Beacon-based Short Message Exchange in an Inner City Environment

Sascha Schnaufer, Stephan Kopf, Hendrik Lemelson and Wolfgang Effelsberg
Computer Science IV - University of Mannheim, Seminargebaeude A5
D-68159 Mannheim, Germany
{schnaufer, kopf, lemelson, effelsberg}@informatik.uni-mannheim.de

***Abstract*— In this paper, we proposes a method to utilize the huge number of IEEE 802.11 access points in an inner city environment to distribute information such as local sensor values, votes or short text messages. Due to the overlapping channels on IEEE 802.11b/g it is possible to receive packets from neighboring channels by overhearing. This fact allows every access point to monitor a part of the frequency band without switching the current radio channel. Our novel idea is to add user data to network layer beacon packets which every access point continuously send on a user-selected radio channel. This method allows transferring small size messages between access points or an access point and a mobile device without any reconfiguration like disabling the WLAN encryption.**

We conduct a measurement in the inner city of Mannheim, Germany to evaluate the typical density and distribution of access points in such an environment. We then use the measured data to calculate for every access point the expected reception probability on the street segments around. Based on these statistics, we appraise the reception probabilities of the access points among each other and perform a simulation study to estimate the size of the dissemination area that can be reached. Additionally, we analyze the required time to distribute the information to all reachable access points. Based on the measurement and simulation results, we can conclude that the estimated connectivity of an ad-hoc network based on access points is sufficient to disseminate information in the proximity.

I. Introduction

In the last years, wireless ad-hoc networks were a hot topic in the area of network research. The vision of many researchers was to create a kind of multi-hop ad-hoc Internet, or at least an ad-hoc extension of the existing Internet. Such an extension would allow users to utilize the same applications no matter whether they are in a wireless or wired environment. This perception is highly focused on the end-to-end communication of devices that is predominant in the Internet. However, we can observe that there exists no widely-used multi-hop ad-hoc network at the consumer market yet.

In this paper, we analyze the causes of the current situation of ad-hoc networks and discuss the possibility to establish an open ad-hoc network based on the huge number of existing IEEE 802.11 devices. The idea is that users in the same geographical area who do not know each other can cooperate by exchanging small sized data such as local sensor values, votes or short text messages. For this purpose we introduce a novel mechanism that we call *Beacon-based Short Message eXchange* (BSMX) that can be utilized to build such applications in a real-world environment. The basic concept of BSMX is to insert user data into the network layer packets of IEEE 802.11 devices. This is possible because the IEEE 802.11 standard allows manufacturer-specific components in beacon, probe request and probe response packets which we can utilize for our purposes. BSMX allows user space applications to send short messages via unencrypted broadcast to all other devices within radio range, and to receive such messages vice versa. An important fact for the acceptance of the device owners is that the BSMX mechanism utilizes the user-selected radio channel and does not change any other configuration parameters of the device. This behavior is a core design aspect of the BSMX system, because we do not want to restrict the main use case of the deployed devices.

The main contributions of this paper are the presentation of a novel mechanism to exchange small sized messaged between IEEE 802.11 devices without complex configuration and a comprehensive measurement and simulation study to determine the expected connectivity of already deployed IEEE 802.11 devices.

The remainder of this paper is structured as follows. The next section analyzes the causes why multi-hop ad-hoc communication is not widely-used in the consumer market yet. Section III describes the details of the BSMX system. The measurement that was performed to estimate the connectivity and link quality between deployed IEEE 802.11 devices is discussed in Section IV. Section V describes how the measured data is utilized to derive information for a network simulation. Furthermore, this section explains the simulation setup and discusses the results. Related work in the field of information dissemination systems is discussed in Section VI. Finally, Section VII summarizes the paper.

II. Ad-Hoc Wireless Communication in the Real World

Several mainstream wireless communication technologies are available on the market today. On the one hand, there are technologies like the Universal Mobile Telecommunications System (UMTS) and the Global System for Mobile Communications (GSM), both with an area-wide infrastructure which is controlled by telecommunication companies who offer the access to the phone network and to the internet via their infrastructure as a commercial product. On the other hand, there are a huge number of IEEE 802.11 (Wireless LAN) and IEEE 802.15.1 (Bluetooth) devices most of them used in private households. In case of UMTS, the telecommunication companies have no interest that devices of their costumers communicate directly with each other without the usage of the companies' infrastructure and control. The situation of the second group of devices differs clearly, because there is no homogeneously area-wide infrastructure that is controlled by few companies or institutions. Most Bluetooth devices in this group are class 2 or class 3 devices and have a very limited radio range. In addition, the Bluetooth stack of these devices usually does not implement the required profiles to form Bluetooth ad-hoc networks which are called piconets.

Therefore, we focus the further discussion on IEEE 802.11 devices that have a higher radio range and support configuration tools to build an ad-hoc or mesh network. So the fascinating question that we want to discuss in this section is: why do we see so many deployed WLAN devices but no multi-hop ad-hoc network?

