

# Pocket Switched Networking: Challenges, Feasibility and Implementation Issues

Pan Hui<sup>1</sup>, Augustin Chaintreau<sup>2</sup>, Richard Gass<sup>2</sup>,  
James Scott<sup>2</sup>, Jon Crowcroft<sup>1</sup>, and Christophe Diot<sup>2</sup>

Cambridge University<sup>1</sup>, Intel Research<sup>2</sup>  
{pan.hui, jon.crowcroft}@cl.cam.ac.uk,  
{augustin.chaintreau, richard.gass,  
james.w.scott, christophe.diot}@intel.com

**Abstract.** The Internet is built around the assumption of contemporaneous end-to-end connectivity. This is at odds with what typically happens in mobile networking, where mobile devices move between islands of connectivity, having opportunity to transmit packets through their wireless interface or simply carrying the data toward a connectivity island. We propose *Pocket Switched Networking*, a communication paradigm which reflects the reality faced by the mobile user. Pocket Networking falls under DTN. We describe the challenges that this approach entails and provide evidence that it is feasible with today's technology.

## 1 Introduction

Mobile networking is finally becoming ubiquitously deployed, due in large part to the convergence of mobile telephony and handheld computing. Current mobile devices typically have one or more wireless interfaces (e.g. Bluetooth, WiFi). The applications which are commonly deployed on such devices (e.g. email, web browsing), however, are rarely able to fully exploit this local wireless connectivity, and instead use it only as a means of acquiring global connectivity via access points.

Therefore, there is currently a large amount of wireless bandwidth capacity that remains unused because the current communication paradigm (i.e. the Internet) has not been designed to take advantage of local and intermittent connectivity. The underlying reason for this failure is that *IP-centric networking* (a term covering everything from the IP network layer through to application-layer protocols such as HTTP) relies on several assumptions which do not hold for mobile users. One such assumption is that the source and recipient of a datagram are *contemporaneously connected*, i.e. that throughout a communication there exists a complete path between the two parties communicating. Another assumption, based on the end-to-end argument, is that it is sensible to determine the precise endpoints of a connection before any application data is transferred, and to have intermediate nodes in the network simply perform best-effort routing.

We propose a new set of assumptions for mobile networking. We argue that these assumptions lead to a new networking model, which we term *Pocket Switched Networking* (or PSN) since it relies on both occasional transmission opportunities and user mobility to carry data to their destination. These assumptions are as follows. Mobile networking users carry one or more devices having significant storage capacity. Their mobility may be useful as a data-carrying mechanism. Devices have local networking interfaces, with which they can exchange data with neighbors. Devices may have access to one or more global networks (e.g., Internet, GSM), which differ in price, bandwidth, and availability. Both global and local connections may provide *opportunities* to transfer data.

We identify two classes of communications that users demand. Local communication allows wireless devices to use their communication infrastructure to provide communication services in the absence of end-to-end infrastructure. Local services are currently not provided by the Internet. Examples are prevention of natural risks and disasters, security, localization, messaging. Global services extent legacy communication services such as those provided by GSM, GPRS, or the Internet. They make these legacy services available to mobile users. Note that some services can make use of both local and global communication paradigms. Examples are "ad-hoc google" and asynchronous messaging.

In the next section, we position Pocket Switched Networking with regard to related initiatives in mobile networking. We then discuss the challenges that Pocket Networking must solve, and present experiments into the feasibility of Pocket Networking.

## 2 Related Architectures

Pocket Switched Networking falls under Delay Tolerant Networking (DTN) umbrella. The delay Tolerant Networking research Group <sup>1</sup> defines itself as follows: "The Delay-Tolerant Networking Research Group (DTNRG) is concerned with how to address the architectural and protocol design principles arising from the need to provide interoperable communications with and among extreme and performance-challenged environments where continuous end-to-end connectivity cannot be assumed. Examples of such environments include spacecraft, military/tactical, some forms of disaster response, underwater, and some forms of ad-hoc sensor/actuator networks". The Delay Tolerant Networking (DTN) architecture, routes self-contained messages ("bundles") through networks with long delays, high error links, and intermittently connected, pre-scheduled, or opportunistic link availability. DTN messages contain information about service requirements and setup, though there is little notion of using application-level information to assist in forwarding decisions. However, the DTN RG does not make the assumption that the current DTN architecture is the only one possible.

