Bluetooth: An Enabler for Personal Area Networking

Per Johansson, Ericsson Research
Manthos Kazantzidis, Rohit Kapoor, and Mario Gerla, UCLA

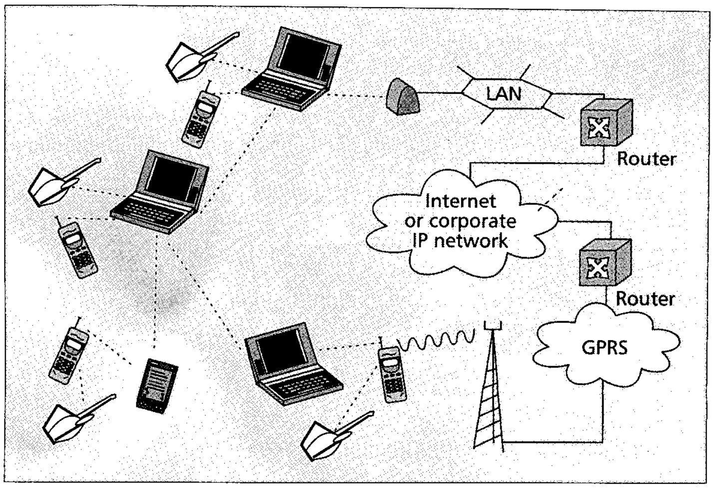
Abstract

We find ourselves today often carrying numerous portable electronic devices, such as notebook computers, mobile phones, PDAs, digital cameras, and mp3/MD/DVD players, used to help and entertain us in our professional as well as private lives. For the most part, these devices are used separately, and their applications do not interact. Imagine, however, if they could interact directly and thus create a network where information may flow seamlessly between the devices — such a network of personal devices is often referred to as a personal area network, or PAN. Moreover, access to the Internet via a (public) wireless LAN access point and/or via a 3G UMTS mobile phone would enable the PAN to be constantly online. The strongest candidate to provide the cheap short-range radio links necessary to enable such networks is the Bluetooth wireless technology. Seen from a networking perspective, a PAN will be expected to have participants, both of its "own" devices and "guest" devices from other PANs, continuously moving in and out of its coverage. To cope with this volatile nature of the network, the concept of *ad hoc networking* may be applied to create robust and flexible connectivity. A major technical step is taken when the Bluetooth *piconet* network architecture, a strict star topology, is extended into a *scatternet* architecture, where piconets are interconnected. A consequence of creating scatternet-based PANs is that some nodes will form gateways between piconets, and these gateways must be capable of time sharing their presence in each piconet of which they are members. While the Bluetooth standard defines the gateway nodes, the actual mechanisms and algorithms that accomplish the interpiconet scheduling (IPS) are left rather open. Given the lack of research literature in the subject, an overall architecture for handling scheduling in a scatternet is presented in this article. A family of feasible IPS algorithms, referred to as *rendezvous point* algorithms, is also introduced and discussed.

The number of mobile devices we carry around these days, for both work and pleasure, seems to steadily increase. Our notebook computer, cellular phone, and PDA have all become almost necessary tools not only for the business traveler but also for an increasing part of the general public. Even if these devices have become both more powerful and smaller in size, we now often add an MD/mp3 player or even a DVD player to complete the electronic "wardrobe." For the most part, these devices are used separately and their applications do not interact, mainly because interconnection via wires becomes tedious and time-consuming. This niche was also identified by a group of mobile phone vendors and notebook computer vendors, who initiated the development of the Bluetooth [1] wireless technology. Bluetooth uses a frequency-hopping scheme in the unlicensed industrial, scientific, and medical (ISM) band at 2.4 GHz. In 1998 the Bluetooth Special Interest Group (SIG) was formed to host the work on Bluetooth specification, and the first version of the specification was released in 1999. The major goal of the Bluetooth wireless technology is to allow relatively cheap electronic devices to communicate directly in an ad hoc fashion, which requires the price of a Bluetooth radio to be only a few dollars. Moreover, Bluetooth-equipped devices can also form networks where information may flow seamlessly between the applications hosted in the devices; such a network of personal devices is often referred to as a *personal area network* (PAN).

One example of an application within one PAN is the electronic business card of a new contact that automatically finds its way across the PAN into the address register on a notebook computer and the number register on a mobile phone. Communication between PANs would, for instance, enable participants at a meeting to share documents or presentations. Access to the Internet via a (public) wireless LAN (WLAN) access point and/or via a third-generation (3G) cellular phone would enable the devices in the PAN to be constantly online. For instance, commuters may have a public WLAN access point in a train to their notebook computers; when exiting the train their notebook computers could remain online via a 3G phone, while incoming e-mail could now be diverted to their PDAs through the PAN. Finally, as they enter their offices, the access could, again automatically, go through the notebook computer via WLAN access to the corporate campus network.

In parallel with the development of the Bluetooth PAN technology, the introduction of 3G mobile networks is underway at a very rapid pace. The 3G systems — that is, Universal Mobile Telecommunications System (UMTS) and cdma2000, will move the Internet into the mobile world in a useful way, moving on to higher data rates on the order of hundreds of



■ Figure 1. *Four interconnected Bluetooth PANs, with both a WLAN and a GPRS access running simultaneously.*

kilobits per second instead of tens. This development will create a push for mobile terminals that can offer high-resolution images and high-quality audio, requiring more computing capacity; the mobile phone will evolve from a pure voice service terminal to a multimedia platform. However, the introduction of the PAN concept may very well widen the design space for 3G terminals. Since a PAN would enable a network of devices, the 3G terminal can be kept rather simple, while, say, a PDA could offer the more powerful computing necessary for multimedia applications. The Bluetooth wireless link interconnecting the two devices would be able to carry the multimedia stream. Moreover, the form factor of the 3G phone in this scenario may vary from a simple voice terminal to a full-fledged palmtop multimedia terminal. Thus, the Bluetooth ad hoc PAN scenario enables the user to personalize her personal media environment by allowing free distribution of applications and functions within the Bluetooth PAN — a wider variety of terminals may be accepted to access, and take advantage of, 3G networks.

Bluetooth presents a number of technical challenges from a networking perspective, such as ad hoc network formation and scheduling of traffic between nodes simultaneously operating on different channels. Bluetooth operates inherently in an ad hoc manner since it does not rely on any infrastructure via, say, an access point or a base station. This is reflected in the way nodes are detected and how networks such as PANs are created without, or with minimal, preconfiguration. The participants of a Bluetooth network are expected to be mobile and to move in and out of coverage, and nodes may also join or leave the network rather frequently. For instance, a PAN may have both its "own" devices and "guest" devices from other PANs. The characteristics of a Bluetooth PAN will in many cases be such that the concepts of ad hoc networking [2] fit very well and could help to create robust and flexible network connectivity. Furthermore, a major technical step is taken when the Bluetooth *piconet* network architecture, a strict star topology,[1] is extended into a *scatternet* architecture by interconnecting piconets. The combinations of network topologies will increase dramatically, and methods to create robust, but still efficient, scatternets become crucial. To create good scatternet connectivity, bandwidth and delay requirements, among others, must be considered. In addition, many mobile phones and other electronic devices already are or will soon be Bluetooth-enabled. Consequently, the groundwork for building more complex ad hoc PANs is being laid.

