

Specknets: New Challenges for Wireless Communication Protocols

K.J Wong and D.K Arvind

*Research Consortium in Speckled Computing
School of Informatics, University of Edinburgh
Mayfield Road, Edinburgh EH9 3JZ, Scotland, U.K.
k.j.wong@sms.ed.ac.uk | dka@inf.ed.ac.uk*

Abstract

Speckled Computing [1] is an emerging technology in which data will be sensed and processed in small (around 5X5 sq. millimeter) semiconductor grains called Specks. A dense and non-static wireless network of thousands of these specks – called a Specknet - will collaborate to extract information from the data. Speckled Computing demands requirements of wireless communications in excess of typical mobile ad-hoc networks (MANET) and sensor networks. This paper presents new challenges for the design of communication protocols for specknets, in which each speck is modest in terms of energy, storage and computational resources

1. Introduction

A Speck integrates three capabilities: sensing, processing and wireless networking in a small (around 5X5 sq. millimeter) semiconductor grain. Specks are intended to be autonomous, each with its own renewable energy source, and can be mobile if needed. Thousands of Specks will collaborate as programmable computational networks called Specknets. Computing with Specknets, or Speckled Computing, will enable linkages between the physical and digital worlds with a finer degree of spatial resolution than hitherto possible. Indeed, Specknets are intended to be platforms for truly ubiquitous or pervasive computing applications.

This paper first highlights the differences between specknets and traditional sensor networks. Next, the design issues for the three layers in the communication protocol - the physical, medium access control (MAC) and network layers - are discussed. Finally, preliminary experimental results and ideas for future work are presented.

2. How is a Specknet different from typical sensor networks?

We next highlight the main differences between specknets and traditional sensor networks from the point of view of networking and communication.

a) *Data-centric/Program-centric networks* - The main aim of any sensor network is to sense data and transmit it back to a central hub or sink node, where it is processed and the extracted information is stored and acted upon. In contrast, a specknet does away with this centralised processing hub; each programmable speck is endowed with the capability to extract information locally in collaboration with its neighbours and act upon the information via embedded actuators in each speck. Specknets disseminate computational tasks as well as information within the network. Specknets are therefore program-centric, whereas sensor networks are typically data-centric.

b) *Range of communication* - Sensor networks are usually deployed over a wide area where each node is capable of transmitting to distances ranging from a few meters to a few kilometers. In the case of specknets, each speck is designed to transmit within a range of tens of centimeters. Specknets are in essence highly dense networks with a deployment area of a few square meters whereas a sensor network is a sparse network covering areas of up to tens of square meters or even a few square kilometers. The difference in the communication model is due to different costs involved in communication. In a typical sensor network, the energy cost for data transmission is higher than for data reception, whereas in a specknet the reverse is true as the ranges are relatively much smaller.

c) *Mobility Model* - Communication protocols designed for specknets have to assume that all specks are non-static by default. The reason for this is

twofold: specks are intended to address a new class of applications in pervasive and ubiquitous computing where mobility is an inherent feature; and given the scale of specks, even small movements could affect the communication between specks. In contrast, sensor networks are mostly treated as static networks.

d) *Data transfer model* – In sensor networks, nodes are classified as source nodes and sink nodes. Source nodes are ones that sense the environment periodically and the sensed data is routed to the sink nodes. In a specknet by contrast, a peer-to-peer model is used with no distinction between source and sink nodes as each Speck can dynamically be tasked to carry out different operations at various times as dictated by needs. Data transfers in a specknet are aperiodic as communication needs are highly dependent on the processing that has to be carried out.

3. The Physical layer

Specks only communicate wirelessly (radio and free-space optics are the options being considered), without cumbersome wires, to provide ease and flexibility of deployment. However, wireless communications present massive challenges given speck's extremely small form factor.

Radio communication systems can be currently classified broadly as either narrowband or wideband systems. Narrowband communication depends on sinusoidal waves transmitted continuously at some known frequency (the carrier frequency), and the data to be transmitted is modulated onto this carrier frequency. Wideband radio, on the other hand, does not require a constant carrier frequency; but instead data is transmitted by sending "impulses" of radio frequency signals that occupy very high bandwidth. Ultra Wide Band (UWB) [2] operating at a radio frequency band of between 3.1GHz and 10GHz is more suited for specks due to the simpler transceiver circuitry (although the complexity is transferred to the signal processing circuitry), possible lower power consumption and the ability to communicate using a smaller antenna.

Free-space optical communication uses light pulses to transmit data. Theoretically, the use of light as the communication medium provides for higher data rates, as light has a shorter wavelength than radio. The disadvantage however is that the medium is susceptible to obstruction: whereas radio can communicate through obstructions such as walls, optical signals would be reflected, diffused or adsorbed.

3.1. Preliminary experiment

To analyse the impact that the different shapes of areas of communication coverage provided by different mediums could have on networking, a simulation program was designed on Ns-2 [3] to compare omni-directional radio [4] communications with unidirectional infrared systems [5] in a network covering 300mm x 300mm with nodes moving using a waypoint walk. To be fair, the radio is given a range of 40mm whereas infrared has a range of 80mm with a 90 degree coverage angle. Thus, both media have equal coverage area of 5026mm². The simulated time is 30 seconds.

Two interesting attributes were measured in the simulations. The *number of link changes* is the number of changes in the direct link (one-hop) communication between any two nodes in the network and the *number of route changes* is the number of changes in the routes whenever a discovered shortest route between any node pairs in the network is broken.