Therefore, Pocket Switched Networking is a specific application domain of DTN. However, we take a radically different approach than most of the DTN

---

<sup>1</sup> [www.dtnrg.org](http://www.dtnrg.org)

related work to date. Instead of trying to extend the Internet legacy applications to support intermittently connected communication environment, we choose to design a new communication architecture, orthogonal to the Internet, that can use the Internet (as any other local communication) when available.

We believe that under the PSN assumptions described above, IP-centric networking is not a sensible approach. Reasons are abundant, from the need for the sender to determine the IP address of the recipient before sending data, to the use of closed-loop protocols such as TCP, SMTP and HTTP which employ a sequence of end-to-end exchanges for data transfer. In addition, IP-centric networking often relies on the availability of infrastructure services (e.g. DNS) that are not systematically available to mobile users. We assert that most attempts in this area are designed to extend IP-centric networking to new environments, and rely on the same invalid end-to-end assumptions.

Mobile Ad-Hoc Networks (MANET)<sup>2</sup> attempt to utilize local bandwidth without the presence of an infrastructure provider. However, they are IP-centric and aim to provide Internet style routes. For example, protocols such as AODV [9] depend on contemporaneous connectivity between the endpoints, and do not work if the only connectivity available is asynchronous and depends on mobility of nodes. Both MANET and DTN require a sender to know the recipient address for a given communication. In PSN, such an assumption cannot be made, as the destination may be a particular node, a class of nodes, or any node able to service the request.

Some sensor networks act in an opportunistic fashion. One example is ZebraNet, which uses intermittent connections between zebra-mounted nodes to transfer sensor data and collect statistics about zebra populations. This and other similar projects do not target the mobile user domain of PSN, and thus do not address challenges such as trust and usability.

There is an interesting synergy between PSN and pervasive computing [11]. Both are user-centric, and face the challenges of trust, usability, and the need to collapse layered networking models to accomplish their goals. It is our belief that PSN provides a networking abstraction which serves the needs of pervasive computing much more cleanly than IP-centric approaches.

### 3 Challenges

We have defined PSN as a communication paradigm capable of taking advantage of both local and global connectivity, as well as device mobility to convey messages or queries<sup>3</sup> to an appropriate endpoint, in the absence of contemporaneous end-to-end connectivity and global services. In this section, we identify the challenges that have to be addressed to successfully implement PSN. We outline previous attempts to address these challenges, and highlight the key problems of PSN yet to be solved.

---

<sup>2</sup> [www.ietf.org/html.charters/manet-charter.html](http://www.ietf.org/html.charters/manet-charter.html)

<sup>3</sup> These two terms are used interchangeably to mean “transmission data units.”

### 3.1 Usability

Opportunities are surprising, and users often dislike surprises. The success of PSN will depend on our ability to address two concerns. First, we need to provide some level of predictability of the behavior of PSN, although we cannot usually provide deterministic performance or 100% provable reliability. Because of this, the second concern is to provide appropriate feedback to users about the state of the system.

These concerns can only be addressed through the development of applications that perform useful tasks.

### 3.2 Naming

Naming fulfills two basic functions: it provides a level of indirection, and a way to identify things meaningfully. Names are bound to identifiers, typically by a *name service*. The service takes the name as a key, possibly with some attributes that provide more semantic clues or hints, and returns a more specific “lower level” identifier.

Traditionally, naming is implemented by some distributed set of services. It is questionable whether a *name service* per se is actually necessary in the context of PSN. A naming scheme is needed so that communication between named entities concerning named objects can be carried out [3]. Such names may need to be constructed dynamically or modified by attributes. Name construction may not need to be specified in advance; it can be an emergent property of the node behavior or state. In this sense PSN has features of distributed systems such as LIME [6] and Content Addressable Networks [10].