The rest of this article is organized as follows. In the following section, the ad hoc PAN, its usage cases, and its typical characteristics are discussed in more detail. Networking of the Bluetooth wireless system is then described, together with some general technical background on Bluetooth. A general architecture for scatternet scheduling is given together with some categories of interpiconet scheduling (IPS) algorithms. Finally, some conclusive remarks are given to summarize the work.

The Ad Hoc PAN: A Network Extension

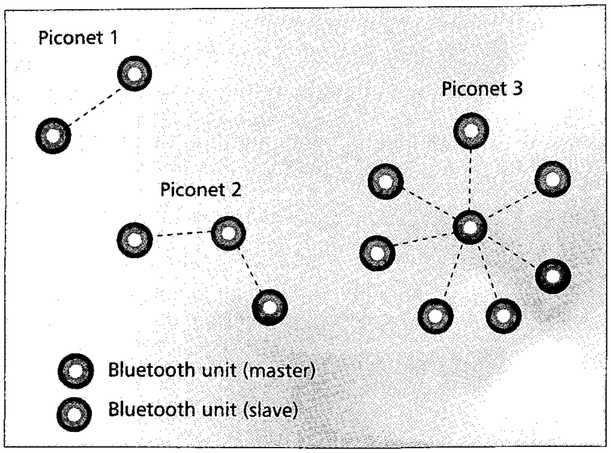
Before continuing with a more detailed technical description of Bluetooth networking, a more general discussion of applications is given to motivate the design of Bluetooth PANs in the first place. Obviously, just a single Bluetooth link by itself will simplify intercommunication between various mobile devices (e.g., a cellular phone and a PDA) by eliminating cables. However, it is with the introduction of Bluetooth ad hoc networking on the PAN level that the true potential of ad hoc PANs can really be utilized. The ad hoc nature of these networks should be emphasized, not only as isolated PANs but also as PANs interconnected with larger public network infrastructures, such as UMTS. It is likely that the first ad hoc network application to really move into the commercial market segment will be in the form of PANs.

Seen from the viewpoint of the traditional mobile network, a Bluetooth-based PAN opens up a new way to extend mobile network services into the user domain, out to other devices further out in the PAN. In terms of traffic load offered to the network, the aggregate traffic of the PAN would typically exceed that of a single mobile phone. In addition, if several Bluetooth PANs are interconnected, this offered traffic will be further increased. For a mobile network operator this could be a very attractive way to get users to increase utilization of the new 3G networks as well as extending the user applications in the 3G phones. The high-bandwidth consumer in these PANs would typically be a notebook computer or a high-end palmtop computer.

Figure 1 illustrates the case discussed above with a scenario of four Bluetooth PANs, consisting of notebook computers, cellular phones, headsets, and PDAs. The PANs are interconnected via three notebook computers and one PDA. In addition, two of the PANs are connected to an IP backbone network, one via a LAN access point and the other via a General Packet Radio Service (GPRS)[2] phone, thus creating two possible ways to access the IP backbone network. If the mobile operator also manages the WLAN access point, mechanisms could be deployed to divide the traffic between the wide-area 3G access and the local-area WLAN access to achieve efficient utilization of network resources. The PAN itself can also encompass several different access technologies, distributed among its member devices, that exploit the ad hoc functionality in the PAN. For instance, a notebook computer could have a WLAN interface (e.g., IEEE 802.11 [3] or HiperLAN/2 [4]) that provides network access when the computer is used where a WLAN access is available. When no WLAN access is available, the GPRS/UMTS phone would be

[1] *In a piconet the central controlling node is called the* master *and other client nodes in the star are called* slaves.

[2] *GPRS is often referred to as 2.5G and is an intermediate step in the evolution from GSM (2G) to UMTS (3G).*



■ Figure 2. *Three piconets.*

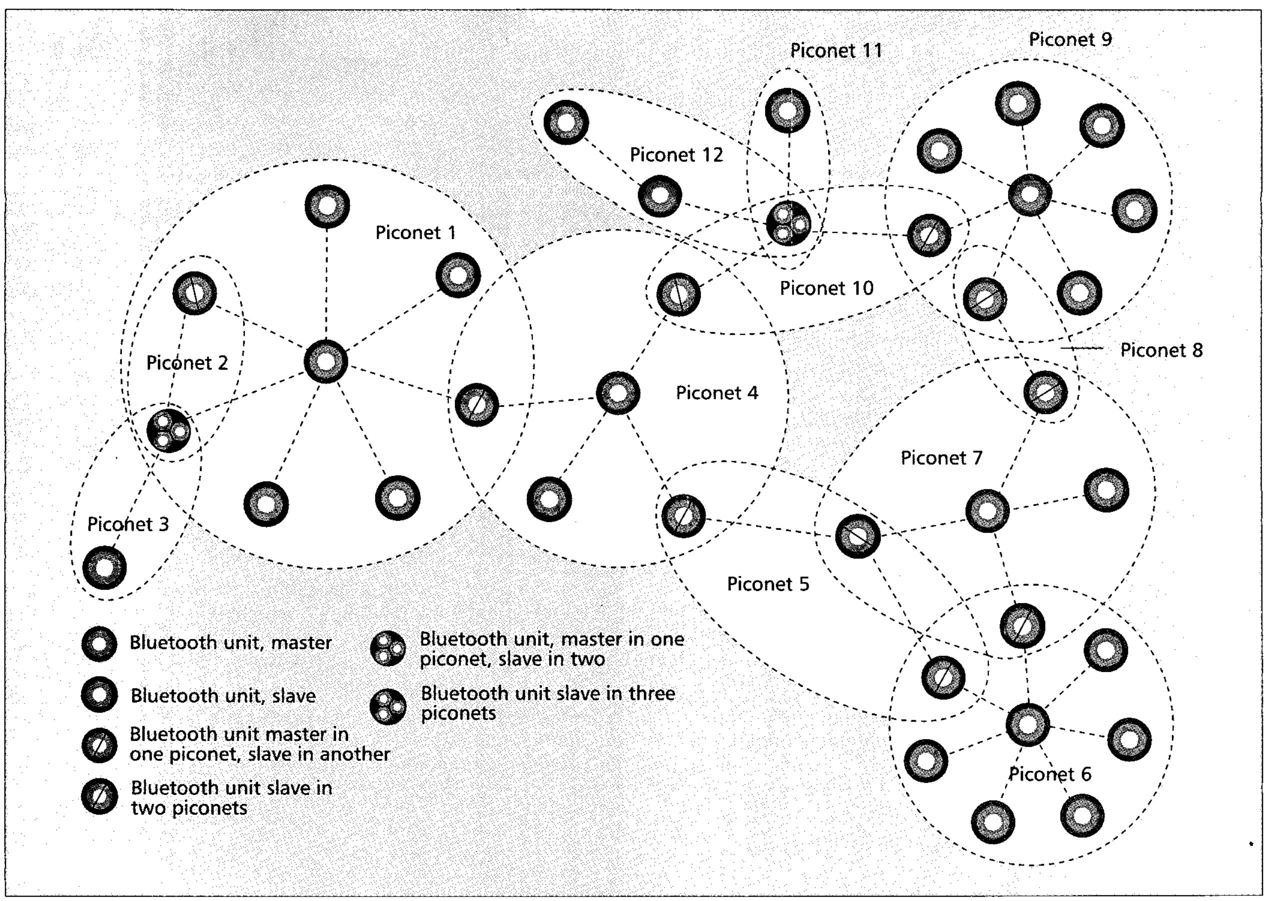
used to access the IP backbone network. Thus, the PAN would benefit from the total aggregate of all access technologies residing in the devices of the PAN.

