CA-Fi: Ubiquitous Mobile Wireless Networking without 802.11 Overhead and Restrictions

Hanno Wirtz, Torsten Zimmermann, Matteo Ceriotti, Klaus Wehrle
Chair of Communication and Distributed Systems, RWTH Aachen
{wirtz, zimmermann, ceriotti, wehrle}@comsys.rwth-aachen.de

***Abstract*—The proliferation, flexibility, and mobility of wireless communication devices provides a readily available basis for ubiquitous mobile communication. However, the network-centric design of 802.11 does not support the spontaneity, continuity, and ubiquitous scope of mobile communication for two reasons: i) The time and maintenance overhead of 802.11 networks prevents spontaneous device interaction, and ii) 802.11 restricts the communication scope to devices in a common network.**

We thus enable ubiquitous wireless communication, independent from 802.11 network structures in CA-Fi, a continuous, low-overhead communication channel *concurrent* to 802.11 associations. CA-Fi augments 802.11 with an association-less broadcast mechanism of up to 30 kB/s, preserving up to 70 % of simultaneous 802.11 throughput and enabling unrestricted and instant networking in the wireless medium. Directly addressing applications in 802.11 frames using Bloom filters enables message aggregation for space efficiency and saves communication overhead by unifying peer and application discovery. CA-Fi inherently supports duty cycling and saves up to 44 % of the energy consumption of the 802.11 ad-hoc mode.

I. Introduction

The proliferation and mobility of diverse wireless communication devices continuously and dynamically surround mobile users with communication opportunities. Among others, data dissemination [1], mobile offloading [2], location-centric content [3], crowd computing [4], and sensing [5] approaches propose to leverage these opportunities. The main challenge in realizing these approaches then becomes *enabling unrestricted wireless communication within the full spatial and temporal ubiquity of communication contexts and opportunities*.

However, 802.11 does not account for the dynamics of ubiquitous mobile networking in its network-centric design. In this design, communication requires the coordination, time, and maintenance overhead of establishing an 802.11 network prior to *any* communication and only affords communication between associated devices. *Network-based* approaches [6], [7] amortize this overhead and leverage the network structure for throughput, routing, addressing, and security (cf. Fig. 1).

This stands in contrast to ubiquitous mobile networking that builds on exploiting short-lived device contacts [1], location contexts [3], and peer discovery [4] (cf. Fig. 1). Namely, i) the time overhead of 802.11 networks impedes truly spontaneous communication, ii) the maintenance overhead renders proactive network provision unaffordable for energy-constrained mobile devices, and iii) restricting the communication scope to associated devices severely hinders exploiting mobile encounters.

As a consequence, ubiquitous mobile networking calls for an *additional* low-effort, permanent communication channel that

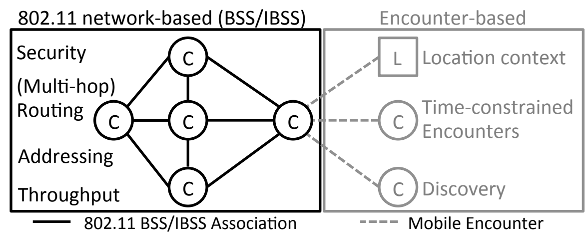


Fig. 1. 802.11 network-based communication only partially meets the requirements of mobile networking, in reach of todays clients (C); 802.11 association restrictions and overhead hinder ubiquitous networking.

affords comprehensive discovery and communication within device and location contexts in transmission range, exceeding the restrictions of 802.11 associations. In this paper, we augment 802.11 with such a channel in CA-Fi (Concurrent Association–less Wi-Fi) to provide a uniform approach to mobile wireless networking. CA-Fi facilitates ubiquitous communication in an *association-less* low-bandwidth broadcast mechanism while preserving and supporting concurrent association-based 802.11 networking. We follow a novel and efficient approach of addressing the diversity of mobile applications using Bloom filters and provide an IP-based publish-subscribe interface for immediate adoption by applications. CA-Fi supports direct application of duty cycling [8] for energy efficiency in its overhead- and coordination-free design. Specifically, CA-Fi's contributions are:

i) A ubiquitous, time- and network-overhead-free communication channel that makes the locality, spontaneity, and unrestricted scope of the wireless medium accessible.
ii) Support for concurrent 802.11 associations allows ubiquitous communication simultaneous to high-bandwidth networking and 802.11 network creation triggered by location or application contexts and peer discovery.
iii) Duty cycling and message aggregation account for the energy efficiency of CA-Fi, while parametrization of messages allows accounting for spatial and temporal communication preferences of applications.

We briefly analyze both the requirements of mobile networking and the shortcomings of available 802.11 solutions in Section II. Section III presents our design of association-less networking with regard to concurrent communication, addressing, application support, and energy efficiency. We realize CA-Fi in the Linux *mac80211* softmac layer and *ath9k* 802.11 driver and evaluate its real-world performance and energy efficiency against 802.11 as well as application benefits in Section IV. In contrast to related works (Section V), CA-Fi provides a comprehensive mechanism for temporally and spatially ubiquitous mobile wireless networking (Section VI).

II. Mobile Ubiquitous Communication: Requirements Analysis

The proliferation and flexibility of mobile wireless devices motivate diverse mobile networking approaches. In this section, we analyze the key requirements of those approaches and highlight how the design of 802.11 standards fail to meet the requirements of ubiquitous networking.

A. Wireless Applications Categories

To motivate the requirements of mobile wireless networking, we categorize the approaches proposed in literature.

Network-based approaches: Approaches that assume a mobile network between devices emphasize performance aspects, such as routing [7] and topography [6] optimization. As such, they require a defined set of participating devices to establish routing and topography bounds and costs. 802.11 implements this set through association to a mobile (multi-hop) network, i.e., virtually binding devices to a single network (infra)structure.

Ubiquitous approaches: Approaches that emphasize ubiquitous communication leverage the spatial [3] and temporal [1], [5] communication context of mobile devices. Fully observing and exploiting this context under mobility thus requires instant and unrestricted access to the wireless medium to i) leverage communication opportunities within short temporal device contacts, and ii) establish a comprehensive view of reachable devices within their dynamic spatial distribution and location semantics. In contrast to network-based approaches, ubiquitous communication approaches thereby embrace fluctuation in the set of communicating devices, exploiting the diversity and dynamics of the respective contexts.

Hybrid approaches: Hybrid applications make situation-dependent use of network-based and ubiquitous communication. Most notably, crowd computing [4] envisions initial work items, e.g., file requests, to be disseminated via ubiquitous transmissions, while subsequent responses, e.g., file transmissions, benefit from high-bandwidth 802.11 networks.