The huge number of deployed IEEE 802.11 devices can be separated in two groups: one group contains all infrastructure devices like access points and the other group consists of client devices like laptops or smartphones which often operate mobile with a limited power capacity. From the technical point of view, all of these devices can operate in access point mode, client mode or ad-hoc mode independent of their intended use. For instance, it is possible to configure the WLAN device of an android based smartphone to run as access point that allows using the Internet connection of the phone. The limitation is only which WLAN modes are implemented in the device driver and which of them are accessible from the user interface.

It is possible to combine several access points to a Wireless Distribution System (WDS) that allows network access in a larger environment. However, the most deployed access points in the private households are configured individually by their owners and are used to grand Internet access to the owners' devices. Although supported by IEEE 802.11, the ad-hoc mode that allows the direct communication from one device to another without using access points is rarely used in practice. We assume that there are three main reasons why access points and mobile devices do not form a multi-hop ad-hoc network:

1) Configuration: If an IEEE 802.11 client device is not connected to a network, it typically starts a search over all radio channels to find access points and ad-hoc networks in radio range. If the user wants to join a network he/she can adopt the required network properties like SSID and radio channel from the detected network. If an encryption like WEP or WPA is used every node in the network requires a consistent configuration. The communication is typically done via IP which requires a mechanism to assign a valid IP address. This makes the configuration very complex, and there is no default ad-hoc network configuration which is used while a device is not connected a network. Furthermore, not all device drivers can be connected to an access point for Internet access and communicate at the same time with an ad-hoc network. We are confident that most of these configuration problems are solvable with the adoption and extension of existing technologies, but the tools that the devices provide today are far too complex.

2) Scalability: If several WLAN devices are configured in ad-hoc mode and have a unique IP address, each device can communicate over IP with every other device in radio range. However, for multi-hop communication a special routing protocol is required that searches and maintains routes to nodes which are not in radio range. It is common practice to evaluate the scalability of such routing protocols via simulation. But it is necessary to keep in mind that the scalability concept in these evaluations is typically focused on the protocol itself. Scalable is defined in this context that the protocol overhead for every additional user is constant or increases only in a logarithmic way independent of the overall number of users. However, this property does not imply that enough bandwidth is available for every user. Let us assume that W is the maximum bandwidth of the shared medium with a constant radio range. Furthermore, assume n is the number of nodes in the multi-hop ad-hoc network. Then the expected bandwidth available from every node to another randomly selected node is $\Theta(W/\sqrt{n \log n})$ [1]. It follows that the basic network architecture does not scale if every node wants to communicate at the same time with another randomly selected node via an end-to-end routing protocol. In other words, it is necessary to limit the maximum number of hops depending on the available channel bandwidth and the required bandwidth per user.

3) Applicability: Every application based on a client-server architecture concentrates the incurred traffic at the server. If the ad-hoc network is not connected to the Internet the server is located inside the ad-hoc network and the available bandwidth in the proximity of the server is exhausted very fast. It is obvious that a better load balancing is possible if the ad-hoc network is connected to the Internet at several places and the server is placed inside the Internet. In this case the server is completely unrelated to the ad-hoc network. Another problem is that it is risky to share the own Internet access with other unknown people because they can misuse it for illegal purposes. Furthermore, the low-priced flat rates for mobile Internet access via UMTS allow users to access existing servers and services over the cell phone infrastructure. This makes it possible that two wireless devices with access to the cell phone infrastructure can communicate with each other through the Internet without multi-hop wireless routing. For these reasons, we conclude that multi-hop ad-hoc communication is not essential for most of the available IP-based applications.

We assume that wireless multi-hop ad-hoc communication finds its way to the customer market only if it provides additional advantages compared to end-to-end communication or Internet access. Furthermore, the configuration has to be very simple, and the primary usage of the mobile devices, typically the Internet access, must not be impaired. It is also necessary to ensure the privacy of the device owner what implies that other users have no access to the Internet over the owners' device or access to the local network. Hence, we have to accept that no common SSIDs and radio channels, no encryption at the data link layer and no IP protocol are available in the ad-hoc part of the network. Our BSMX system avoids all these problems and enables to utilize ad-hoc networks. This creates additional user benefit by enabling new types of information distribution applications. The general idea is to provide a common mechanism that allows users to share some small pieces of information with other persons in their local environment without establishing an explicit connection to them. The most important aspect here is that these applications do not address a specific device. Instead, the information is sent via broadcast to all other devices in radio range. The receiver devices can process the content and decide what to do.

A decentralized friend localization application is a simple example for such an application. Assume you have a date with a friend on a populated square and you are not sure whether your friend is around. The friend localization application allows you to see if your friend has arrived, in addition you can see if other friends are around. Several applications are available on the Internet which show the place where a friend currently is and support features like chat or speech communication. However, an ad-hoc friend localization application has some advantages; it can easily detect the spatial closeness without GPS or Internet connection, it has a "build-in" area of interest, and it does not store the location on a

company server with commercial interests. In the next section we present our novel technical solution to implement such applications.