As the PAN concept matures, new devices and new access technologies will be able to be incorporated into the PAN framework, as well as eliminate the need to create hybrid devices, such as a combined PDA-mobile phone, because the PAN network will instead allow wireless integration. In other words, it will not be necessary to trade off form for function.

PAN Performance Characteristics

An ad hoc Bluetooth PAN would be expected to operate in a network environment in which some or all of the nodes are mobile, in contrast to traditional wireline and wireless networks. In general, however, the same basic user requirements for connectivity and traffic delivery that apply to traditional networks will apply to ad hoc networks:

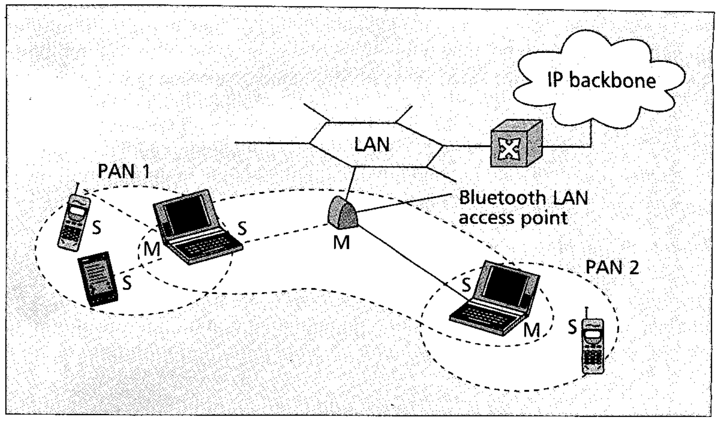
- Distributed operation: A node in the PAN cannot rely on a network in the background to support security and routing functions. Instead, these functions must be designed so that they can operate efficiently under distributed conditions.
- Dynamic network topology: In general, nodes will be mobile, which sooner or later will result in varying network topology. Nonetheless, connectivity in the PAN should be maintained to allow applications and services to operate undisrupted. In particular, this will influence the design of routing protocols. Moreover, a user in the ad hoc network will also require access to a fixed network (e.g., the Internet via a 3G access network) even if nodes are moving around. This calls for mobility management functions that allow network access for devices located several Bluetooth radio hops away from a network access point.
- Fluctuating link capacity: The effects of high bit error rates are more profound in a multihop ad hoc network, such as a Bluetooth PAN, since the aggregate of all link errors is what affects a multihop path. In addition, more than one end-to-end path can use a given link, so if the link were to break, it could disrupt several sessions during periods of



■ **Figure 3.** *Twelve piconets interconnected into one single scatternet.*

high bit error transmission rates. However, the Bluetooth link layer uses both automatic repeat request (ARQ) and forward error correction (FEC) techniques to counter these problems.

- Low-power devices: In most cases, PAN nodes will be battery-driven, which will make the power budget tight for all the power-consuming components in a device. This will affect, for instance, CPU processing, memory size/usage, signal processing, and transceiver output/input power.



■ Figure 4. *The notebook computer in PAN 1 operates as an interpiconet node (master in PAN 1 and slave to the Bluetooth access point). In this scenario packets are typically not forwarded by the interpiconet node.*

Bluetooth Networking

In order to implement ad hoc PANs based on Bluetooth wireless technology [1, 5–9], work is underway in the Bluetooth SIG to enhance current Bluetooth functionality to provide improved networking support. One key necessary feature is a very good capability to carry IP efficiently to/from and within a Bluetooth PAN. This is called for since Bluetooth PANs will be connected to the Internet via either 3G networks or corporate/public WLANs, and will contain IP-enabled hosts. Generally speaking, good capacity for carrying IP would give Bluetooth networks a wider and more open interface, which would most certainly boost the development of new applications for Bluetooth.

Before elaborating more on the technical aspects related to Bluetooth network functionality, a brief introduction to Bluetooth in general is given below.

Bluetooth Basics

Two or more Bluetooth units that share the same channel form a *piconet*. Figure 2 shows three examples of different piconet configurations. Within a piconet, a Bluetooth unit can play either of two roles: master or slave. Each piconet may only contain one master (and there must always be one) and up to seven active slaves. Any Bluetooth unit can become a master in a piconet.

Furthermore, two or more piconets can be interconnected, forming what is called a *scatternet*. In Fig. 3 a *scatternet* consisting of 12 piconets is depicted. The connection point between two piconets consists of a Bluetooth unit that is a member of both piconets. A Bluetooth unit can simultaneously be a slave member of multiple piconets, but only a master in one. Moreover, because a Bluetooth unit can only transmit and receive data in one piconet at a time, its participation in multiple piconets has to be on a time-division multiplex (TDM) basis.

The Bluetooth system provides full duplex transmission based on slotted time-division duplex (TDD), where the duration of each slot is 0.625 ms. There is no direct transmission between slaves in a Bluetooth piconet, only from master to slave and vice versa.

Communication in a piconet is organized so that the master polls each slave according to a polling scheme. A slave is only allowed to transmit after having been polled by the master. The slave will start its transmission in the slave-to-master time slot immediately after it has received a packet from the master. The master may or may not include data in the packet used to poll a slave. However, it is possible to send packets that cover multiple slots. These multislot packets may be either three or five slots long.

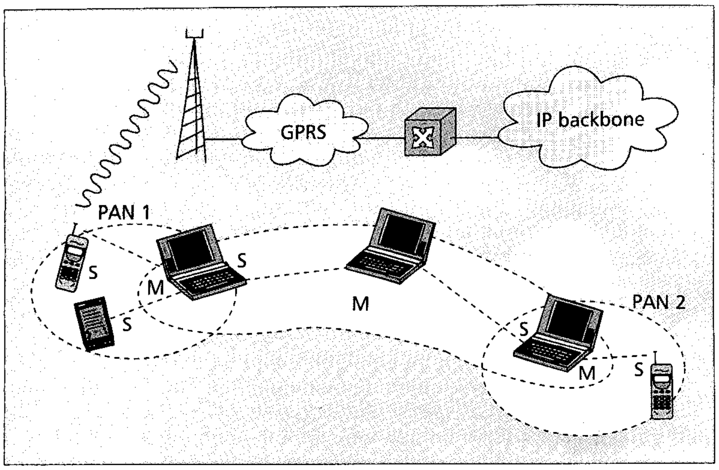
Scatternet-Based PANs

In a Bluetooth PAN context, the scatternet functionality is important to allow a flexible forming of ad hoc PANs. Even though a Bluetooth PAN will often be based on a single piconet, the possibility for a node in the PAN to be present in another piconet is essential to allow the combination of various Bluetooth usage cases. Nodes that are present in multiple piconets (i.e., interpiconet nodes) may either have applications that operate independently in the piconets, or function as a gateway between the piconets and forward traffic between them. The former, nonforwarding, case will typically occur when a notebook computer in the PAN has small accessories such as a mouse or joystick and at the same time needs to reach a Bluetooth LAN access point in the room. Then the notebook computer will in general be a master in the (single piconet) PAN and the accessories will be slaves. However, the LAN access point will in most cases need to be the master of the nodes with which it communicates. So if the notebook computer connects to the LAN access point, it will remain a master in its "own" PAN, while operating as a slave unit for the data access point (i.e., the notebook computer is an interpiconet node). However, there is no need to forward traffic between the data access point and the accessories, and thus no need to consider networking in this case. Figure 4 shows a case where two PANs are connected to a Bluetooth access point, forming a scatternet of three piconets. In this scenario the interpiconet nodes (the notebook computers in the PANs) do not forward traffic since the slave nodes in the PANs typically only need to exchange local information with their masters (the notebook computers in this case).