B. Communication Requirements

We now derive the requirements and characteristics of a uniform mechanism for mobile wireless networking.

Enable communication: Stable communication between devices requires a strong binding inside a network infrastructure that offers guarantees as long as the interacting entities can reach each other. Conversely, full leverage of device encounters and location contexts requires a ubiquitous channel enabling instant, network-independent message transmissions and reception.

Minimize coordination overhead: Establishing a network requires coordination to initiate communication. However, in unplanned mobile contexts, prior coordination of, e.g., 802.11 network SSIDs or common channels can not be assumed.

Heterogeneous and concurrent applications: Devices typically execute diverse applications at the same time, demanding simultaneous network-based and ubiquitous communication. This motivates the concurrent execution and mutual triggering of 802.11 and ubiquitous networking.

Flexible addressing: Executing multiple approaches that regard different aspects of ubiquitous communication simultaneously requires efficiently addressing messages to application-specific identifiers. Identifier examples may be locations, social networking IDs, or custom application identifiers.

Energy efficiency: While 802.11 networking implements power saving mechanisms (PSM), duty cycling [8] benefits ubiquitous networking. Especially, adapting awake times in device and location context discovery affords energy savings over always-on schedules or permanent 802.11 scans.

C. Shortcomings of 802.11 Standards

IEEE standardizes [9] wireless networking in several application areas. Namely, 802.11 provides a basis for personal wireless communication, while 802.11p adapts layer 1 and 2 to vehicular networking and 1609.3 specifies vehicular networking services and addressing. We briefly analyze the shortcomings of these standards in enabling ubiquitous wireless communication.

802.11: 802.11 *restricts the communication scope* to devices associated to a common ad-hoc or infrastructure mode network. Devices thereby discard all data frames that do not match the BSSID network identifier of the association. To mitigate this restriction, devices may associate to multiple networks [10], overload 802.11 management frames [11], or accept all wireless traffic in 802.11 monitor mode. Still, this restricts ubiquitous communication to the respective association(s) and limited space of management frames or incurs significant computation and energy overhead. Section IV-D evaluates this overhead.

The *time overhead* of creating a network or associating to one hampers instant leveraging of mobile contexts. Including network scans, this overhead (cf. Section IV-B1) exceeds 13 % of device contact durations in mobility traces with one-second granularity [12], preventing discovery and communication.

Last, 802.11 incurs a *maintenance overhead* of sending beacons and monitoring of associations to operate a network. Continuously operating a BSS or IBSS network, that provides a continuous communication scope, is thus impractical. Additional virtual interfaces [10] further aggravate this overhead.

802.11p and 1609.3: 802.11p allows communication outside of an association while 1609.3 provides service-oriented addressing in the layer 2 WAVE Short Message protocol (WSMP). As such, this partially fulfills the presented requirements.

However, 802.11p modifies the 802.11 MAC and PHY layers for vehicular environments. Similarly, 1609.3 requires an alternative networking stack on top of 802.11p. Designed for vehicular deployments, vendor adoption in personal mobile devices is thus highly doubtful. Service-oriented addressing, using pre-defined Provider Service Identifiers (PSIDs), in WSMP *bases on* traditional 802.11 layer 2 MAC addresses, preventing application-centric addressing at lower layers. Layer 2 addressing further prevents address and message aggregation in frames to increase efficiency of ubiquitous communication.

III. CA-Fi Design

Fig. 2 illustrates the usage scenario of CA-Fi, 802.11 with a concurrent, association-less side channel (Section III-A) that enables ubiquitous discovery, communication, and multi-hop

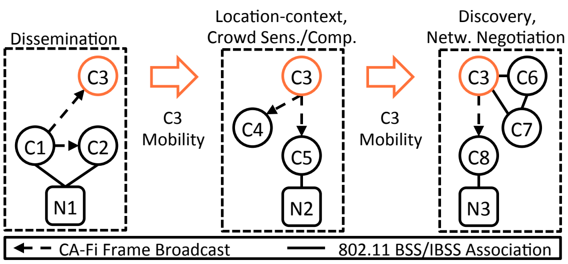


Fig. 2. Concurrent to 802.11 networking (N*i*), CA-Fi enables ubiquitous mobile networking and network negotiation for mobile clients (C) through message aggregation and parameterization in association-less broadcast frames.

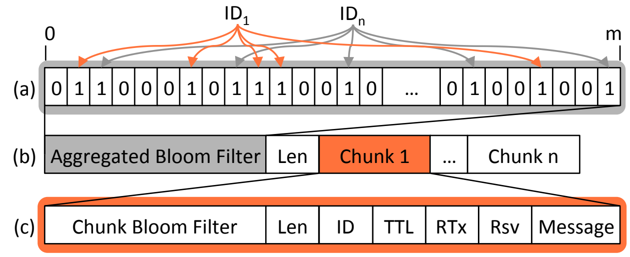


Fig. 3. Bloom filters *(a)* allow aggregation of both address identifiers and CA-Fi chunks *(c)* in 802.11 frames *(b)*. Chunks wrap application messages and parameters: ID, Time-to-live (TTL), and number of Retransmissions (RTx).

forwarding. In this, Bloom filter-based addressing and message aggregation supports execution of multiple applications simultaneously (Section III-B). CA-Fi provides a generic subscription interface and represents application preferences through spatial and temporal message parameterization (Section III-C). We design CA-Fi to incorporate duty cycling and to support spontaneous, purpose-driven creation of 802.11 networks to satisfy bandwidth requirements (Section III-D).

A. *Concurrent Association-less Networking*

CA-Fi mitigates the drawbacks of 802.11 in ubiquitous networking while preserving 802.11 association-based networking and performance. We propose *association-less* wireless networking as broadcast-based communication instantiated and observed spontaneously by devices in communication range, as shown in Fig. 2.

CA-Fi tightly integrates into 802.11 by sending customized, reserved 802.11 frames[1], which do not carry any network identifier, at the respective 802.11 base rate of 1 Mbps or 2 Mbps. Using standardized reserved 802.11 frame types enables all 802.11-capable devices to recognize and receive CA-Fi frames without firmware modifications. Apart from a standard 16 Byte 802.11 header, CA-Fi is thereby able to exploit the frame space, i.e., MTU, of typically 1500 Bytes for addressing and messaging (cf. Fig. 3). We detail the frame structure for addressing and payload in Sections III-B and III-C, respectively.