### 3.3 Security

PSN operates in an environment where a number of resources are at risk. Adversaries have several targets, including messages, nodes, transfer opportunities, and the models of user mobility embedded in individual nodes. Potential classes attacks include redirection, impersonation, eavesdropping, piercing of anonymity, fabrication, denial of service, and poisoning.

Solutions designed for ad-hoc networks may not be appropriate. Techniques which rely upon on-demand access to a centralized service cannot be used, nor can the assumption be made that all intermediate nodes are trusted. Admission control and in-network authentication, although effective in other contexts, are not sufficient to protect against malicious nodes in PSN, as all nodes are potentially malicious.

The DTN Research Group (DTN-RG) has suggested the use of identity-based encryption (IBE) [1], which has the property that public keys can be generated off-line on the basis of an arbitrary string (often a node identifier) and before the private key is calculated. Naming and addressing increase in importance in such a network. Private keys could be obtained while global connectivity is available, either before or after receipt of an encrypted message.

There are many opportunities for innovative work in the area of securing PSN. Locality information could be used to prevent Sybil and other identity attacks. Nearby (either logically or physically) nodes could create localized incentive or reputation systems. Finally, it will be necessary to develop mechanisms for preserving a user's privacy (both of location and identity) whilst still allowing messages to reach them.

### 3.4 Forwarding

Forwarding is the key challenge in opportunistic networking, as the utility of PSN is strongly correlated with the number of messages that reach their destination. The problem of forwarding is simple to describe: when nodes have a local or global connection opportunity, messages are forwarded according to some policy, with the intention that they are brought "closer" to their destination.

Local forwarding makes use of intermittent and mobility-based connectivity. This precludes formation of routes; instead, nodes must forward messages according to knowledge of their local environment and of the messages themselves. How best to acquire and interpret this information is a difficult problem. In addition, availability of storage and energy may affect the willingness of a node to forward messages, as discussed in Section 3.6.

When global connectivity is available to a node, messages can be forwarded directly to suitable nodes which are also globally connected, or to available proxies for those recipients (e.g. by encapsulating a message as an email and sending it to a recipient's IMAP server). The latter allows for a recipient's device to retrieve the message during a subsequent period of global connectivity. However, the sending device should not necessarily discard the message after forwarding it, as it may encounter the recipient directly before the recipient has had a global connection opportunity.

Prior work on message forwarding has focused on making Internet services available in a disconnected setting, exploiting nearby resources where possible; the 7DS system [8] is an example of this approach. An initial scheme for true message forwarding was proposed by Davis et al. [4] on the basis of last seen nodes, and variants of the algorithm were later presented by others [5]. There have been algorithms reported for rumor- and gossip-type communication at the application layer<sup>4</sup>. There have also been biologically and physically inspired schemes for communication in specific problem domains (e.g., sensor nets). Zhao, et al. [12], Burns, et al. [2] and DTNRG examined the use of a series of predictable, reliable, but non-contemporaneous links for routing.

For PSN, the challenge is in developing methods to determine which neighboring nodes provide good forwarding opportunities for a given message. To guide this process, meaningful communication and mobility data for the problem domain are required. Real world data of this sort are scarce, and random models are inappropriate as there is no structure for intelligent forwarding al-

---

<sup>4</sup> [www.grapewineproject.org](http://www.grapewineproject.org)

gorithms to exploit. Real systems must be built, measured, and learned from in order to make progress on this most important facet of PSN.

### 3.5 Mobility

A number of major challenges arise from the mobility of nodes. Intermittent communication links, the associated long messages lifetimes, and the movement of nodes within the network all pose problems for the timely, reliable and efficient delivery of messages. The short-lived nature of links presents a further problem, as currently deployed wireless technologies such as Bluetooth and WiFi were not designed with short-lived connection opportunities between power-limited devices in mind.

