive sector, e.g., notifications on specific events happening on a route. Existing offloading solutions for this application scenario do not incorporate any knowledge available in an event-based communication system [17]. Combining the stateless gateway approach with locally available computational facilities, so-called *cloudlets*, is another promising direction for future research. As a cloudlet could facilitate regional brokerage of events on behalf of the global cloud-based broker to reduce the overall latency, it can also incorporate our extension for better resource utilization. At the same time, shifting the gateway selection to local cloudlets helps in maintaining scalability of our approach with respect to computational requirements at the central broker.

## Acknowledgment

This work has been partially funded by the German Research Foundation (DFG) as part of projects C2, C3 and B1 within the Collaborative Research Center (CRC) 1053 – MAKI and by E-Finance Lab e.V., Frankfurt a.M., Germany.

## References

[1] M. V. Barbera, A. C. Viana, M. D. De Amorim, and J. Stefa, “Data offloading in social mobile networks through vip delegation,” *Ad Hoc Networks*, vol. 19, 2014.

[2] A. Beach, M. Gartrell, S. Akkala, J. Elston, J. Kelley, K. Nishimoto, B. Ray *et al.*, “Whozthat? evolving an ecosystem for context-aware mobile social networks,” *IEEE Network*, vol. 22, no. 4, 2008.

[3] M. Chatterjee, S. K. Das, and D. Turgut, “Wca: A weighted clustering algorithm for mobile ad hoc networks,” *Cluster computing*, vol. 5, no. 2, 2002.

[4] X. Chen, Y. Chen, and F. Rao, “An efficient spatial publish/subscribe system for intelligent location-based services,” in *Proc. International Workshop on Distributed Event-based Systems (DEBS)*. ACM, 2003.

[5] H. Deng, C. Yang, and Y. Sun, “A novel algorithm for optimized cluster head selection,” *Science*, vol. 2276, no. 6340, 2013.

[6] N. Dezfuli, J. Huber, S. Olberding, and M. Mühlhäuser, “Costream: in-situ co-construction of shared experiences through mobile video sharing during live events,” in *CHI'12 Extended Abstracts on Human Factors in Computing Systems*. ACM, 2012.

[7] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, “A density-based algorithm for discovering clusters in large spatial databases with noise.” in *Kdd*, vol. 96, no. 34, 1996.

[8] P. T. Eugster, B. Garbinato, and A. Holzer, “Location-based publish/subscribe,” in *Proc. IEEE International Symposium on Network Computing and Applications*. IEEE, 2005.

[9] B. Han, P. Hui, V. A. Kumar, M. V. Marathe, J. Shao, and A. Srinivasan, “Mobile data offloading through opportunistic communications and social participation,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 5, 2012.

[10] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-efficient communication protocol for wireless microsensor networks,” in *Proc. International Conference on System Sciences*. IEEE, 2000.

[11] A. H. Hussein, A. O. Abu Salem, and S. Yousef, “A Flexible Weighted Clustering Algorithm Based On Battery Power for Mobile Ad Hoc Networks,” in *IEEE International Symposium on Industrial Electronics (ISIE)*, 2008.

[12] K. Ito, G. Hirakawa, and Y. Shibata, “Omnidirectional video and sensor data collection and distribution system on challenged communication environment,” in *Proc. International Conference on Advanced Information Networking and Applications Workshops (WAINA)*. IEEE, 2014.

[13] R. Jain, D.-M. Chiu, and W. R. Hawe, *A quantitative measure of fairness and discrimination for resource allocation in shared computer system*. Eastern Research Laboratory, Digital Equipment Corporation Hudson, MA, 1984, vol. 38.

[14] S.-S. Kang and M. W. Mutka, “A mobile peer-to-peer approach for multimedia content sharing using 3g/wlan dual mode channels,” *Wireless Communications and Mobile Computing*, vol. 5, no. 6, 2005.

[15] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong, “Slaw: A new mobility model for human walks,” in *Proc. INFOCOM*. IEEE, 2009.

[16] K. Lee, J. Lee, Y. Yi, I. Rhee, and S. Chong, “Mobile data offloading: How much can wifi deliver?” *IEEE/ACM Transactions on Networking (TON)*, vol. 21, no. 2, 2013.

[17] M. Lee, J. Song, J. Jeong, and T. Kwon, “Dove: Data offloading through spatio-temporal rendezvous in vehicular networks,” in *Proc. International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2015.

[18] F. Rebecchi, M. Dias de Amorim, V. Conan, A. Passarella, R. Bruno, and M. Conti, “Data offloading techniques in cellular networks: a survey,” *Communications Surveys & Tutorials*, vol. 17, no. 2, 2015.

[19] B. Richerzhagen, D. Stingl, J. Rueckert, and R. Steinmetz, “Simonstrator: Simulation and prototyping platform for distributed mobile applications,” in *Proc. EAI International Conference on Simulation Tools and Techniques (SIMUTOOLS)*. ACM, 2015.

[20] B. Richerzhagen, M. Schiller, M. Lehn, D. Lapiner, and R. Steinmetz, “Transition-enabled event dissemination for pervasive mobile multiplayer games,” in *Proc. International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*. IEEE, 2015.

[21] B. Richerzhagen, D. Stingl, R. Hans, C. Gross, and R. Steinmetz, “Bypassing the cloud: Peer-assisted event dissemination for augmented reality games,” in *International Conference on Peer-to-Peer Computing (P2P)*. IEEE, 2014.

[22] N. Richerzhagen, D. Stingl, B. Richerzhagen, A. Mauthe, and R. Steinmetz, “Adaptive monitoring for mobile networks in challenging environments,” in *Proc. International Conference on Computer Communication and Networks (ICCCN)*. IEEE, 2015.

[23] G. Riley and T. Henderson, “The NS-3 Network Simulator,” in *Modeling and Tools for Network Simulation*. Springer, 2010.

[24] J. M. Rodriguez Castillo, H. Lundqvist, and C. Qvarfordt, “Energy consumption impact from wi-fi traffic offload,” in *Proc. International Symposium on Wireless Communication Systems (ISWCS)*. VDE, 2013.

[25] D. Stingl, B. Richerzhagen, F. Zöllner, C. Gross, and R. Steinmetz, “PeerfactSim.KOM: Take it back to the streets,” in *Proc. High Performance Computing and Simulation (HPCS)*. IEEE, 2013.

[26] X. Wang, M. Chen, Z. Han, D. O. Wu, and T. T. Kwon, “Toss: Traffic offloading by social network service-based opportunistic sharing in mobile social networks,” in *Proc. IEEE INFOCOM*. IEEE, 2014.

[27] E. Yoneki, P. Hui, S. Chan, and J. Crowcroft, “A socio-aware overlay for publish/subscribe communication in delay tolerant networks,” in *Proc. ACM Symposium on Modeling, analysis, and simulation of wireless and mobile systems*. ACM, 2007.