CA-Fi implements this ubiquitous side channel *concurrently* to 802.11 association-based networking to support simultaneous purpose-driven data-intensive communication. Devices periodically "inject" CA-Fi frames outside of any network context, i.e., the device being associated or not, using a parallel transmit queue. The presence of frames in this queue triggers their periodic transmission in-between potential 802.11 transmissions. While association-less frames thus merge with incumbent network traffic, the inherent 802.11 queue management mechanisms remain untouched.

Concurrent 802.11 associations and the lack of coordination prior to communication prevents the assumption of a pre-defined common 802.11 channel in mobile contexts. Indeed, devices listen to a channel mandated by CA-Fi when unassociated but stay tuned to the respective channel of their 802.11 association otherwise. To enable ubiquitous communication even with associated devices, we exploit the high probability of overhearing wireless frame transmission on adjacent channels [13].

[1]Reusing defined frames, e.g., Beacon frames, misuses the original function and clutters the wireless environment as observed by legacy devices [11].

Devices thus send CA-Fi frames by broadcasting them on a non-overlapping subset of the 802.11 channels, allowing receiving devices tuned to any 802.11 channel to receive the transmission through wireless overhearing. For example, a device associated to a network on channel 2 would send a frame on channels 2, 5, 8, and 11. On each channel, standard 802.11 CSMA-CA is performed before a transmission to avoid interference with existing transmissions. To *receive* frames in CA-Fi, devices listen on their association channel or the mandated one. Associated devices that *send* CA-Fi frames are unavailable in the network when switching channels. Thus, devices indicate their unavailability to the network, i.e., the AP or the IBSS, via a switch to 802.11 power save mode (PSM). Sending devices will then buffer frames for this device.

CA-Fi thus provides mobile communication applications with i) maintenance- and time overhead-free spontaneous transmissions that ii) devices in range opportunistically receive by iii) exploiting the inherent locality and overhearing character of wireless broadcasts to leverage location and device contexts. CA-Fi thereby provides a channel for low-volume communication that enables ubiquitous networking. We envision applications to discover and interact with contexts via low-volume CA-Fi messages and, if necessary, use them to negotiate high-bandwidth 802.11 networks tailored to the use case.

B. *Flexible Bloom Filter-based Addressing*

Departing from network-based communication, layer 3 and 2 addressing mechanisms are no longer viable as no configuration or lookup mechanisms, e.g., DHCP, ARP, or DNS, are applicable. Similar, multicast approaches on layer 2 or 3 fail without a common network association.

Conversely, CA-Fi envisions flexible *application-based* addressing. Given the multitude and diversity of mobile applications and their respective identifiers, we propose to address messages using already available application identifiers. To illustrate the possibilities, mobile social networking [14] may use user or group names while location-based communication [3] may combine location and application identifiers. Using public keys as identifiers allows mobile applications to securely address messages and encrypt content. An addressing scheme supporting this diversity then facilitates devices to i) subscribe to application messages by checking for the respective identifier, ii) store and forward (selected) application traffic, and iii) address messages to user, group, or application identifiers to discover participants or disseminate data.

We assume that the respective identifiers are exchanged out-of-band or within the respective application. However, nearby users may announce themselves when receiving frames carrying solely an application identifier of interest. Similarly, a dedicated wildcard identifier allows polling for applications.

To enable this flexibility in addressing, CA-Fi builds on Bloom filters [15] as address fields in frames (cf. Fig. 3). A Bloom filter is a bit array of size m used in conjunction with a set of k independent hash functions $h_1, ..., h_k$, each mapping an arbitrary input element to one bit position between 1 and m. The filter contains an element e if the k bit positions derived from $h_1(e), ..., h_k(e)$, are set to 1. Using Bloom filters, we obtain a *flexible* and *efficient* addressing scheme, capable of reducing the overhead by enabling *aggregation*.

Flexibility: Bloom filters abstract from variety of the identifiers used as input. As such, applications can simply make use of their given identifiers. By abstracting from input identifiers, CA-Fi is able to compress and combine application-specific identifiers in a uniform, space-efficient address field. Furthermore, devices may pre-compute the Bloom filter elements of relevant own or application identifiers for comparison with received messages. Bloom filters thus allow incorporating the diversity of mobile wireless networking in a uniform addressing mechanism, in contrast to single-purpose layer 3 or layer 2 (multicast) addresses.

Efficiency: Similar to layer 3 and layer 2 addressing, Bloom filters allow fixed-space address fields. This allows the implementation of efficient lower-layer filters for received frames, analogous to the (additive) bit mask comparison of the BSSID field when receiving 802.11 frames or efficient Berkeley packet filters when capturing frames. This motivates our design of maintaining the aggregated bloom filter in each frame, in addition to single chunk bloom filters, to support a per-frame discard decision to avoid useless frame processing,

Adding identifiers to a filter is efficient through hash operations to generate and bitwise OR operations to insert identifiers. Similar, checking for an identifier only requires comparison of the identifier and filter bitfields. Thus, devices can efficiently check whether they want to receive or discard frames based on their Bloom filters.

Aggregation: We assume that mobile users leverage the heterogeneity of mobile networking approaches and take part in multiple applications simultaneously. In traditional ISO/OSI layer-compliant 802.11 networking, each application message triggers a dedicated layer 3 and layer 2 encapsulation and subsequent 802.11 frame transmission. We refrain from this design for three reasons. First, any association-less transmission of CA-Fi frames harms the performance of a concurrent 802.11 network association. Second, inadvertently re-broadcasting received CA-Fi frames leads to a broadcast storm, causing interference and collisions [16]. Last, each transmission depletes the energy resources of mobile devices.

CA-Fi thus makes use of its layer-less design and Bloom filter-based addressing scheme to *aggregate* application messages on sending and forwarding devices, as hinted at in Fig. 2. Fig. 3 shows the layout of an 802.11 frame containing multiple applications messages encapsulated in *chunks*. We detail the purpose of the shown parameters in Section III-C. The frame bloom filter thereby aggregates the identifiers of the contained messages, allowing receiving devices to implement layer 2 identifier filters on a fixed-size address field. Aggregating

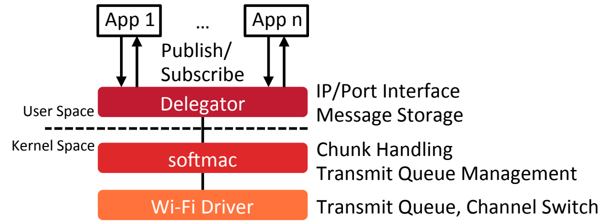


Fig. 4. Components and respective functionality in CA-Fi.

multiple messages in a single frame thus saves transmissions and the associated energy overhead.