III. Beacon-based Short Message Exchange

We propose to utilize the huge number of access points available in most big cities as a distributed heterogeneous infrastructure for a best-effort information distribution system. Access points and ad-hoc network nodes send periodically, typically ever 100 ms, unencrypted beacon packets. The IEEE 802.11 standard allows manufacturer-specific components in the beacon packets which are called tagged parameters. We have developed a small IEEE 802.11-compliant extension that allows user space applications like a protocol daemon to add short messages to these network layer beacons, or to send additional unencrypted network layer frames. Furthermore, it is necessary to forward such additional user data to the related application. A major advantage of our proposed approach is that it does not require a common SSID or routing layer configuration. The idea is that the access points and ad-hoc devices can operate unmodified without changing their network and security configurations. Mobile devices with limited battery capacity usually do not operate with a continuously activated WLAN device. Furthermore, such devices typically run in client mode and hence they do not send continuous beacon packets. Nevertheless, these devices can inject and receive information by starting a search for access points. In case of a search for networks, a device sends so called probe request packets that are answered form receiving access points by returning a probe response packet. This mechanism is used to enable a fast discovery of access points in the proximity and is called active scan. Both the probe request and the probe response packets contain tagged parameters and can be extended to exchange user data. This way a mobile device can exchange information on demand and extend the inter access point connectivity. We call this novel communication method *Beacon-based Short Message eXchange* (BSMX).

The described exchange methods lead to a heterogeneous radio channel configuration. However, the distance between the IEEE 802.11b/g channels is only 5 MHz, and the used channel width is 22 MHz. This overlap and the used encoding technique allow the successful decoding of some packets which are transmitted on the neighboring channels by using a technique called overhearing. Network devices drop such packets by default, but our BSMX system uses this overlap of the channels to monitor a part of the frequency band without changing the radio channel of the device. The remaining channels and the non-overlapping IEEE 802.11a channels can be monitored with temporal channel changes only. In Section IV, we analyze the described overhearing effect and estimate the expected packet reception rates on neighboring channels if no channel modifications are allowed.

We realize a proof of concept implementation of the described IEEE 802.11-compliant extension by modifying the open source driver MadWifi which supports Atheros based chipsets. Figure 1 shows a part of a captured beacon packet that was sent by an access point that runs with the modified MadWifi driver. The first part of the packet that contains the packet type and source address is cut out. The top of the screenshot shows the so called fixed parameter like timestamp and beacon interval. The following tagged parameters are stored as a sequential list of elements, each with tag number, tag length and the related data. At the end of the list, an additionally BSMX element with the tag number 222, a length of 19 byte and a text message that is highlighted at the bottom of the screenshot is added. In addition to the MadWifi driver that can also operate in client mode, we currently work on an implementation for the HTC Hero which uses the Android operating system.

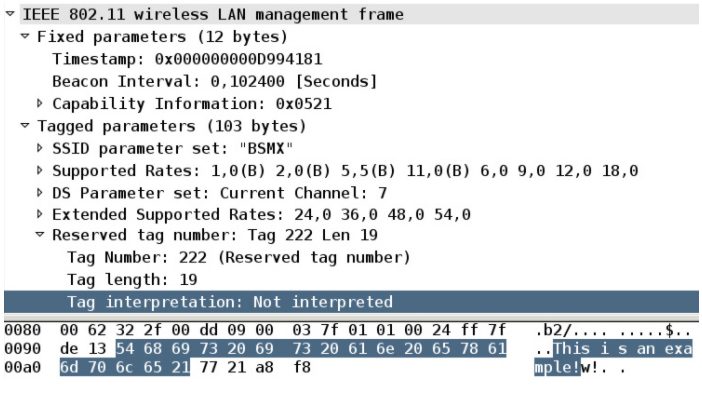


Fig. 1. Captured beacon packet that contains an additional tagged parameter with a text message.

Our BSMX system enables the implementation of new protocols that can send and receive small messages via WLAN without complex configuration. We propose to use a new protocol for this purpose which we call *Information Distribution Protocol* (IDP). This situation is comparable to the Internet Protocol in the routing layer. On an open operating system it is possible to install any kind of routing layer, but in reality most devices operate with IP only. As a result of this situation, most applications presume a TCP/IP environment by utilizing features of this protocol family. IDP should offer a common set of features that developer can utilize to build applications like the distributed friend localization application described above. Currently we have designed IDP and are in an early implementation phase and give an overview of the operating principles.

At the beginning IDP creates a local knowledge base and inserts its currently known information. Every element in the knowledge base stores the information itself, its expiring date an information type ID. The information type ID specifies the used aggregation and forwarding mechanisms, e.g., several temperature sensor values can be aggregated to the average temperature in an environment. For every device, IDP periodically adds a subset of its knowledge to the network layer beacon that is send via one-hop broadcast to all neighbors within radio range. The user can define how much of the available bandwidth he wants to use for the distribution while the device is idle and how much bandwidth can be used while the device is busy. If the allowed bandwidth is not completely utilized and there is enough information to send, IDP can send additional unencrypted frames via broadcast. An application can injected new information into the distributed system by utilizing the IDP daemon of the local device. Every node listens on the user configured radio channel, and if a beacon or probe request is received the information is extracted and forwarded to the IDP daemon. If the received information matches the stored user interests, IDP adds it to the local knowledge base or aggregates it with existing information, respectively. The IDP daemon also provides an interface that can be utilized by applications to subscribe the reception of incoming information of a specific type. Optionally, an idle system can temporally change the channel and send some frames or listen for a certain time for

incoming beacons.