Short-lived connection opportunities are an inevitable consequence of high mobility and short radio range. In a realistic environment, Class 2 Bluetooth devices provide usable throughput at distances of about ten meters (see Section 4). Class 1 Bluetooth devices and 802.11 radios are too power hungry for continuous use in battery powered devices, though they may see use in other environments. These ranges support connection opportunities which last on the order of tens of seconds with typical pedestrian and vehicle speeds.

Searching for and connecting to other nodes opportunistically must be made efficient in the absence of a central coordinator. Protocols that minimize transmission delay and maximize the amount of data sent over short-lived, error-prone links must be developed and evaluated. As mentioned in Section 3.4, traces of real-world mobility must be collected and analyzed, potentially leading to the development of more realistic synthetic models.

### 3.6 Resource Management

There are two main resource management issues in PSN: network scheduling and energy conservation. With opportunistic forwarding, network interface scheduling becomes much more complex than IP-centric outgoing queues. It includes issues such as balancing time spent discovering neighbors with time spent transmitting data, handling limits on transfer opportunities imposed by mobility, and ensuring fair sharing of available radio spectrum with other devices. These problems are similar to those of congestion control in global end-to-end networks.

Energy concerns are likely to lead to culling of potential transfers, where the energy cost outweighs the expected benefit of the transfer. Other conservation techniques include duty cycling and wake-on-LAN which avoid the continuous powering of network interface receivers. Among the techniques known are using a high-power and low-power radio, adaptation to observed temporal and spatial availability of power, and preferentially forwarding via powered nodes when they are available.

## 4 Feasibility of Pocket Switched Networking

As described in Section 3.5, transfer opportunities are time-limited, and existing technologies and protocols are not designed for this case. Nonetheless, devices supporting Bluetooth and 802.11 are widely deployed, and their numbers are expected to increase rapidly in the future<sup>5</sup>. In this section, we use measurements and/or simulations of Bluetooth and 802.11 data transfers. The amount of data that can be transferred between two mobile nodes that encounter one another is dependent upon several factors: the time required for the nodes to discover one another, the time that the nodes are within radio range of one another, and the variation of throughput with range and operating environment. We measured each of these factors for both Bluetooth and 802.11. Results are described below

### 4.1 Bluetooth transfer opportunities

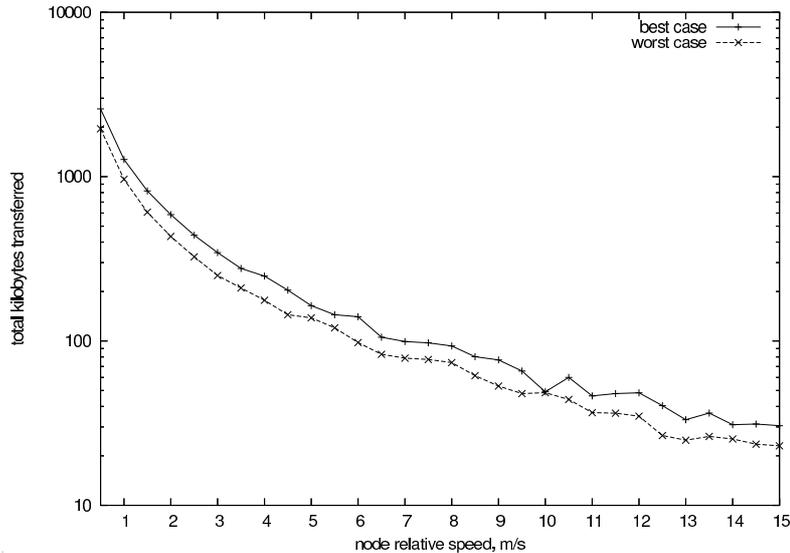
We performed our measurements using the PC laptops running Windows XP and class II Bluetooth USB devices manufactured by MSI and Belkin. We observed no significant performance differences between the two varieties of device. The measurements presented here were obtained using the MSI devices, and were taken one meter above the ground. All other wireless devices in the machines were removed. The goodput between devices was measured at various distances by opening an RFCOMM connection between the machines and sending 64 kilobyte messages from the initiator to the slave. We performed this measurement in two environments: indoors, in an office corridor in the presence of background 802.11 and Bluetooth interference, and outdoors, in a field far from such interference. Finally, we created a simulation of limited-duration transfer opportunities. In the simulation, two nodes performing inquiry approach one another head on, pass at some relative speed, and eventually move out of range. Once inquiry is successful, one node sends data until out of radio range. The average number of kilobytes transferred (in 50,000 experiments at each point) in two cases is shown in Fig. 1. The solid line indicates the results obtained using the best case values from the experiments. The dashed line indicates the results obtained using the worst case values. At walking speed, around 1Mb of data can be exchanged during a contact opportunity that would last approximately 10s. Although these results are preliminary, they indicate that Bluetooth is usable for opportunistic data transmission.