When devices receiving a message that is addressed to them, they remove the respective chunk from the aggregated frame. To forward the remaining messages (chunks), devices re-construct the frame and the Bloom filter according to the remaining messages. To this end, chunks carry a Bloom filter that only contains the identifier of the respective message. By deleting the chunk and bitwise ORing the remaining Bloom filters, devices are able to reconstruct the frame Bloom filter. Chunks carry the Bloom filter instead of the original identifier for space efficiency and identifier obfuscation.

Alternatives: *Counting* and *spectral* Bloom filters specifically facilitate insertion and deletion. However, we use standard Bloom filters in our current design due to i) their minimal space requirement, ii) identifiers being the input to the filter, negating the benefits of counting filters that assume data (streams) to be the input, and iii) frequent checks and reconstruction of the filter at forwarding devices, requiring computational efficiency.

C. Application Support

The diverse benefits afforded by mobile applications motivate their concurrent existence on the same mobile device. To allow for easy adoption by applications, CA-Fi provides an (IP, Port)-based *delegator* interface. However, given the spontaneous and uncoordinated communication scenario, we depart from the traditional layered networking approach of transport control above routing functionality in the network layer. Fig. 4 illustrates the message flow in CA-Fi.

Applications interact with the user-space interface in a publish-subscribe fashion by publishing (sending) messages to a port and subscribing (listening) to any messages that arrive on this port. Non-permanent applications may also delegate message handling and storage to the delegator, e.g., to continuously sample location contexts without running the actual application. A user may thus participate in a number of concurrent applications through multiple subscriptions.

Mobile wireless networking furthermore emphasizes diverse communication characteristics, e.g., pure location-centric communication [3], maximum dissemination [1], continuous result collection [5], or event-driven, timely message delivery [4]. As a first step to meet this heterogeneity of preferences, CA-Fi supports message parameterization on two axes: i) A Time-To-Live value (TTL) indicates the spatial dissemination range. ii) The number of retransmissions (RTx) by the original and each forwarding device determines the temporal validity and dissemination. Store-and-forward mechanisms may thus control message dissemination through high TTL and appropriate RTx values. Conversely, low TTL value and RTx values indicate immediate local communication. Applications set these values in their messages (cf. Fig. 3) along with a pre-defined application ID.

The delegator then encapsulates application messages in CA-Fi chunks along with the respective parameters and identifiers (cf. Fig. 3), and passes chunks to CA-Fi functionality in the softmac layer. There, chunks are aggregated into 802.11 frames for space- and energy-efficiency. Last, the 802.11 driver periodically transmits CA-Fi frames by iterating over a subset of channels, concurrent to a given association. For receiving devices, this message path is reversed. Chunks are handed to the delegator if identifiers match or are stored, aggregated into 802.11 frames, and forwarded according to the parameters. CA-Fi removes chunks if either the TTL or RTx value is zero.

Chunks thereby allow exploiting the full design space of message handling. For example, CA-Fi may be instructed to prioritize specific application IDs that indicate emergency or urgent messages. Storing a message digest of chunks, e.g., frame check sequences or hashes, allows receiving and forwarding devices to identify and filter duplicate chunks.

D. Energy Efficiency

When not associated to an 802.11 network, CA-Fi devices save the energy overhead of maintaining a BSS association or an IBSS while still facilitating wireless communication. Thus improving on network-based communication, CA-Fi further benefits from duty cycling, i.e., scheduling sleep and awake cycles while maintaining a high probability of successful device encounters. Acknowledging the large body of existing research, e.g., [8], we inherently design CA-Fi to incorporate schedules as coordinated by proposed approaches. Namely, CA-Fi allows adjusting the frequency of sending and listening for frames in accordance to the implemented schedule. CA-Fi's independence from 802.11 networking overhead thereby affords truly opportunistic duty cycling, further motivating the exclusive on-purpose creation of 802.11 networks if the expected communication amortizes the 802.11 overhead.

When associated, CA-Fi performs unmodified 802.11 power save mode (PSM) as indicated by the delivery traffic indication (DTIM) period set in the network to not negatively influence network performance through, e.g., missed packets. This is because we place higher priority on traffic in purposefully created networks so as to maximize their efficiency and minimize their amortization time. Furthermore, given typical Beacon intervals of 100 ms and low DTIM periods of 1 or 2, devices may adjust the PSM schedule to the given duty cycle schedule and vice versa. For example, a device may adjust its probe slots in restricted randomized probing in [8].

IV. Evaluation

We implemented CA-Fi in Ubuntu 12.10, modifying the mac80211 softmac layer and the 802.11 driver. In contrast to alternative network stacks, e.g., in 802.11p, this allows simple adoption in 802.11 drivers and current operating systems. For evaluation, we use Lenovo Ideapad S10-3 netbooks with 1.5 GHz dual-core CPUs and Atheros AR9285 802.11n wireless cards using the popular ath9k Wi-Fi driver. The netbooks serve as mobile clients that send and receive CA-Fi frames concurrent to a BSS or IBSS association.

TABLE I. Bloom filter dimensions [15] with regard to the false-positive-rate f and message payload in 802.11 frames.

m (bit)	k	n	f	Average message payload (byte)
8	3	2	≤ 0.1	741.5
24	7	2	≤ 0.01	738.5
32	10	2	≤ 0.001	737
48	3	10	≤ 0.1	137.2
96	7	10	≤ 0.01	130.6
144	10	10	≤ 0.001	124
120	3	25	≤ 0.1	38.32
240	7	25	≤ 0.01	22.72
360	10	25	≤ 0.001	7.12

In this section, we first briefly specify the dimension tradeoffs in our Bloom filter addressing design. Second, we evaluate the feasibility and performance of association-less communication. We then describe and evaluate spontaneous network negotiation using CA-Fi to originate high-bandwidth, purposeful 802.11 networking from ubiquitous encounters and discovery. Third, we analyze the energy profile of CA-Fi and the impact of duty cycling. Last, we prototypically highlight the simplicity of realizing BUBBLE Rap [1], Floating Content [3], and MobiClique [14] in CA-Fi and evaluate the performance gains in comparison to 802.11-based networking.