The major challenges should be handled by IDP: First, the system needs an aggregation mechanism. In this context, aggregation means the combination of atomic data to a data structure that contains all *relevant* information. For instance all votes in a distributed poll can be combined into the poll result, or the data of different temperature sensors can be combined into the average temperature of an area. In the latter example, the average temperature itself is a floating point number; it does not contain any information about which atomic data is included. Therefore, it is necessary to add additional data that enables further processing, e.g. how to combine several aggregated temperature values. For this purpose, the IDP implementation requires a set of suitable data structures. One design option here is to use probabilistic data structures which allow to store the required additional data in a very compact format. However, this method has the disadvantage that it is only possible to calculate an approximation of operations like average, sum or count. The developer has the choice to select the preferred compromise between size and accuracy. Flajolet-Martin sketches [2] are a well-known example for such a data structure. We briefly discuss several suitable approaches and applications in the related work section.

The second challenge is to develop a selection mechanism that decides which and how often information is sent. This includes an approach to limit the geographical dissemination range of a message with respect to the age, the user relevance and the network load. The distribution area depends also on the aggregation ability of the regarded information and is limited in size, e.g., it might be possible to estimate the result of the distributed poll on adjacent streets but not of the whole city. The idea is to choose the level of aggregation based on the distance from the point of origin and to stop the distribution when we get too far away. The system can also distribute unspecified data like text messages that can not be aggregated. Depending on the current network load the distribution area for such messages might be very limited. However, this is sufficient to develop applications like a public digital pin board that allows residents to exchange small messages such as ”we arrange a party today in apartment X, everybody welcome”. The selection mechanism also considers how often the information is distributed from neighboring nodes. Furthermore, outdated entries are removed periodically.

The third challenge is the implementation of a component that continuously measures the current radio channel utilization to decide how many packets can be send with respect to the user-selected bandwidth limits. If the available bandwidth is exhausted, the distribution works slowly and the distribution area is very limited. However, the main use case of the device is usually not affected. Furthermore, such a mechanism can use idle times to switch the current radio channel temporarily.

After this short description of the concept of the BSMX system and the basic functionality of IDP, the purpose of the rest of this paper is to evaluate the typical radio characteristics in an inner city environment and to use this knowledge to estimate the expected connectivity of deployed IEEE 802.11 devices by a simulation study.

IV. Measurement setup and results

The first step to prove the feasibility of our approach is to determine the typical properties of IEEE 802.11 devices in an inner city environment. These properties are amongst others the device density, the radio channels in use and the packet reception rate on the street. Typically the WLAN module of mobile devices is not continuously active, and furthermore, the density of these devices highly depends on the time of day and the number of pedestrians on the streets. For this reasons the measurement is focused on the huge number of deployed and permanent active access points. By sending a probe request packet via broadcast an active scan can discover the existence of an access point, its basic setup and the signal strength, but it is insufficient to measure the packet reception rate. Therefore, we chose a completely passive method and equipped the roof of a car with five embedded devices that are listen on different channels, namely two, five, eight and eleven. In addition to these five devices we added a sixth device which continuously switches between the channels and is also used as a control device. Every device operates in monitor mode that allows the reception of all WLAN frames independently of the send channel, SSID or network layer. This way, it is possible to capture management frames like beacons which normally can not be received from user space. The five devices with the fixed radio channels allow - with respect to the channel overhearing phenomenon - the monitoring of most of the useable frequency spectrum. One device was additionally equipped with GPS and stores the current position each second. We drove with this setup along every street in the inner city of Mannheim and recorded every frame received. Mannheim is unusual among German cities in that its central area is laid out in a grid pattern (see Figure 2). Every road in this grid is a one-way street, and some are accessible by pedestrians only. Anyhow, it is possible to drive through enough streets by car to determine the desired network characteristics.

TABLE I
THE MOST COMMONLY USED SSIDS

SSID	NUMBER	PERCENTAGE
FRITZ!Box Fon WLAN 71XX	374	9.85%
WLAN	90	2.37%
o2DSL	80	2.11%
DSLWLANModem200	78	2.05%
default	67	1.76%

After the measurement we have stored all the relevant fields of the captured packets in a database for further processing. That makes it possible to identify all discovered networks by selecting all unique combinations of BSSID, SSID and radio channel. We received beacons from 3797 different networks in an area of approximately $1.42km^2$, including some unreachable pedestrian streets. The most commonly used radio channels are one, six and eleven, but all other radio channels are chosen by at least 200 access points. Table I shows the most commonly used SSIDs in our area. The first table entry combines the default names of three different device variants from the device manufacture AVM which is widely used in Germany. The other entries are default values from other manufactures or Internet service providers.