When the interpiconet nodes forward packets between piconets, Bluetooth ad hoc PANs belong to the class of multihop ad hoc networks. These scatternet-based PANs may be used when information needs to be spread widely among the PANs residing within reach through a "reasonable" number of radio hops. Here, the reasonable number of hops depends on the type of information in terms of required data rate and end-to-end delay. However, the number of possible hops depends heavily on the scatternet application. Where the scatternet is used to interconnect, say, a large network of Bluetooth-equipped sensors, the flow of information may settle on very low data rates and high delays, so the hop count could be in the orders of hundreds. On the other hand, a high-quality interactive video application within a PAN may handle only one Bluetooth hop and not be able to cross any interpiconet gateway due to the imposed delay.

Figure 5 shows a scenario in which a GPRS phone provides Internet access to all the notebook computers in the scatternet. This means that the notebook computer in PAN 1 operates as an interpiconet gateway, forwarding packets directly between PAN 1 and the notebook computers. In this case the interpiconet node needs to have network forwarding functions to be able to achieve the end-to-end path.

The scatternet functionality may also be used to improve



■ Figure 5. *The master of PAN 1 forwards packets from the GPRS phone to the rest of the notebook computers to give Internet access to the entire scatternet.*

the performance of a group of Bluetooth nodes that are either already part of a scatternet, or part of separate piconets. The roles of the nodes in such a group may be rearranged to adapt to a new traffic distribution among the nodes by changing the allocation of masters, slaves, and interpiconet nodes. For instance, if two slave nodes need to communicate, it might be wiser to create a new piconet that contains only these two nodes (Fig. 6). The nodes can still be part of their original piconets if traffic flows to or from them, or they need to receive control information. Since the frequency-hopping spread-spectrum (FHSS) system combined with fast ARQ makes Bluetooth very robust against interference, new piconets gain substantially more capacity than they lose as a result of increased interference between them.

Packet Forwarding in the Scatternet

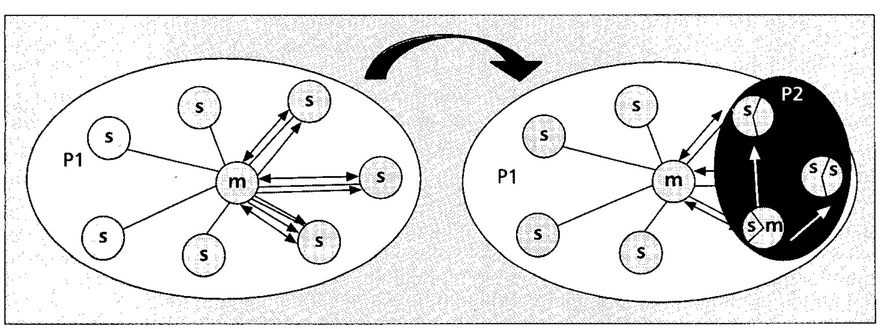
Packet forwarding — or routing — becomes necessary when packets must traverse multiple hops between the source and destination nodes. Given that IP will be commonplace in scatternet contexts, one might conclude that routing over the scatternet should be handled within the IP layer. However, there are good arguments for taking another course in this respect:

- The current IP Dynamic Host Configuration Protocol (DHCP) [10] and the emerging zero-configuration methods [10, 11] (IETF Zero Configuration Networking Working Group, zeroconfig) rely on link layer connectivity. These protocols are typically used to attain a dynamic IP address for an IP host or to select a random IP address [11]; both operations the devices of a PAN are most likely to need to perform. Generally, the protocols will not work beyond an IP router, which means they will not reach nodes located more than one Bluetooth hop away in an IP-routed scatternet. A scatternet that provides broadcast-segment-like connectivity would enable these protocols to work for Bluetooth-based IP hosts that are separated by multiple hops:
- To operate efficiently, the routing function should be joined with the function for forming scatternets. A routing function on the IP layer would thus need to be adapted to, or interact very closely with, the underlying Bluetooth layer, which violates the idea of keeping the IP layer as independent as possible from the link layer technology.
- Other non-IP-based applications may use the scatternet functionality provided by the Bluetooth networking layer, that is, the Bluetooth devices are not forced to host an IP layer to utilize the networking features of the scatternet.

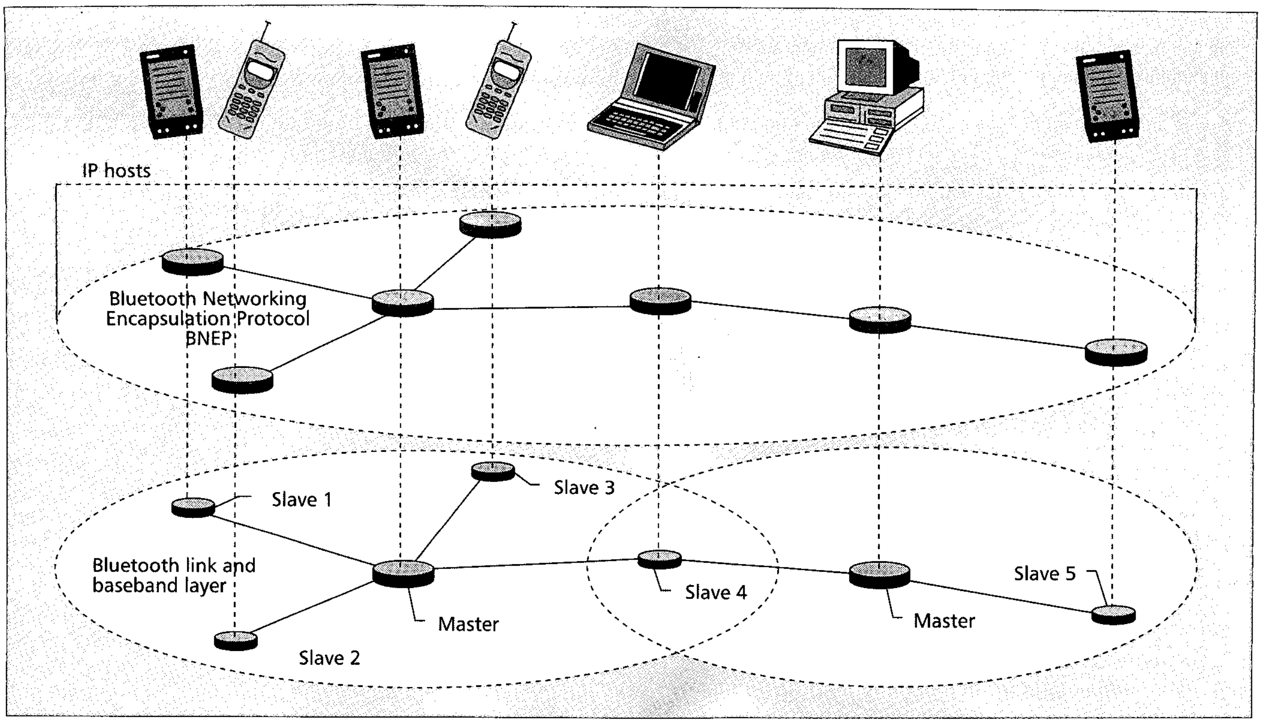
In summary, the best way to provide networking among the nodes in a Bluetooth scatternet would be to perform the routing on a Bluetooth network layer residing below IP. This approach is also pursued within the Bluetooth SIG PAN working group, where a networking protocol referred to as the Bluetooth Network Encapsulation Protocol (BNEP) is being developed to provide an Ethernet-like interface to IP. In Fig. 7 BNEP provides a broadcast segment across a scatternet consisting of two piconets, which enables ordinary IP hosts to be interconnected in a Bluetooth ad hoc scatternet. The scenario in Fig. 7 may be a case where several PANs are involved in an ad hoc meeting. Note that the piconets and PANs do not necessarily have to be mapped one to one; depending on the best topology, several PANs may be in one piconet or a PAN may consist of multiple piconets (a scatternet). This independence allows for efficient and flexible ad hoc PANs since rearrangements may be done without changing the PAN memberships.