A. Bloom Filter Parameterization

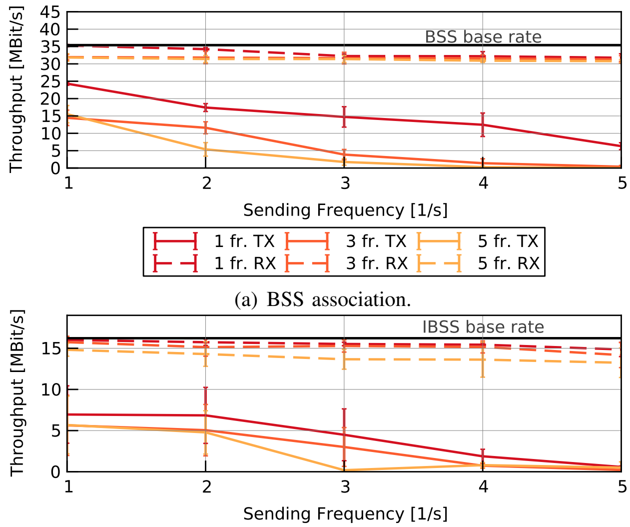
The dimensions of a Bloom filter comprise the size of the bit field m, the number of hash functions k, and the number of expected items in the filter n. The choice of m, k, and n induces a false positive rate f, i.e., falsely assuming a filter includes an element because existing elements set all the respective bit positions to 1. Furthermore, as m bits are required for the main Bloom filter and the Bloom filter in each chunk, the choice of m influences the available space in the 802.11 frame. Table I lists example dimensions for m, k, and f with regard to the possible number and size of chunks, assuming 1500 byte payload in an 802.11 frame, the typical MTU of layer 3 interfaces.

In our prototype implementation we require the false-positive-rate f to be less than 1 %. Assuming a maximal number of 10 messages per frame, a Bloom filter size of 96 bit achieves a good compromise between the possible number of messages and the individual payload.

B. CA-Fi Performance

We evaluate the three distinct pillars of providing an association-less communication channel in CA-Fi: i) The communication performance of CA-Fi concurrent to an association and without an association, ii) wireless overhearing with regard to distance, and iii) the impact of parameters TTL and RTx on message dissemination.

1) Association-less Wireless Networking: We first measure the impact of *sending* and *receiving* CA-Fi frames while striving for maximal association-based TCP throughput using *iperf*. First, a netbook is associated to an 802.11n AP and sends (TX) CA-Fi frames concurrently to maximizing TCP throughput towards the AP. Second, the netbook receives (RX) CA-Fi frames from another (unassociated) device while serving as the receiver of TCP throughput from the AP. To measure the impact on IBSS associations, two netbooks establish the IBSS and act as *iperf* client and server, respectively. In sending an increasing number of frames per second concurrent to maximal TCP



(a) BSS association.

(b) IBSS association (legend as in (a)).

Fig. 5. Impact of concurrent association-less sending (TX) and receiving (RX) of increasing number of frames on TCP throughput.

throughput, we provide a worst case performance evaluation. Namely, we assume that the device has an unlimited number of chunks to send and simultaneously executes a data-intensive network-based application.

Fig. 5 thus shows the respective throughput degradation with regard to the frequency of sending up to 5 CA-Fi frames per second. Each frame contains one chunk of 1500 bytes and is discarded by the receiver upon reception. Due to the overhead of switching 802.11 channels, the throughput decreases with increasing sending frequency, regardless of the network type. In contrast, concurrently receiving CA-Fi frames only negligibly impacts 802.11 throughput.

Associated devices may adjust their association-less sending frequency as appropriate for their association. In this, a sending frequency of two frames per second still preserves 50 % of throughput. Also, unassociated devices may carry the main load of association-less communication, allowing reduced effort of associated devices in heterogeneous scenarios with unassociated and associated devices that execute multiple applications.

Precisely, we evaluated that devices that do not maintain an association are able to send 20 CA-Fi frames of 1500 Byte per second, resulting in a throughput of 30 kB/s. The overhead of switching 802.11 channels thereby prevents possible rates of 125 kB/s at a base rate of 1 MBit/s. For comparison, Fig. 6 shows the number of CA-Fi frames sent in relation to the time overhead of 802.11 operations as measured in our evaluation setting. From this comparison, we derive that i) CA-Fi allows substantial communication within the time overhead of 802.11, and ii) thereby enables communication opportunities that are otherwise lost to 802.11 overhead. In a large-scale mobility trace with one-second granularity [12], opportunities made available this way amount to 13 % of all contacts.

While the achieved throughput allows unassociated devices to fully partake in association-less networking, it does not suffice for data-intensive applications, such as photo-sharing or file transfers. As hinted at in Section III-A, we envision association-free communication to provide a basis for negotiation and establishment of purposeful 802.11 networks

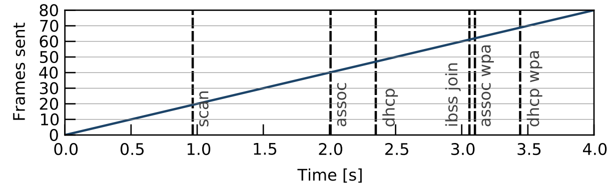


Fig. 6. Association-less CA-Fi frames within the time overhead of 802.11.

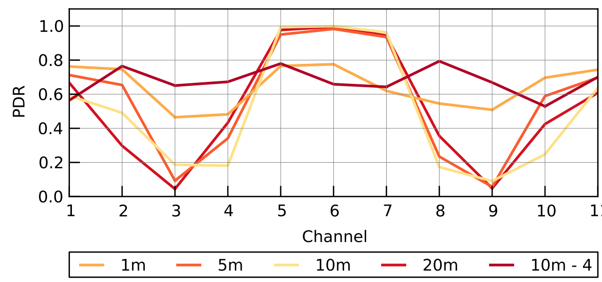


Fig. 7. Wireless overhearing between mobile devices for increasing distances. Sending on four channels (10m-4) increases the reception rate.

that fulfill the respective indicated application demands and amortize the associated overhead. In this, we follow the intuitive notion of continuous, ubiquitous association-less communication triggering 802.11 networks at locations or with communication partners of interest (or merit).

2) Mobile Wireless Overhearing: We propose to exploit mobile wireless overhearing, as quantified for stationary dedicated networks [13], to enable association-less communication without prior sender-receiver coordination. To evaluate the feasibility, we measure the packet delivery rate (PDR) between two netbook devices and emulate a mobile scenario by measuring at distances of 1 m, 5 m, 10 m, and 20 m.

Fig. 7 shows the PDR with respect to the transmission distance when transmitting on the non-overlapping channels 1, 6, and 11. Measured in an office environment, the results account for ambient 802.11 traffic and local 802.11 deployments, as indicated by the poor PDR of listening to the transmission channels 1 and 11. The low delivery rate on adjacent channels thereby motivates our heuristic of sending on the channel of the current association and three additional channels, or on four channels when not associated (cf. Section III-A). In figure 7, "10m - 4" displays the PDR when following this heuristic and sending on channels 2, 5, 8, and 11, allowing a distribution of higher PDRs over all channels. From these results, we conclude the feasibility of mobile wireless overhearing, although the respective performance depends on the local 802.11 environment.