The channel hopping mechanism that was used by the control device was able to discover nearly all networks. The devices with fixed channels detected approximately 90-100% of networks which sent on a neighboring channel, and 60-80% of networks which sent on a channel that was two channels away. Keep in mind that network detection only implies that at least one packet from the network was received on the considered channel.

From an application point of view the unicast communication

over a WLAN connection with good radio signals typically works very well and no packet retransmissions at application layer is necessary. However, even in situations with good preconditions many packet collisions and packet losses occur that require retransmissions on the link layer (hidden from the higher layers). All broadcast packets like beacons are not acknowledged and not secured by retransmission on the link layer. Thus, the packet reception probability is an important characteristic value to estimate the expected connectivity in an inner city environment. With respect to the heterogeneous device configuration there is no common radio channel, hence the reception probabilities on the neighboring radio channels are also highly relevant. The computation of the reception probability per channel is comparatively complex because it is necessary to find a method to estimate the number of packets that an access point sends without receiving them. To approximate these probabilities we determine which device can receive which access points at every logged GPS position. The logged GPS positions are sequentially ordered and correspond to the driven route. It is now possible to determine for every combination of access point and device at which successive positions it was feasible to receive beacons from the considered access point on the according radio channel. Such a list of successive positions has also a temporal dimension what is the reason why we call it communication period. The temporal dimension of a communication period is the time between the first and the last received beacon of the considered access point at these positions. Caused by the driven route and the radio propagation, many access points have a set of several communication periods, e.g., one for each side street and one for every visit. The beacon send interval of an access point – that is sent with every beacon packet – can now be used to estimate the number of packets sent during a communication period. This way we can calculate the percentage of received beacons on a specific device and radio channel for every access point.

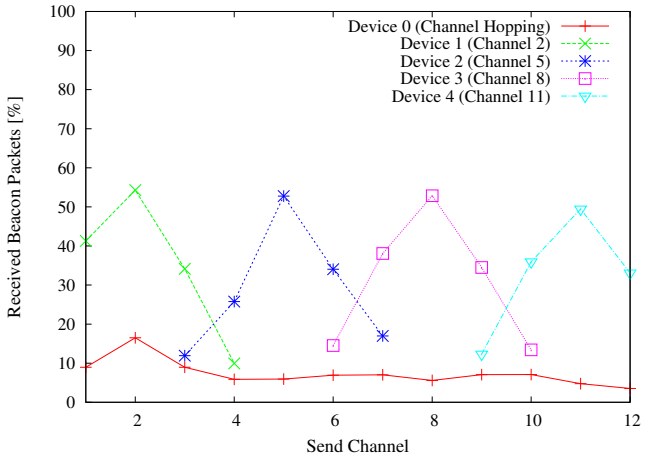


Fig. 2. Measured packet reception rates on streets in the inner city of Mannheim

Figure 2 shows for every capturing device the average packet reception rate over all access points that send beacons on the considered radio channel. The average reception rate of all static channel devices is 53% for packets that are sent on the same channel, 35% for packets that are sent on neighboring channels and 14% for packets that are sent on channels that are two channels away. The low rate of 53% is caused by the fact that most access points should only cover indoor areas, whereas the measurement was performed on the street. The capturing device that continuously changes the radio channel has higher reception rates on the first three channels compared to the other channels. This behavior is caused by the channel hopping mechanism which stays longer on the first channels.

It is conceivable that beacons are sent by access points on neighboring channels that the capturing device does not discover and thus the real reception rate on a neighboring channel is maybe lower. Therefore, we verify the presented results by using the following method: First, we select all received beacons from the channel-hopping device that are sent on channels that are next to the reception channels of the other capturing devices. Then, we check how many of these packets that are send on channel x are received on the devices that listen on channels x-1 or x+1. This way, we calculate that on average 63% of these packets are also received by a static channel device. If we now assume that the channel-hopping device itself receives 53% of the packets sent on the current channel, we can conclude that $53\% \cdot 63\% = 33\%$ of these packets are received on the device that captures the neighboring channel. This result confirms the results from the communication period analysis.

In the next section, we describe how we utilized the measured results to build a simulation environment to analyze the inter access point connectivity and the anticipated information dissemination of BSMX messages.

V. Simulation of a BSMX System

In the first subsection we describe how we use the measured data to form a simulation environment. The setup of the simulation is discussed in the following, and the results of the simulation are presented last.

A. Simulation Environment

Classic propagation models like free space or two-ray ground assume that all devices run on the same radio frequency, and thus they are not suitable for the simulation of a heterogeneous network that uses overhearing intensely. Therefore, we use a probabilistic method that is based on our measurement to decide which devices can communicate with each other.

We have introduced the concept of communication periods above that describe a spatially and temporally cohesive list of GPS positions at which it was possible to receive packets from the related access point. Based on this, we calculates the average packet reception rate for every radio channel for every such communication period. This can be done by searching the device whose fixed radio channel was the nearest to the channel on which the access point sent. The radio channel of the selected device is with respect to the measurement setup identical with the sending channel or at most a direct neighboring channel. In the former case it is possible to use the measured reception rate of the device directly, and in the latter case the measured reception rate has to be modified by multiplying it with a correction factor. Based on our measurements we estimated that the reception rate on the same channel is 53% and on a neighboring channel 35%. This proportion can be used to estimate the required correction factor $c = 0.35/0.53 = 0.66$. The described procedure can be used to evaluate the average reception

Fig. 3. Network construction example

rate on each radio channel on every street segment that was visited. Based on these statistics, can we decide with which probability which access points can be found by a mobile IEEE 802.11 client device that scans at the related position.