In the first release of BNEP, the focus is to provide the broadcast segment within one piconet only, but BNEP holds the potential to offer a scatternet-wide broadcast segment. From a protocol point of view, BNEP will be placed on top of the link layer protocol of Bluetooth, the Logical Link and Control Adaptation Protocol (L2CAP). L2CAP comprises functions for segmentation and reassembling between packets from higher-layer protocols and Bluetooth baseband packets. Multiplexing of protocols apart from BNEP (e.g., OBEX, RFCOMM) over the Bluetooth radio link is also made possible in L2CAP. However, L2CAP is strictly link-oriented and cannot forward packets beyond one link since there is no concept of networkwide addressing within L2CAP. BNEP will add the networking functionality by utilizing the unique media access control (MAC) address of each Bluetooth interface. The location of BNEP in the Bluetooth IP stack is depicted in Fig. 8.

For many IP applications BNEP will replace the point-to-point approach which is currently the only supported IP stack within Bluetooth, that is, via Point-to-Point Protocol (PPP) over RFCOMM. The point-to-point approach may still be used for point-to-point-oriented applications such as dialup via a Bluetooth-enabled mobile phone. However, the introduction of broadcast support for IP will enable easy interworking with protocols that are



■ **Figure 6.** *The new piconet (P2) adds more capacity for the slave-to-slave traffic in P2 and also for the traffic remaining in P1.*

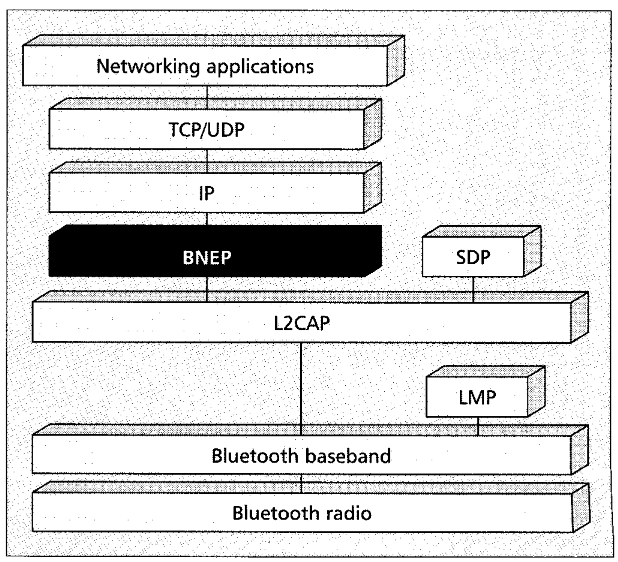


■ **Figure 7.** *The Bluetooth Networking Encapsulation Protocol will offer a broadcast segment infrastructure to the IP layer, similar to Ethernet, potentially spanning over an entire scatternet.*

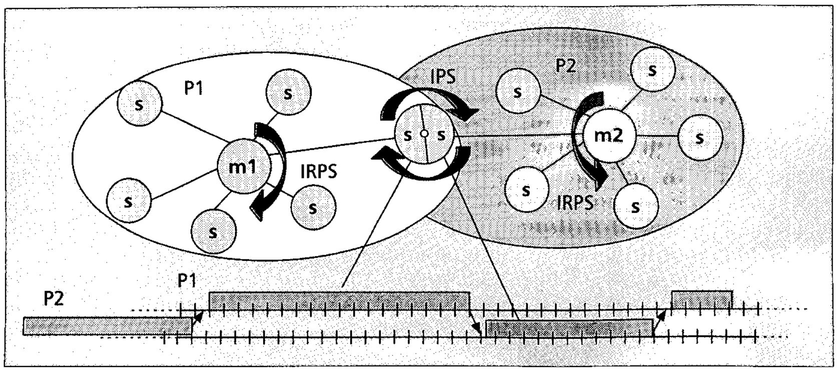
commonplace in the LAN/WLAN environment and also bring the same networking concepts into the Bluetooth PAN domain. Similar to Ethernet, BNEP may also be used by protocols other than IP, thus making scatternet capabilities available for more than just IP-based applications. Moreover, BNEP will be able to interact closely with the Bluetooth baseband functions during the establishment or teardown of piconets, which will facilitate the formation of efficient Bluetooth scatternets.

Scatternet Routing — Given the fact that IP will be serviced by a broadcast segment provided by BNEP, the issue arises of BNEP forwarding packets between source and destination across a scatternet. However, based on its ad hoc nature the Bluetooth PAN, or a network of such PANs, can be seen as a mobile ad hoc network (MANET). This means that results from research on ad hoc routing protocols may also be used for scatternet routing. Several proposed routing protocols for ad hoc IP networks (e.g., [12–14] have been submitted to the MANET working group [2] within the Internet Engineering Task Force (IETF). Even though these protocols are intended for IP routing, the algorithms can in general be applied to any network where the task is to establish a path between any two nodes with unique addresses. In the case of a scatternet, the 48-bit Bluetooth device address (BD_ADDR) may be used as the unique identifier of the scatternet nodes. Thus, an ad hoc routing protocol can use these in the same way as IP addresses are used in the IP versions of the routing algorithms. The capability to aggregate the IP address to allow scalable IP routing protocols can in most cases never be used in a MANET since the nodes may move randomly relative to each other. That is, the IP address is used purely as an identifier of the node (or the interface of the node) in the MANET context.

The inherent node mobility in an ad hoc network may cause a path, considered optimal at one time instant, to not work at all a few moments later. Moreover, the stochastic properties of the wireless channels and operating environment add to the uncertainty of path quality. Traditional routing protocols in the Internet are proactive in that they maintain routes to all nodes, including nodes to which no packets are being sent. They react to any change in the topology even if no traffic is affected by the change, which requires periodic control messages to maintain routes to every node in the network. The rate at which these control messages are sent must reflect the dynamics of the network in order to maintain valid routes (e.g., [15]). Thus, scarce resources such as power and link bandwidth will be used more frequently for control traffic as node mobility increases. An alternative approach involves establishing reactive routes, which dictates that routes between nodes are determined solely when they are explicitly needed to route packets.