3) Message Parameter Impact: CA-Fi reflects the communication requirements of mobile approaches through message parameterization, namely a TTL parameter indicating the requested spatial dissemination and a RTx parameter for temporal validity. To evaluate the impact of parameter choices, we measure the packet delivery rate (PDR) in a real-life mobility scenario of 789 devices recorded over the course of one day [17]. Each device originates a single message and sends or discards all stored messages with respect to their parameters at each timestamp. We measure the PDR relative to the total PDR, i.e., if each message was received by each device. We skip the first 550 timestamps (1 timestamp = 20 s) due to negligible activity

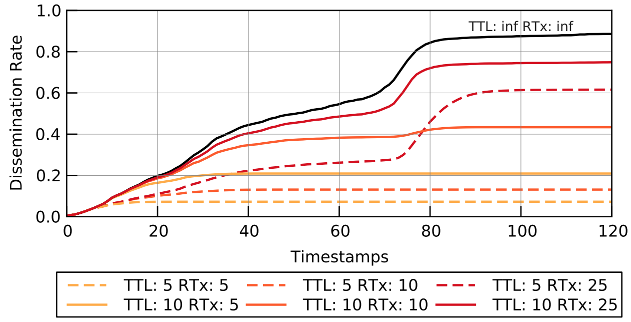


Fig. 8. Parameter impact on message dissemination in real-life mobility [17].

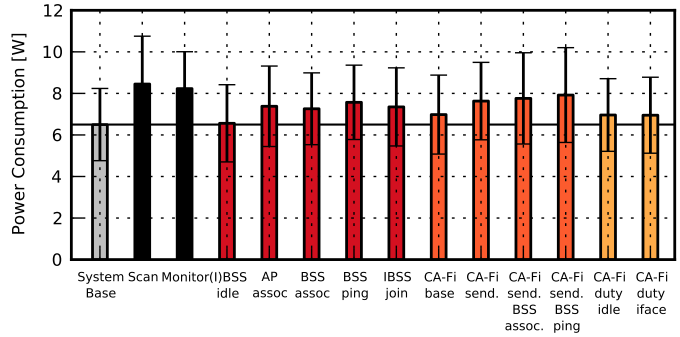


Fig. 9. CA-Fi energy consumption compared to 802.11 in addition to base system energy consumption (6.5 W).

and only display 120 timestamps as no further progress is made afterwards for finite parameter values.

Fig. 8 shows the PDR evolution for representative combinations of TTL and RTx in comparison to the upper PDR bound achieved by setting both parameters to infinite. Combinations of low TTL and RTx values limit dissemination to local scopes due to the limited number of possible forwarding steps and the restricted validity time. While increasing the TTL value allows messages to spread further, longer timeframes of retransmitting a message enables exploiting additional contact opportunities. This tradeoff becomes apparent when comparing the PDR evolution of (TTL: 10, RTx: 10) and (TTL: 5, RTx: 25) at timestamp 80. Message parameterization thus allows defining a communication scope appropriate to application requirements.

C. Tool: Network Negotiation

CA-Fi simultaneously enables low-effort, ubiquitous communication *and* association-based networking for data-intensive communication. For example, crowd sourcing [4] and sensing [5] may quickly distribute small requests through association-less broadcasts and collect data-intensive results in dedicated networks. Contrary, network-based approaches [6] require a means to establish a mobile network. We thus implement spontaneous low-effort 802.11 network negotiation.

We realized spontaneous network negotiation on top of CA-Fi. The requesting device broadcasts a CA-Fi chunk carrying the respective application type with a TTL of 1. A predefined value in the reserved field ("Rsv", cf. Fig. 3) indicates a network request, while the payload contains the requested network SSID for identification. The chunk may address a specific user or application via the respective identifier. Using the public key of a user or an application allows encrypting the message payload. A receiving device replies by incrementing the "Rsv" field and encrypting a WPA2 network key in the payload. Including the client IP address avoids DHCP overhead.

Over 30 runs, devices on average negotiate and create the requested network and establish an association within 3.5 s. The difference of 0.5 s to an association with an existing 802.11 network (cf. Fig. 6) is due to the on-demand network configuration and creation as well as WPA2 key generation.

D. Energy Efficiency

We evaluate the energy efficiency of CA-Fi by first measuring the actual energy consumption of our prototype implementation and the impact of duty cycling schedules. Our measurement setup follows the setup proposed in [18], i.e., via an oscilloscope that measures the current draw of the netbook device. We compare the energy consumption of association-less communication with 802.11 networking as well as concurrent CA-Fi and 802.11 operation. As we are not able to isolate the built-in network card, we first measure the base energy consumption (6.5 W) of the device with the 802.11 card deactivated and driver unladed and adjust all further results by this factor. We further experiment with two options of duty cycling, namely i) periodically toggling the idle mode of the 802.11 card to prevent activity ("CA-Fi duty idle") and ii) removing the wireless network interface and unloading the 802.11 driver ("CA-Fi duty iface"). In idle mode, the 802.11 driver is loaded but the Wi-Fi card does neither send nor receive nor process any 802.11 frames. Please note the prototypical character of these options. We currently work on integrating duty cycling mechanisms in the OS and networking stack.

Fig. 9 shows the mean energy consumption of 802.11 device states and CA-Fi functionality measured over 30 runs of 100 s. Enabling continuous ubiquitous communication (CA-Fi base) by listening for additional 802.11 frames reduces energy consumption by 44 % compared to 802.11 states that allow communication (AP and BSS assoc, IBSS join). Moreover, ubiquitous communication in CA-Fi requires significantly less energy than the 802.11 monitor mode, i.e., unconditional reception of 802.11 frames, due to low-level filtering of frames by addresses in Bloom filters prior to further processing in the stack. Duty cycling the device via the 802.11 "idle" state (CA-Fi duty idle) with an awake time of 100 ms and complementary sleep cycles of 900 ms allows a small reduction of the energy consumption compared to "CA-Fi base". Disabling the device's Wi-Fi functionality further affords a reduction of energy consumption, while still supporting association-less communication in periodic awake cycles (e.g., according to [8]).

CA-Fi's energy consumption of *sending* 2 frames 3 times a second (CA-Fi send.), equals the consumption of an associated station sending 6 ICMP packets/s (BSS ping). In real-life scenarios, a lower sending frequency may be sufficient, resulting in lower energy consumption. Concurrent sending in 802.11 and CA-Fi (CA-Fi send. BSS ping) moderately exceeds pure 802.11 (BSS ping), due to switching channels 3 times per second. We argue that the addition of a continuous spatial and temporal communication channel justifies this overhead.