However, this strategy can not be used directly to estimate which access points can communicate with each other. The first step in order to solve this problem is to estimate the position of the considered devices. Unfortunately, it is not possible to estimate the exact position of an indoor access point automatically by driving by. If data from an access point is received on one street only, it is not even possible to decide on which street side the device is located. But an approximate position can be estimated by calculating the centroid of the area where the observed network was received. In the observed inner city case the reception area is equivalent to the geometric union of all routes that are related to the communication periods of the observed access point. The centroid of this geometric union is typically inside a building what is obviously a plausible position for an access point. However, we have no reception statistics for these positions, and without an indoor measurement in all houses of the inner city it is still not possible to estimate accurate reception rates. Therefore, we search the point inside the considered geometric union that have the smallest distance to the related centroid and regard this point as estimated position of the access point. This trick allows us to use the data that was measured at this position and the resultant reception probabilities for each channel.

We exported the measured data to Google Earth for visualization purposes. Figure 3 shows a screenshot of Google Earth at a very high zoom level. Every pin represents the estimated position of a discovered access point; it is labeled with the network number and the used radio channel in brackets. The access point with number 510 located in the middle of the picture operates on radio channel six. All street segments in which that access point can be received are drawn as a solid red line. All other access point whose estimated position lays on this solid line and which operate on a suited radio channel can receive packets from the access point with number 510. In this example, the access point 4231 (channel 9) has a reception probability of 6% and the access point 3673 (channel 6) 12%, respectively. The green solid line between network 510 and 3673 indicates that there is a bidirectional virtual link which means that it is also possible to receive packets from 3673 at the position of 510.

The simulation network is generated by the following steps:

- Estimate for every access point x the geometrical union F_x of all street segments on which it is possible to receive packets from the access point. These street segments are the spatial dimension of the related communication periods.
- Calculate for every network x the centroid $c(x)$ of the geometrical figure F_x from the step before.
- Select the point $p(x) \in F_x$ that minimizes $d(p(x), c(x))$ as estimated position of the access point x.
- Create for every access point x with $F_x \cap p(y) \neq \emptyset$ an outgoing network link $x \rightarrow y$.
- Select for every link $x \rightarrow y$ the measurement device d_z whose channel is the nearest to the sending channel of x. Calculate the average reception rate of packets from access point x that device d_z has received at position $p(y)$. Store the calculated reception rate as the reception probability of the link $x \rightarrow y$. If the sending channel and the device channel are not equal, the channel of d_z is a neighboring channel with respect to the measurement setup. In this case correct the calculated reception probability by multiplying it with the correction factor $c = 0.66$ introduced above.

Fig. 4. The access points in the inner city of Mannheim and the information injection zone (in white)

The generated network consists of 3797 nodes and 27135 links. 74% of these links are unidirectional. 970 nodes have incoming links only and 56 outgoing links only.

B. Simulation Setup and Procedure

In this subsection, we describe the simulation setup and the detailed procedure. The purpose of this simulation is to prove the feasibility of our BSMX approach by evaluating the dissemination process of injected messages that highly depend on the inter access point connectivity.

Nodes inside the border area of the simulated network have other connectivity characteristics than nodes at the center. Therefore, we define an area in the middle of the network that we call injection zone. Only nodes inside this zone can inject a message and start the dissemination this way. Figure 4 shows the inner city of Mannheim with all detected access points and the injection zone in the middle of the network.

We assume that every access point continuously sends beacon packets at regular intervals. In our measurement we determined that

97% of the detected access points use the default delay of 100 ms. However, in reality the beacon sending times of different access points are not synchronized. This means that the time between receiving a beacon from a neighboring access point and the next own transmission is on the average half of the sending interval. In the simulation we simplify this by defining that every access point in the network sends one beacon packet per simulation round. This is equivalent to the worst-case delay in a real-world environment, but allows a simulation without specifying a beacon sending interval. The number of simulation rounds can be converted to a runtime equivalent by multiplying it with the desired beacon send delay.

At the beginning of the simulation the information store of every node is empty. Then one access point in the injection zone is selected as start node that begins the dissemination. In the first round the start node sends the information by broadcast to all neighboring nodes. The broadcast is simulated by sending the packet on all outgoing links of the sender node. The simulator utilizes the previously calculated reception probabilities to decide if a packet is successfully received on the regarded link. Every node that has successfully received the information stores it in the information store and supports the dissemination by sending it out over beacons in the following rounds. If all reachable nodes have received the information, the simulation ends. Please consider, that node b is reachable by node a if a path from node a to node b exist and thus the set of reachable nodes depends on the start node and has no fixed size.