■ Figure 8. *The location of BNEP in the Bluetooth IP stack.*



■ Figure 9. *A scatternet with one interpiconet unit that divides its time between the two piconets.*

the bandwidth capacity will be distributed among the slave units. The polling of slaves within a piconet results in scheduling of the slaves in the master unit, which is referred to as *intrapiconet scheduling* (IRPS). The IRPS function in the master should assess the capacity needs of the units in the scatternet to ensure that capacity is shared fairly, or according to any other preferred capacity-sharing policy. In [17] an efficient IRPS algorithm, Fair Exhaustive Polling (FEP), was introduced and shown to provide both high utilization and fair distribution of piconet capacity among the participating slaves. The main contribution of the algorithm is that it focuses the capacity on the active nodes with maintained fairness, while still being very simple from an implementation perspective.

In a scatternet, at least one Bluetooth unit is a member of more than one piconet. These interpiconet nodes might have a slave role in numerous piconets but can have the master role in only one. Irrespective of the roles, the interpiconet node must schedule its presence in all the piconets of which it is a member; hence, an *interpiconet scheduling* (IPS) algorithm is necessary in addition to the IRPS algorithm. The main challenge for the IPS is to schedule the presence of the interpiconet node in its different piconets such that the traffic can flow within and between the piconets as efficiently as possible. Given that the interpiconet node is a single transceiver unit, only one of its entities (master or slaves) can be active at a time, resulting in the node being blind in all piconets other than the one it is active in at any given moment. This means that any data exchange with an interpiconet node in a particular piconet must be done while it is present in that piconet.

The master unit in a piconet always expects a slave unit to be present when it sends a data or poll packet to it. If the slave unit is not present, the master may choose to disconnect the slave after some predefined timeout period since it may be seen as a case of an erroneous radio channel (e.g., the slave unit is out of range or the level of interference is too high). However, in a scatternet the slave may be an interpiconet node and may be visiting another piconet when the master sends it a packet. Thus, the main issue with IPS is to coordinate the simultaneous presence of an interpiconet unit and a master (this node may be a slave if the interpiconet unit itself has a master role).

Moreover, the time slots of piconet channels are, in general, not synchronized, which introduces a delay of one or two slots (0.625 ms each) each time a unit switches between piconets. This results in a piconet switching overhead that should be taken into account in the design of the IPS algorithm.

In addition, the combination of IPS and IRPS schedulers in the scatternet should be coordinated in order to give efficient scheduling of the units of the scatternet. For instance, an interpiconet node may be removed from the scheduling list of the IRPS for the period it is not present, but reinstalled in the list for more frequent scheduling once it is back in the piconet. In Fig. 9, a scatternet consisting of two piconets with one interpiconet unit (double slave role) is depicted. The two time lines in the figure show how the TDM for the interpiconet unit needs to take the phase shift into account when switching between the two piconets. When the interpiconet unit is active in one of the piconets, the IRPS algorithm of that piconet determines the amount of polling received by it.

This prevents the nodes from updating every possible route in the network, and instead allows them to focus on routes that are either being used or in the process of being set up.

Applied to the scatternet (e.g., as part of BNEP), a reactive ad hoc routing algorithm would also have to consider establishment of new Bluetooth connections in order to find efficient paths through the scatternet. Hence, forming and rearranging the scatternet should be part of or closely integrated with the scatternet routing algorithm. If the routing were purely IP-based, this integration would be more difficult to achieve since the scatternet would then be seen as a link layer and routing-initiated link establishments would need to be indirectly triggered through a message exchange over the Bluetooth host controller interface (HCI). Otherwise, the IP implementation for Bluetooth scatternets needs to be modified to contain Bluetooth-specific elements, generally not a preferred approach to IP networking.

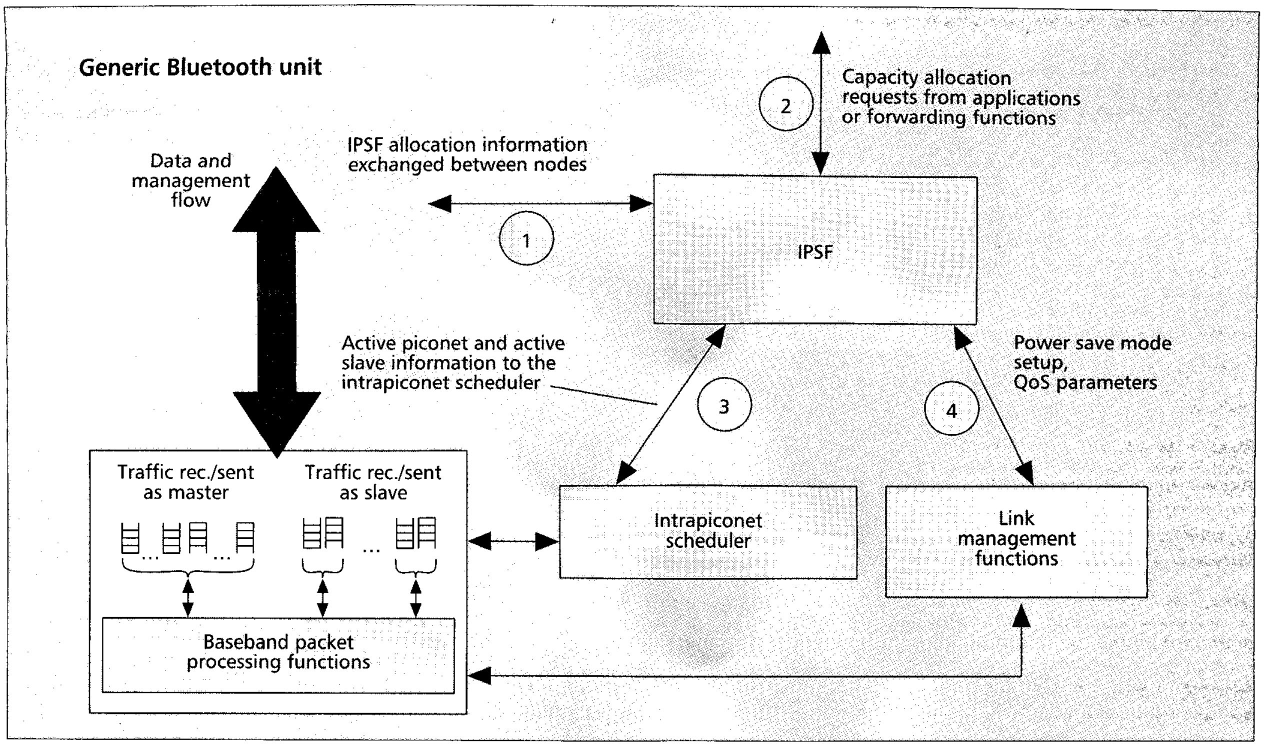
Scatternet Forming

In order to have an efficient infrastructure for IP networking on Bluetooth, piconets and scatternets must be able to adapt to the connectivity, traffic distribution, and node mobility in the network. This is mainly achieved by setting up new piconets or terminating others to attain the optimal scatternet topology. In this context, *optimal* refers to a scatternet that, for instance, yields minimum delay or maximum throughput. But it could also mean minimizing energy consumption in network nodes. Obviously, the scatternet routing protocol should have an impact on how the scatternet is formed. If a reactive ad hoc routing approach is used, the search for new paths (routes) will be one important way to adapt the scatternet topology to the current traffic distribution. Consequently, the routing protocol needs to consider what the optimal formation objective is (e.g., best throughput per energy unit).