E. Application Utility and Performance

Complementing the micro-benchmark evaluation of CA-Fi, we illustrate the design space of mobile applications in 802.11 as well as the benefits provided by CA-Fi. We thus implement the discovery and communication aspects of BUBBLE Rap [1] message forwarding, location-based Floating Content [3], and mobile social networking in MobiClique [14] on top of CA-Fi.

Implementing mobile applications using 802.11 entails *peer discovery* and subsequent *content exchange*. A permanent, well-known network, in which all applications coexist, satisfies both peer discovery aspect, e.g., using a pre-defined SSID, and content exchange, but is costly in terms of energy and wasteful if no peers are around [8], [14]. Furthermore, typical device usage suggests that users would refrain from permanently dedicating 802.11 interfaces to operating this network.

Conversely, each application may operate a designated 802.11 network, enabling peer discovery by way of a defined SSID [14] per application. To augment peer discovery with content exchange, applications might employ Beacon stuffing [11] to *push* low-volume information. However, message reception then requires costly 802.11 scans (cf. Fig. 9), while responding via stuffed beacons results in an awkward design and polluted 802.11 environment. We thus argue that content exchange using 802.11 requires a network association.

1) Application Design in CA-Fi: We briefly outline each application design and its implementation on top of CA-Fi.

BUBBLE Rap [1] forwarding bases on community memberships of nodes, expressed by labels, and a global rank as well as one within local communities. Nodes then forward messages greedily, i.e., over the encountered nodes of highest rank, with global forwarding bridging local forwarding in communities.

Floating Content [3] associates content with anchor zones in which it is kept "floating", i.e., forwarded and replicated. Nodes then discover peers and their content and exchange it depending on node locations and the respective anchor zones.

MobiClique [14] enables social networking over opportunistic links between mobile users. Upon discovery, devices exchange social profiles or their updates as well as content based on pre-defined relationships.

CA-Fi implements the communication mechanisms, i.e., discovery and exchange, of these applications within a two-step process. First, devices subscribe to the respective application identifier, e.g., "BubbleRap", meaning CA-Fi accepts frames carrying this identifier in the aggregate Bloom filter (cf. Section III-B). Upon reception of a matching frame, i.e., *discovery* of a peer announcement, CA-Fi delivers the message payload to the application. In our implementation, message payload contains, next to the application-specific peer ID, the set of labels and ranks in BUBBLE Rap, the list of stored content locations and anchor zones in Floating Content, and a digest of profile updates in MobiClique, respectively. Announcement frames further carry a payload flag to indicate if the content exchange requires a high-bandwidth 802.11 network.

Second, applications respond directly by addressing the peer ID. If the exchange requires an 802.11 network, CA-Fi starts the network negotiation as described in Section IV-C. Otherwise, it immediately responds with a CA-Fi frame containing, next to its peer ID, messages along with their labels in BUBBLE Rap and the list of replicable content items in Floating Content, respectively. In mobile social networking, transmitting the peer ID suffices for the peer to check for friendship relations or requests and initiate or block communication.

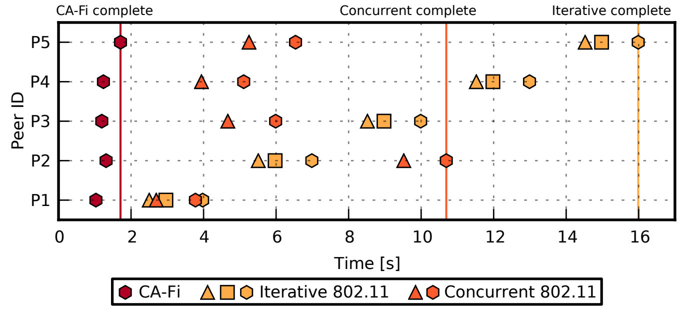


Fig. 10. Peer information exchange steps and information payload over the required time in 802.11, concurrent 802.11 [10], [19], and CA-Fi.

In this, CA-Fi inherently enables both peer discovery within short-lived encounters as well as low volume message exchanges in a self-contained communication mechanism. Please note that, in contrast to 802.11, multiple applications can co-exist in single CA-Fi frames, mitigating the overhead of multiple 802.11 networks.

2) Performance Evaluation: We evaluate the performance gains of building applications on top of CA-Fi. To this end, we measure the time overhead of exchanging a message with five peers in BUBBLE Rap, as a representative for mobile applications, a) if each peer operates a well-known 802.11 network with a pre-defined SSID, e.g., `bubblerap`, and b) if peers communicate using the outlined CA-Fi-based implementation. In 802.11, we model a bidirectional exchange using two *ping* messages.

Fig. 10 thus shows the average time overhead over 30 runs for each peer exchange step in 802.11 and CA-Fi. To model a best-case 802.11 scenario, the peer only needs to scan once; the duration of roughly 1 s is included in the first 802.11 association. Subsequently, it is able to iteratively associate to each network (point in time indicated by a triangle), acquire a DHCP lease (indicated by a square), and exchange information (indicated by a diamond). On average, each peer communication step requires 3 s, of which the information exchange consumes 1 s, resulting in a 16 s duration for the overall process. Assuming a pedestrian mobility of 1.5 m/s, a peer thus covers 22.5 m over this duration, inducing the risk of moving out of another peer's range, more so if both peers are mobile.

To provide a more sophisticated comparison, we evaluate the time overhead of *concurrent* 802.11 associations [10], [19], as natively supported by the ath9k driver. Fig. 10 shows the timings of concurrently associating to the peer networks (triangle), including the 1 s duration of a single 802.11 scan, and exchanging messages (diamond). We pre-set IP addresses in this evaluation, as the AP-driven DHCP process does not agree with client-controlled network switching [19]. While providing a speedup of roughly 33 % as communicating with all peers takes 10.7 s, concurrent 802.11 allows only partial parallelization of associations, as switching interfaces and networks induces a time overhead and missed (protocol) timeouts.

Last, CA-Fi affords instant and truly simultaneous information exchange (diamond) with all five peers within 1.7 s. In this, the main overhead is induced by the peers unsynchronized announcement frequency of 1/s. The 802.11-overhead-free, immediate exchange of information payload, combined with the low time overhead, thereby highlights the efficiency of CA-Fi in leveraging mobile ubiquitous communication contexts.