In the network generation section we have assumed that every existing device participates in our BSMX system. However, the simulator can either use the complete network or it can randomly filter out a subset of access points by deleting the nodes and all related links. In this way, we simulate different participation rates from 5% - 100% in steps of five percent. We chose per start node and participation rate 100 different sets of participators and simulate each set five times. The injection zone contains 264 access points that can start the dissemination. Hence, we run for each of those access points the simulation $100 \cdot 5 = 500$ times with different seeds and aggregate the results.

C. Simulation Results

The purpose of our simulation is to determine which distribution inside the network can be reached by an injecting node on the average, and how long the dissemination requires. We differentiate the dissemination progress between how many nodes are reached and the spatial coverage of the dissemination area. The latter can be determined for a given set of nodes A by calculating the geometric union of all street segments on which it was possible to receive at least one node $x \in A$. This means that the spatial coverage is the combination of one-dimensional segments, and is not necessarily connected. If we consider the set of nodes reached, then every mobile device on a street segment inside the spatial coverage can receive the information by scanning the radio spectrum.

Figure 5(a) shows the percentage of informed nodes and the required simulation rounds for different participation rates. The percentage rate of informed nodes is related to the total number of nodes in the simulation setup. If all nodes participate, an injecting node inside the start zone can reach on the average 72% of all other nodes in the network. However, if only 55% of all nodes participate, the complete network consists of 2088 nodes of which 28% on the average can be reached from the start zone.

Aside from the number of reached nodes, the required time is an important aspect. Each node sends one beacon per simulation round. That is the worst case compared to randomly selected sending times in the real world. The simulation time can now be determined by multiplying the round numbers with the preferred sending delay. Please consider that we have limited the maximum number of simulation rounds in the figure to achieve more clearness. With all nodes participating, the network needs 20 rounds on the average to inform 40% of all nodes. If we assume the default beacon delay of 100 ms, the 20 rounds require two seconds only. The dissemination to 40% of all nodes in a network with a participating rate of 70% require on the average 41 rounds what corresponds to 4.1 seconds.

If we consider also mobile client devices like smartphones, handhelds and laptops, the spatial coverage is another important aspect. Figure 5(b) shows the percentage of street segments on which an injected information can be received by channel scanning. In this figure, the coverage percentage is related to all street segments that we have visited. The simulation results show clearly that the maximum spatial coverage per setup is reached significantly faster than the related maximum number of nodes, e.g., the 70% participation setup reach 40% of the street segments in 27 rounds compared to the 41 rounds discussed above. Another interesting result is that every coverage percentage is in total higher than the related reachable node percentage.

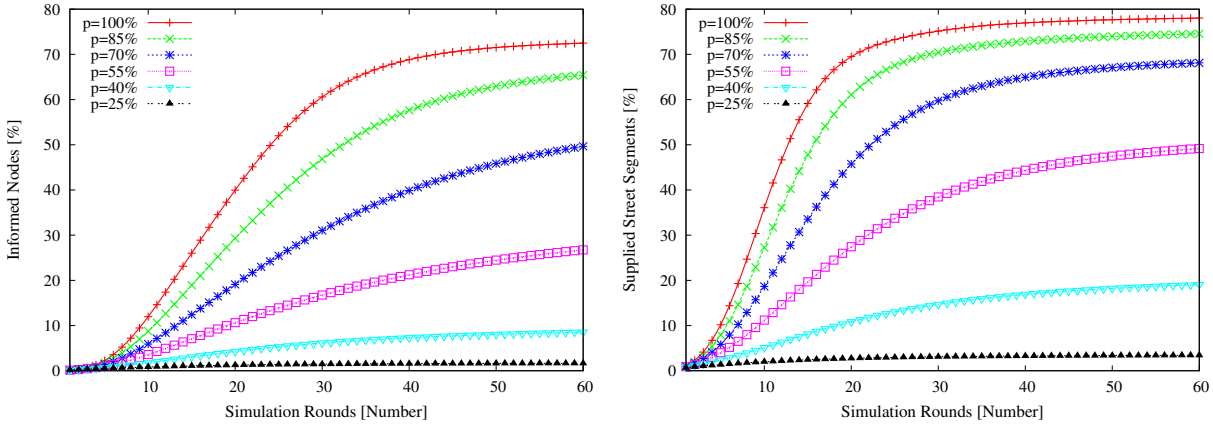
The average reachable dissemination is mainly limited by the network connectivity which can be increased by implementing a temporal channel switching algorithm. Furthermore, mobile devices can also overcome connectivity problems by sending previously received information to unconnected access points. Based on these results, we conclude that the estimated connectivity between access points in an inner city environment is sufficient to deploy a best-effort BSMX system.

VI. Related Work

A sensor network consists of spatially distributed microcomputers with very limited memory and battery capacity. These computers are equipped with a radio system and several sensors, and typically observe an area of interest. In most cases, the network is connected to the outside world by one node only that is called data sink. An outside device can access the sensor values by sending a query to the data sink. A large number of methods was developed to gather the required data and to respond to the query e.g., Directed Diffusion [3] or COUGAR [4]. The requested data can also be aggregated in a distributed fashion with approaches like Tiny AGgregation [5] or TiNA [6]. All these approaches build a spanning tree with the data sink as source and aggregate the data along the tree.