Apart from (reactive) routing-oriented scatternet forming operations, formation functions are also needed to establish more generic connectivity in the scatternet in order to be able to find nodes searched for by the routing protocol. Typically these functions operate in the background to include new nodes or rearrange old ones in the scatternet and will use the Bluetooth connection establishment functions (INQUIRY and PAGE) on a periodic basis. In [16] one of the first known simulation studies on scatternet forming aspects was presented. The study showed that the total number of Bluetooth links and the overhead imposed in conjunction with interpiconet forwarding have a large impact on system performance.

Intra- and Interpiconet Scheduling

The master unit of a piconet controls the traffic within the piconet by means of polling the slaves according to any preferred algorithm (e.g., Round Robin), which determines how



■ **Figure 10.** *The functional architecture of the interpiconet scheduling function (IPSF) and its relation to other functions in a generic Bluetooth unit.*

Functional Architecture for Scatternet Scheduling

In its current version, the Bluetooth specification contains very limited information on interpiconet scheduling in general and does not mention how interpiconet scheduling should fit into the overall Bluetooth architecture. This section proposes a generic architecture for scheduling functions within a scatternet that will allow a wide variety of different scheduling algorithms to coexist. Figure 10 gives a proposed functional architecture of how an interpiconet scheduling function (IPSF) would interact with other functions in a generic Bluetooth unit. A subset of these functions is active in a Bluetooth unit, depending on the role the unit has in the scatternet.

The IPSF will have the following relation to other functions in the Bluetooth unit (the list relates to the corresponding numbers in Fig. 10):

1. The IPSF will need to exchange coordination information with a peer IPSF unit in some other node. This information may contain the points in time when a master and an interpiconet device will meet and may also involve an information exchange for negotiation of these points.
2. The IPSF may receive information on requirements posed by applications, such as delay constraints. For an interpiconet unit, the "duration of its visit" in a piconet and the "time elapsed" between such visits are parameters that determine the capacity and delay to the piconet. Thus, the IPSF may try to negotiate these parameters in accordance with the application requirements.
3. The interaction of the IPSF with the IRPS is only active when the device is a master. The IPSF should inform the IRPS about the presence of interpiconet slave devices so that the IRPS may appropriately poll them.
4. The Link Manager describes functions to handle establishment, security, and control of the radio link within a piconet. It also defines a set of power save modes (SNIFF, HOLD, PARK) that may be used by the IPSF to implement coordination between interpiconet units. Moreover, new releases of the Link Manager may host specialized interpiconet functions the IPSF may invoke.

Rendezvous Point IPS Algorithms

The functional architecture for scatternet scheduling outlined in the previous section allows for a wide variety of ways to do IPS. In this section we discuss a family of algorithms we refer to as *rendezvous point* algorithms. A rendezvous point is a slot at which a master and an interpiconet unit have decided to meet; that is, at this slot the master has agreed to address a packet to the interpiconet unit and the interpiconet unit has agreed to listen to the master.

Two major issues that need to be addressed by a rendezvous point interpiconet algorithm are:

- The rendezvous point (RP) issue: How do the master and slave decide on the RP, and how strict is the commitment to this RP? In a strict algorithm the master and slave units will always honor a RP. In fact, a wide variety of rules on honoring RPs may be defined. The distribution of RPs over time could be periodic, decided on each visit, or spread out in a pseudo-random sequence (known by both nodes). The delay property of an interpiconet algorithm will, to a great extent, depend on the time between mutually honored RPs.
- The rendezvous window (RW) issue: given that the master and slave units are both present at an RP, how much data will they be able to exchange? This depends on the duration of the period the units are mutually present and how much of this time is used to send data. In a strict algorithm, a time window, the RW, could be defined in which both master and gateway must be present and exchange data in every available slot. A relaxed algorithm would not require any of the master or slave units to stay any predetermined time. The achieved throughput for the master slave pair will depend on the duration of these RWs.

Based on how the issues listed above are approached, categories of IPS algorithms of the RP family can be identified; below a list of some of these categories are given. Note that the list is not exhaustive, but includes the most feasible ones:

- Honoring-Periodic Static-Window (HPSW): In this category, units always honor RPs. Also, RPs occur periodically, which gives a constant period between RPs for a particular pair of units, henceforth referred to as a *superframe*. The size of the RW for a piconet is static and remains the same throughout the duration of a connection.
- Honoring-Periodic Dynamic-Window (HPDW): This algorithm category always honors RPs that are distributed periodically. The RW is dynamic in order to adapt to both topology changes and traffic dynamics. For instance, if the RW is defined to be the time between RPs, the RW size will change if a new interpiconet node joins a piconet or an existing one leaves.
- Honoring-Random Static-Window (HRSW): The RPs are always honored but, unlike the previous schemes, they are spread out according to a pseudorandom pattern known to both the master and the interpiconet node. Due to the random spreading, RWs may end up quite close to each other. The size of the RW is in this case static and may give a limit on the number of piconets in which the interpiconet node may take part.
- Master-Honoring Dynamic-Window (MHDW): In this category the master unit always honors the RPs but the (slave) interpiconet unit may skip an RP in order to give priority to another piconet. The RPs may be distributed periodically as in HPSW or according to a pseudorandom sequence as in HRSW. Since the interpiconet unit may not honor RPs, the RW size may change to adapt to new traffic conditions or topology changes.

Generally, the always honoring (HPSW, HPDW, HRSW) IPS categories will give a more strict scheduling that on one hand enables better traffic delay guarantees but on the other hand is less flexible to change and less adaptive to traffic. When an HPDW algorithm wants to change the RW size, it may need to reallocate some RPs as well, which requires the exchange of control information, causing a penalty in terms of overhead. The non-always-honoring category (MHDW) will potentially result in less delay guarantees but will enable traffic adaptivity without requiring exchange of control information.

Simulation studies of the various IPS algorithms are ongoing within several research groups in both academia and industry (e.g., [18]), but so far no results have been presented in any comparative form to position one algorithm over another. However, several studies of IPS algorithms are expected to be published in the near future.

Conclusions

Ad hoc networks have mostly been used in the military sector, where being able to establish ad hoc communication is often a necessity. On the other hand, in the commercial sector, successful examples of ad hoc radio networks are few if any so far. However, instead of large-scale networks, small-scale personal area networks are emerging in response to the introduction of short-range radio technologies such as Bluetooth. Here, ease of use and flexibility are fueling the demand for ad hoc operation. In addition, a centralized network architecture would have serious problems trying to control all PAN devices. In particular, ad hoc Bluetooth networks — scatternets — will give rise to a whole new set of business and consumer applications for small, battery-driven user devices such as mobile phones, PDAs, and notebook computers. The combination of wide-area IP connectivity via 3G (mobile phone) access and personal area connectivity in the PAN presents new opportunities for the user on the go. End-to-end IP networking is a key component in this respect, providing the basis on which to develop applications for PAN products. Thus, the current development of IP support for Bluetooth networks in general and for 3G access of Bluetooth PAN in particular is crucial.