V. Related Work

Beacon Stuffing [11] overloads 802.11 beacon frames enable communication from APs to unassociated clients. It requires devices to operate in AP mode, assumes only 1-hop connectivity, and does not envision aggregation, forwarding, or flexible addressing in the presence of multiple applications. Although we share the observation that communication can happen without Wi-Fi associations, CA-Fi enables bi-directional multi-hop communication and addressing between arbitrary devices.

WiFi-Opp [20] addresses the rigid design and maintenance overhead of 802.11 networks by duty cycling both active phases and AP/client roles of devices. However, the overhead of network discovery and association as well as the restriction of communication to a single network remain. In contrast, CA-Fi provides a ubiquitous side-channel without discovery and association overhead, surpassing 802.11 network scopes.

MultiNet [10] mitigates the restriction of a single network association through virtualization of BSS or IBSS wireless network interfaces. However, each association restricts communication to the respective network, induces maintenance overhead, and requires prior identifier exchanges for discovery.

Among others, *eDiscovery* [21] proposes adaptive device discovery as a first step to ubiquitous communication in mobile contexts. In contrast to application-centric addressing in CA-Fi frames, the proposed discovery approaches do not incorporate such semantics. We aim to investigate the applicability and benefits of our approach to Bluetooth Low Energy, as motivated by the energy efficiency results in [21].

E-Smalltalker [22], among others, proactively broadcasts application information to enable use case-centric peer discovery. In contrast to our approach, such approaches do not establish a viable and general bi-directional communication channel.

VI. Conclusion

CA-Fi provides a truly spontaneous, broadcast, and ubiquitous communication channel, enabling association-less data exchange concurrent to bandwidth-intensive purpose-driven or incumbent 802.11 networking. Flexible addressing and message aggregation enable efficient co-existence of multiple applications for heterogeneous ubiquitous mobile communication. CA-Fi presents a tradeoff between preserving up to 70 % of 802.11 throughput and association-less data rates of up to 30 kB/s without 802.11 networking time overhead. Consuming 44 % less energy than associated devices when idle and comparable energy when sending, our evaluation on mobility traces shows the applicability and energy-efficiency of CA-Fi as a basis for purpose-driven 802.11 networking and as a self-contained communication channel. CA-Fi's concurrency, lack of 802.11 overhead, and instantaneity benefits mobile applications, as evaluated in comparison to iterative and concurrent 802.11.

Researchers may easily adopt CA-Fi through the integration of our publicly available prototype code [23] in 802.11 and Linux primitives. Future work targets better realization of duty cycling mechanisms, application of our approach in Bluetooth as motivated in [21], and real-life use case deployments that benefit from the increased communication and interaction scope.

Acknowledgements

This work was funded by the DFG Cluster of Excellence on Ultra High-Speed Mobile Information and Communication (UMIC) and the DFG SFB 1053 "MAKI".

References

[1] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble rap: social-based forwarding in delay tolerant networks," in *MobiHoc*, 2008.

[2] B. Han, P. Hui, V. A. Kumar, M. V. Marathe, G. Pei, and A. Srinivasan, "Cellular traffic offloading through opportunistic communications: a case study," in *CHANTS*, 2010.

[3] J. Ott, E. Hyytia, P. Lassila, T. Vaegs, and J. Kangasharju, "Floating content: Information sharing in urban areas," in *PerCom*, 2011.

[4] C. Shi, V. Lakafosis, M. H. Ammar, and E. W. Zegura, "Serendipity: enabling remote computing among intermittently connected mobile devices," in *MobiHoc*, 2012.

[5] M.-R. Ra, B. Liu, T. F. La Porta, and R. Govindan, "Medusa: a programming framework for crowd-sensing applications," in *MobiSys*, 2012.

[6] R. Friedman and N. Mori, "3dls: density-driven data location service for mobile ad-hoc networks," in *MobiHoc*, 2009.

[7] M. K. Han, A. Bhartia, L. Qiu, and E. Rozner, "O3: optimized overlay-based opportunistic routing," in *MobiHoc*, 2011.

[8] M. Bakht, M. Trower, and R. H. Kravets, "Searchlight: won't you be my neighbor?," in *MobiCom*, 2012.

[9] IEEE, "IEEE Standards." http://standards.ieee.org/.

[10] R. Chandra and P. Bahl, "Multinet: connecting to multiple ieee 802.11 networks using a single wireless card," in *INFOCOM*, 2004.

[11] R. Chandra, J. Padhye, L. Ravindranath, and A. Wolman, "Beacon-stuffing: Wi-fi without associations," in *HotMobile*, 2007.

[12] P. Meroni, S. Gaito, E. Pagani, and G. P. Rossi, "CRAWDAD trace set unimi/pmtr/txt." http://crawdad.cs.dartmouth.edu/unimi/pmtr/txt, 2008.

[13] T. King, T. Haenselmann, S. Kopf, and W. Effelsberg, "Overhearing the wireless interface for 802.11-based positioning systems," in *PerCom*, 2007.

[14] A.-K. Pietiläinen, E. Oliver, J. LeBrun, G. Varghese, and C. Diot, "Mobiclique: middleware for mobile social networking," in *WOSN*, 2009.

[15] A. Broder and M. Mitzenmacher, "Network Applications of Bloom Filters: A Survey," in *Internet Mathematics*, vol. 1, 2002.

[16] S.-Y. Ni, Y.-C. Tseng, Y.-S. Chen, and J.-P. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *MobiCom*, 1999.

[17] M. Salathé, M. Kazandjieva, J. W. Lee, P. Levis, M. W. Feldman, and J. H. Jones, "A high-resolution human contact network for infectious disease transmission," *PNAS*, 2010.

[18] L. Feeney and M. Nilsson, "Investigating the energy consumption of a wireless network interface in an ad hoc networking environment," in *INFOCOM*, 2001.

[19] H. Soroush, P. Gilbert, N. Banerjee, B. N. Levine, M. Corner, and L. Cox, "Concurrent wi-fi for mobile users: analysis and measurements," in *CoNEXT*, 2011.

[20] S. Trifunovic, B. Distl, D. Schatzmann, and F. Legendre, "Wifi-opp: ad-hoc-less opportunistic networking," in *CHANTS*, 2011.

[21] B. Han and A. Srinivasan, "ediscovery: Energy efficient device discovery for mobile opportunistic communications," in *ICNP*, 2012.

[22] Z. Yang, B. Zhang, J. Dai, A. C. Champion, D. Xuan, and D. Li, "E-smalltalker: A distributed mobile system for social networking in physical proximity," in *ICDCS*, 2010.

[23] "CA-Fi source code." https://github.com/flottetotte/CA-Fi.