Contrary to the described methods, our approach does not assume a consistent configured network that was deployed for one specific purpose. Furthermore, there is no data sink or query mechanism. A device can receive information only that is forwarded by a neighboring access point or mobile device without any multi-hop request method. This way, every device can decide how much bandwidth it wants to consume by sending additional data in the network layer packets.

Techniques for the dissemination and aggregation of information were also proposed for *Vehicular Ad-Hoc Networks* (VANET).



(a) Informed Nodes

(b) Street Segments reached

Fig. 5. Progress of the information dissemination in an inner city environment

SOTIS [7] and TrafficView [8] exchange information like speed and position among vehicles in order to enable users to access the current traffic conditions. Another decentralized approach to find free parking places is presented in [9]. Similar to our approach, these concepts exchange information by sending broadcast messages periodically. However, the main purpose of a VANET is to increase the security by warning other cars of dangerous situations like an emergency braking. Due to the high risk of misuse, such a network does not use an open architecture that can be utilized to develop novel applications by everyone. Furthermore, all participating vehicles have a standardized setup and configuration. The contribution of this paper is the novel idea and technical solution to build such systems based on already deployed IEEE 802.11 devices that have primarily another intended use. Anyhow, we can adapt relevant ideas from this research area for the further development of the drafted Information Distribution Protocol, e.g., the analysis of probabilistic data structures for the aggregation in a VANET environment [10].

VII. Conclusion

Based on the results of our measurement und simulation study, we conclude that the estimated connectivity between access points in an inner city environment is sufficient to deploy a best-effort BSMX system. With the introduced IDP and BSMX system, we have achieved our goal to specify a mechanism that can be utilized to create novel applications without complex device configuration or significant impairment of its main functions. Furthermore, this mechanism can be implemented in an IEEE 802.11-compliant way and can be adapted to previously deployed devices. We believe that many users that have bought an equipped device or obtained the functionality by a firmware update will give such a system a break. Most important is the fact that the system avoids any direct or indirect share of the user's Internet connection. Another interesting aspect is that the system provides user benefit also in areas with a spare participation rates. In these areas it is not guaranteed that all access points can communicate with each other, but it is still possible to run simple applications such as a wireless pin board to publish messages inside an apartment building or a hotel facility. Furthermore, the simple exchange like contact details, public IP addresses or web links between devices can generate a benefit for users.

As future work, we will improve the implementation of our prototype system to make it available for access points, laptops and Android based smartphones. Therefore, we want to analyze and simulate the aggregation procedure in more detail. We also want to adapt and extend existing approaches from the sensor network and VANET area. Our goal is to provide a common interface for the distributed calculation of operations like average, sum or count that can be utilized by application developers. Furthermore, we want to implement an algorithm that temporary switches the radio channel while the device is idle to overcome connectivity problems in areas with spare participation rates.

References

[1] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, March 2000.

[2] P. Flajolet and G. N. Martin, "Probabilistic counting algorithms for data base applications," *J. Comput. Syst. Sci.*, vol. 31, no. 2, pp. 182–209, 1985.

[3] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: a scalable and robust communication paradigm for sensor networks," in *MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking*. New York, NY, USA: ACM, 2000, pp. 56–67.

[4] Y. Yao and J. Gehrke, "The cougar approach to in-network query processing in sensor networks," *SIGMOD Rec.*, vol. 31, no. 3, pp. 9–18, 2002.

[5] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "Tag: a tiny aggregation service for ad-hoc sensor networks," *SIGOPS Oper. Syst. Rev.*, vol. 36, no. SI, pp. 131–146, 2002.

[6] M. A. Sharaf, J. Beaver, A. Labrinidis, and P. K. Chrysanthis, "Tina: a scheme for temporal coherency-aware in-network aggregation," in *MobiDe '03: Proceedings of the 3rd ACM international workshop on Data engineering for wireless and mobile access*. New York, NY, USA: ACM, 2003, pp. 69–76.

[7] L. Wischhof, A. Ebner, H. Rohling, M. Lott, and R. Halfmann, "SOTIS - A Self-Organizing Traffic Information System," in *Proceedings of the IEEE 57th Vehicular Technology Conference (VTC 2003 Spring)*, Jeju, Korea, April 2003.

[8] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "Trafficview: traffic data dissemination using car-to-car communication," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 8, no. 3, pp. 6–19, 2004.

[9] M. Caliskan, D. Graupner, and M. Mauve, "Decentralized discovery of free parking places," in *VANET '06: Proceedings of the 3rd international workshop on Vehicular ad hoc networks*. New York, NY, USA: ACM, 2006, pp. 30–39.

[10] C. Lochert, B. Scheuermann, and M. Mauve, "Probabilistic aggregation for data dissemination in vanets," in *VANET '07: Proceedings of the fourth ACM international workshop on Vehicular ad hoc networks*. New York, NY, USA: ACM, 2007, pp. 1–8.