### 4.2 802.11 transfer opportunities

In [7], experiments are performed with a wireless Host in a car passing by a 802.11 Access Point at various speeds. They show that even at 180kph, 1.5Mb of data can be sent from the fixed point to the mobile host with both TCP and UDP in a single transfer opportunity (i.e within a 10s time interval). At 80kph, they could transfer up to 6Mb of data in one transfer opportunity (i.e. 36s).

---

<sup>5</sup> [www.bluetooth.org](http://www.bluetooth.org)



**Fig. 1.** Expected volume sent during a transfer opportunity with Bluetooth.

We have also performed our own 802.11 experiment Fig. 2. We observe similar results as in [7]. below 20 meters per second (around 70kph), above 10Mb of data can be transferred with TCP. And a couple of Mb can still be transferred at and above 100 kph.

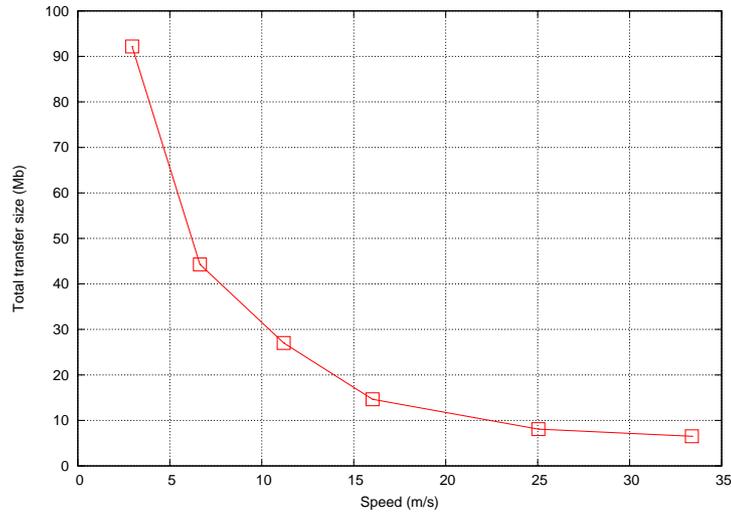
In both Bluetooth and WiFi case, the numbers above, despite realistic, should be considered upper bounds as more protocols can interfere (e.g. VPN, encryption, etc.) to reduce the amount of data transferable. However, new technologies can be designed that allow more data to be transferred in short time intervals. The main observation to retain from this section is that both Bluetooth or WiFi can be used in the context of PSN.

## 5 Enabling technologies

We now discuss components that need to be provided in order to support the challenges of PSN.

### 5.1 Community-based Networking

We expect PSN to enable a new family of applications with a high degree of spatial or logical locality. We refer to these areas of networking as communities, and provide explicit support for community formation and management with PSN. We believe that this notion of community will make propagation of information in PSN easier to achieve and control. Service requests may have more success if forwarded preferentially through a community. Communities can help improving



**Fig. 2.** Expected volume sent during a transfer opportunity with 802.11.

the security of transactions. A node might give preferential treatment to a service request related to a community for which this node has state information. Examples of such communities are participants to a symposium, fans of Elvis Presley, London subway users, etc.