The current work in the PAN WG of the Bluetooth SIG focuses on developing IP support based on a network layer on Open Systems Interconnection (OSI) level 2, creating a broadcast segment similar to Ethernet. This will enable straightforward reuse of a number of IP-related protocols for configuration and address resolution (DHCP, ARP etc.) typically suited to LAN access environments, but also to standalone or interconnected PANs. Using the Bluetooth PAN protocol will, from the upper protocol levels, be very similar to connecting devices on an Ethernet segment. As an interesting parallel, one may consider how mobile phones would support laptops or other mobile devices to access the phone, or the Internet through the phone, if the phone were equipped with a physical Ethernet adapter.

The use of scatternets in Bluetooth networks introduces interpiconet gateways, which need to be scheduled between the piconets of which they are members. The choice of interpiconet scheduling scheme is crucial to the overall performance of the scatternet since it controls the efficiency of Bluetooth packet forwarding between piconets and, moreover, affects the performance within the piconets where the gateways reside. An overall architecture for scheduling in scatternets is outlined here to capture how the various Bluetooth functions will interact to achieve an efficient system. Moreover, a family of IPS algorithms, referred to as rendezvous point algorithms, was introduced as a feasible way to design IPS algorithms.

Bluetooth wireless technology in PANs will most likely change the way we handle and access information in the near future. A similar development during the past 10 years can be observed in the way the mobile phone has changed our behavior in terms of information vs. independence of location. The actual impact of Bluetooth in general, and its use in PANs in particular, is of course still mostly speculation. However, it is the first realistic attempt on a large scale to solve the last meter problem we often encounter: simply getting our personal devices to "talk" to each other without a hassle. Thus, this PAN, in combination with the emerging third-generation mobile systems, provides an intriguing entry for the Internet into more than one of our pockets.

References

[1] The Bluetooth Special Interest Group, "Baseband Specification," http://www.bluetooth.com

[2] Mobile Ad hoc Networks (MANET): http://www.ietf.org/html.charters/manet-charter.html, May 28, 2000, work in progress.

[3] IEEE Comp. Soc. LAN MAN Stds. Comm., "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," IEEE Std 802.11-1997.

[4] J. Khun-Jush *et al.*, "HIPERLAN Type 2 for Broadband Wireless Communication," *Ericsson Rev.*, vol. 77, no. 2000:2, pp. 108–19.

[5] J. Haartsen, "Bluetooth — The Universal Radio Interface for Ad Hoc, Wireless Connectivity," *Ericsson Rev.*, vol. 75, no. 1998:3, pp. 110–17.

[6] J. Hartsen *et al.*, "Bluetooth: Visions, Goals, and Architecture," *ACM Mobile Comp. Commun. Rev.*, vol. 4, no. 2, 1998, pp. 38–45.

[7] J. Haartsen, "The Bluetooth Radio System," *IEEE Pers. Commun.*, vol. 7, no. 1, Feb. 2000 , pp. 28–36.

[8] P. Johansson *et al.*, "Short-Range Radio-Based Ad hoc Networking: Performance and Properties," *ICC '99*, Vancouver, British Columbia, Canada, 1999.

[9] P. Bhagwat, "Bluetooth: Technology for Short-Range Wireless Apps," *IEEE Internet Comp.*, vol. 5, no. 3, May/June 2001; http://www.computer.org/internet/ic2001/w3toc.htm

[10] R. Droms, "RFC 2131: Dynamic Host Configuration Protocol," Mar. 1997; http://www.ietf.org/rfc/rfc2131.txt

[11] IETF, "Charter of the Zero Configuration Networking (zeroconf) WG," http://www.ietf.org/html.charters/zeroconf-charter.html

[12] C. E. Perkins, "Ad Hoc On Demand Distance Vector (AODV) Routing," Internet draft, draft-ietf-manet-aodv-02.txt, Nov. 1998, work in progress.
[13] J. Broch, D. B. Johnsson, and D. A. Maltz, "The Dynamic Source Routing Protocol for Mobile Ad hoc networks," Internet Draft, draft-ietf-manet-dsr-00.txt, Mar. 1998, work in progress.
[14] C. E. Perkins and P. Bhagwat, "Highly Dynamic Destination-Sequenced Distance-Vector routing (DSDV) for Mobile Computers," *Proc. SIGCOM '94 Conf. Commun. Architecture, Protocols and Apps.*, Aug. 1994, pp. 234–44; rev. version: http://www.cs.umd.edu/projects/mcml/papers/Sigcomm94.ps
[15] P. Johansson *et al.*, "Scenario-Based Performance Analysis of Routing Protocols for Mobile Ad Hoc Networks," *Proc. 5th Annual Int'l. Conf. Mobile Comp. and Networking*, Aug. 1999.
[16] G. Miklos *et al.*, "Performance Aspects of Bluetooth Scatternet Formation," *MobiHoc 2000*, Boston, MA, Aug. 11, 2000.
[17] N. Johansson, U. Korner, and P. Johansson, "Performance Evaluation of Scheduling Algorithms for Bluetooth," *Broadband Communications: Convergence of Network Technologies*, Danny H. K. Tsang and Paul J. Kuhn, Eds., Kluwer, 2000, pp. 139–50.
[18] Wireless Adaptive Mobility (WAM) Lab, http://www.cs.ucla.edu/NRL/wireless

Additional Reading

[1] S. Cheshire, "Dynamic Configuration of IPv4 link-local addresses," Internet draft, Oct. 8, 2000, http://www.ietf.org/internet-drafts/draft-ietf-zeroconf-ipv4-linklocal-00.txt

Biographies

PER JOHANSSON (Per.Johansson@ericsson.com,) Tekn. Lic., is a senior researcher at Ericsson Corporate Research, Stockholm, Sweden. He joined Ericsson in 1992 to work in the areas of traffic management and performance analysis of ATM networks. He later moved into research on wireless systems, where his research has focused on ad hoc networks and, in particular, on Bluetooth ad hoc networking. Since 1998 he has managed a research team at Ericsson Research that focuses on IP networking aspects on Bluetooth. Currently, he is a visiting researcher at the Wireless Adaptive Mobility Lab at University of California, Los Angeles (UCLA), where he takes an active part in the Bluetooth ad hoc networking research.

MANTHOS I. KAZANTZIDIS (kazantz@cs.ucla.edu) received his diploma degree in computer engineering and Informatics in 1995 from the University of Patras, Greece. He received his M.S. in computer science in 1998 from University of California, Los Angeles and is currently a Ph.D. candidate at UCLA. His research focuses on adaptive multimedia over networks with wireless and mobile links and is a member of the Wireless Adaptive Mobility Lab at UCLA.

ROHIT KAPOOR (rohitk@cs.ucla.edu) received his B.E. in computer science in 1999 from the University of Roorkee, India. He is currently a Ph.D. candidate at the UCLA. His research focuses on performance issues in Bluetooth piconets and scatternets. He is a member of the Network Research Lab at UCLA.

MARIO GERLA (gerla@cs.ucla.edu) is a professor in the computer science department at UCLA. He received his graduate degree in engineering from the Politecnico di Milano in 1966, and his M.S. and Ph.D. degrees in engineering from UCLA in 1970 and 1973, respectively. He joined the faculty of the UCLA Computer Science Department in 1977. His current research is in the area of analysis, design and control of communications networks. Ongoing projects include the design and evaluation of QoS routing and multicast algorithms for IP domains, the design and evaluation of all-optical network topologies and access protocols, the design of wireless mobile, multimedia networks for mobile computing applications, and the development of measurement methods and tools for evaluating the performance of high-speed networks and applications.