Communities are formed in a distributed fashion. The impetus for community formation may be implicit or explicit. An example of the former would be a common geographical location or a common interest, while communities may form in the latter manner around a given event, such as a conference. To form communities based upon space-time proximity, a TTL field in messages may be appropriate. Other communities with more tightly controlled access may allow open membership, vote to decide upon membership, or appoint a trusted authority to control the membership of the community. Communities might provide some information that can later be used by a node to make a decision (i.e. in a conference, the list of registered participants can be provided prior to the events). Community is also a convenient paradigm to secure PSN: nodes can create shared keys to allow for private communications, and utilize known techniques for key revocation within the community.

## 5.2 Security

Security is a major issue in PSN and several important security services must be provided to users. In this section, we discuss issues related to secure distributed naming, authentication, trust, reputation systems, and incentive to cooperate.

**Identity and Trust** A user can have one or more identities. Some of these identities will be specific to a community. On the other hand, they are not

device-specific and a PSN user can use the same identity on various devices. An identity can be (non exhaustive list) an email address, a URL, a names, a picture, or any combination of the above.

Identities will be tied to public/private key pairs generated by the user, and may be shared or moved between nodes. In traditional public key cryptosystems, the bootstrapping of trust<sup>6</sup> is a difficult problem, solved by either a trusted third party or a distributed web of trust, such as that in PGP. PSN is particularly well suited to easy deployment of a web of trust, as users will be carrying small devices and can quickly and conveniently bootstrap trust between one another. In the case of more managed communities, a trusted third party can serve as an authenticator of identity. This notion of identity will serve as a building block for more advanced security services, as other users will attach notions of trust, reputation, content, and capability to an identity.

**Reputation System and Incentive to Cooperate** A device implementing PSN will provide a reputation system to discourage malicious behaviors. Reputation systems have a different (thus complementary) purpose than trust control. Trust control is about getting the guarantee a user really is who he claims to be. While reputation is about quantifying how good a citizen a node or user is.

In PSN, we must be able to adjust reputations not only on the basis of performance, but on degree of trust, on the community, on the kind of service request, and other metrics. Application level behavior may draw upon this reputation among other mechanisms. For example, a user might trust anyone to forward his messages, but only accept address book update from identities whose reputation exceeds a certain threshold.

Therefore, reputation systems will be a strong components (not necessarily the only) in creating an incentive to cooperate. Encouraging users to contribute resource (memory, battery, time, etc.) will be a major problem in PSN. The problem is solved in Kazaa by limiting the amount of information that is made available to those that do not want to contribute, but just consume. We can implement the same kind of mechanism in PSN. However, we believe that communities will be a strong element in getting a given user more cooperative.

### 5.3 Localization

Most PSN applications will rely on locality information such as geographical location, or neighborhood. However, localization does not necessarily mean GPS. There are numerous localization algorithm that could be implemented. Each of them will match specific community needs, and have an impact on the way services can be provided and service requests are forwarded.

### 5.4 User Interface

In the current networking world, users are often forced to make routing decisions when trying to send data to local recipients, e.g. having to pick one of email,

---

<sup>6</sup> Trust is the belief that a person is who his identity claims he is.

infrared transfer, Bluetooth transfer, USB key, or another method when wishing to transfer files to other local recipients — and the list keeps growing. This is precisely opposed to the goals of transparency and ease of use which are held dear to computer users. In the PSN, we may be able to offer the advantage that both local and wide-area connectivity are made transparent to users.

However, transparency is not necessarily the only goal. It is important that users remain apprised of the status of the delivery of the data they send and request. With end to end communications, this is relatively simple to determine and to display for the users convenience, e.g. using a spinning globe icon in a web browser, which stops spinning when page-load is complete. In PSN, not only are there many more states which one may wish to communicate, but a PSN client may also have little indication of the network status, since some nodes it was communicating with are no longer visible. Allowing PSN users to achieve and maintain an intuitive mental model of the status of their on-going data transfers may be a key issue in providing usable and deployable PSN applications.

Explicit user involvement in certain situations is also necessary, for example in determining a trust relationship if there is no prior community network of trust to draw upon, or in mapping a device's public key to a user that it claims to represent (similar to the way `ssh` maps keys to hosts). Users should also retain control over all operations since they may involve the spending of scarce resources such as storage, bandwidth or battery.

## 5.5 Monitoring

A monitoring module will collect traffic information to make it possible to analyze a number of "dispatching" or "forwarding" strategies, applications behavior and so forth. Monitoring is also key to PSN as the information collected by the monitoring modules could be used to optimize forwarding decisions or to take part to decisions on the trustability of some node.

## 6 Conclusion

We aim to implement PSN for *mobile computing devices*, which obviously includes notebook PCs and PDAs. Recently, this term has also become applicable to mobile phones, which now have significant storage, computing power, local networking (generally in the form of Bluetooth), and support for dynamically-loaded applications. Our implementation of PSN is known as Huggle.

Over the next few years, we aim to address the research challenges described above and to build Huggle-based applications including distributed Usenet-style newsgroups, messaging, file-sharing, and web browsing with automatic use of neighbors' caches. We plan to test these applications by rolling out prototypes to both local users and larger groups such as conference attendees. This will enable us to study aspects of Huggle including usability, scalability, network congestion, and user behavior, which can only be conclusively studied in deployments.

We also expect to release our implementation of the Huggle infrastructure and example applications under an open source license, and to encourage downloads and additional deployments by users as well as other research groups.

Please visit the Huggle project web<sup>7</sup> for access to all project resources.

## 7 Acknowledgments

David Blunden and Neeraj Sharma (with their team) helped us collect the 802.11 in-motion data. Marc Liberatore and Brian Levine from University of Massachusetts Amherst contributed to Bluetooth data collection and analysis. Ralph Kling and the SNO team helped us with the iMotes.

This work has been partly supported by the European Union under the E-Next NoE FP6-506869, and under the ACCA CA FP6-IST-6475.

## References

1. D. Boneh and M. Franklin. Identity based encryption from the Weil pairing. *SIAM Journal of Computing*, 32(3):586–615, 2003.
2. B. Burns, O. Brock, and B. N. Levine. MV routing and capacity building in disruption tolerant networks. Technical Report TR-04-68, University of Massachusetts at Amherst, July 2004.
3. J. Crowcroft, S. Hand, R. Mortier, T. Roscoe, and A. Warfield. Plutarch: An argument for network pluralism. In *ACM SIGCOMM*, Aug. 2003.
4. J. A. Davis, A. Fagg, and B. N. Levine. Wearable computers as packet transfer mechanisms in ad-hoc networks. In *International Symposium on Wearable Computing*, Oct. 2001.
5. M. Grossglauser and M. Vetterli. Locating nodes with EASE: Mobility diffusion of last encounters in ad hoc networks. In *IEEE INFOCOM*, 2003.
6. A. L. Murphy, G. P. Picco, and G.-C. Roman. LIME: A middleware for physical and logical mobility. In *International Conference on Distributed Computing Systems*, pages 524–233, June 2000.
7. J. Ott and D. Kutscher. Drive-thru internet: IEEE 802.11b for automobile users. In *IEEE INFOCOM*, 2004.
8. M. Papadopouli and H. Schulzrinne. Performance of data dissemination among mobile devices. Technical Report 005, Columbia University, 2001.
9. C. E. Perkins and E. M. Royer. Ad hoc on-demand distance vector routing. In *IEEE Workshop on Mobile Computing Systems and Applications*, pages 90–100, feb 1999.
10. S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content addressable network. In *ACM SIGCOMM*, 2001.
11. M. Satyanarayanan. Pervasive computing: Vision and challenges. *IEEE Personal Communications*, Aug. 2001.
12. W. Zhao and M. Ammar. Message ferrying: Proactive routing in highly-partitioned wireless ad hoc networks. In *IEEE Workshop on Future Trends in Distributed Computing Systems*, May 2003.

---

<sup>7</sup> [www.cambridge.intel-research.net/huggle/](http://www.cambridge.intel-research.net/huggle/)