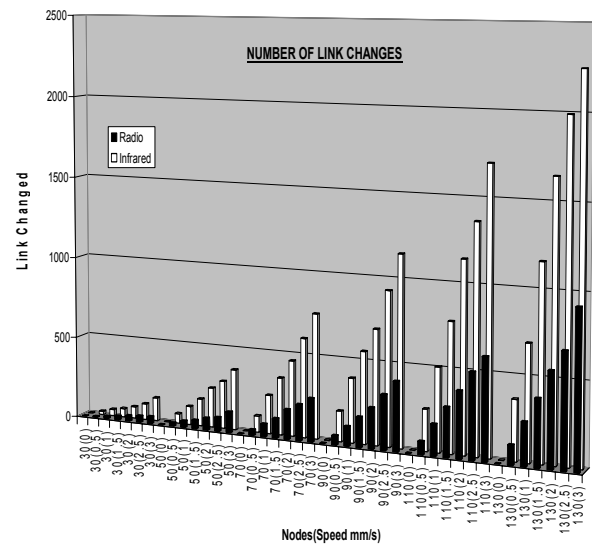


Figure 1. The number of link changes over different network densities and nodal speed

Figure 1 shows the number of link changes over the different simulation scenarios. When the network is static, no link changes would ever occur for all densities. As the speed of the nodes increases, the number of link changes would occur more frequently as expected due to the higher number of connections and disconnections. It can be observed, that the number of link changes that occur over infrared links is almost three times greater than in the case of radio links, although both cover the same area of communication.

The number of link changes that occurs in a network has a huge impact on the performance of any network routing algorithm. An example of such an impact can be demonstrated using routes calculated by the shortest path algorithm and looking at the number of route changes that occur, as shown in Figure 2.

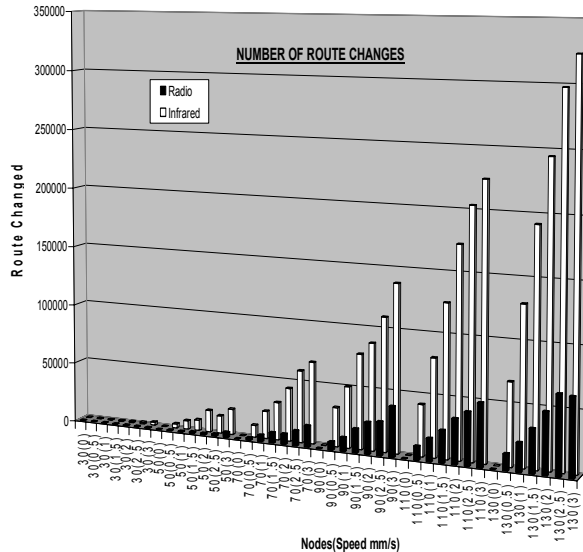


Figure 2. The number of route changes over different network densities and nodal speeds

The simulation results demonstrate that infrared systems would incur more route changes than radio systems, resulting in greater overheads in the network layer as routes are often invalidated quickly. Therefore, omni-directional coverage provided by radio communications is preferred over infrared.

3.2. Future work on the physical layer

Radio seems more suited for inter-speck communication based on the network connectivity properties. In future experiments, other attributes such as power consumption would be investigated for the two media.

4. The MAC layer

The Medium Access Control (MAC) layer manages access to the physical communication channel. It is responsible for ensuring that a given quality of service is maintained for throughput, reliability, and robustness. Given that the specks will have limited energy storage, the most important function of the MAC layer is to maintain communication between specks for as long as possible.

One way of achieving this goal is by trading power consumption with data latency by using duty-cycling of the communication channel. Several power-aware MAC schemes had been designed for MANET and sensor networks using such an approach. TDMA-based [6] protocols divides time into small slots and nodes are assigned slots in which they can communicate. These protocols require individual nodes to keep track of their neighbours' slot assignment in order to communicate in a peer-to-peer fashion, and therefore require significant memory resources. The complexity of assigning slots between nodes without base-stations or cluster-heads would also render TDMA-based protocols unsuitable for specks.

Other MAC schemes, such as PRAMAS [7], use a separate radio channel to achieve power savings. However, this requires additional hardware circuitry. Given the limited space available on each speck, such an approach would not be feasible. MAC protocols, such as S-MAC [8], have been created for sensor networks and require "loose" synchronisation between nodes. Just as in TDMA-based protocols, time is divided into large frames and each frame is divided into active and sleeping parts. Nodes can only communicate during their active parts. In order for S-MAC to operate optimally, nodes in the network would have to be either synchronised to one schedule so that all nodes wake up at the same time, or nodes would have to keep track of their neighbours' schedule. The complexity of synchronisation, either global or local within clusters, not only creates overheads (via the SYNC packets that are used in S-MAC, or the need to keep track of its neighbours' schedule) but also makes the algorithm unsuitable for dense networks in which the nodes are also mobile, as in the case of specknets.

The ideal MAC protocol for specknets should allow each speck to duty-cycle its radio transceiver and still communicate with others specks without the need for any synchronisation or coordination. The ability to communicate without synchronisation would also allow specknets to be scalable and for such a protocol to function effectively with mobile specks.

4.1. Preliminary experiment

For specks to communicate effectively using low duty-cycle rates and in an unsynchronised manner, a simple MAC – SimMAC - was designed based on the principle of using redundant retransmissions to remove the need for any synchronisation within the network. SimMAC therefore allows specknets to be

scaled and communicate in a peer-to-peer fashion in the face of node mobility.

Figure 3 shows the consumption of current in milliamperes of running SimMAC on a speck prototype – the ProSpeckz [1]. Each speck in the specknet would turn on its receiver at a fixed duty cycle, D_C , and would retransmit intended messages x times at intervals of I_C using the following formula:

$$D_C = P_R / P_{DC}$$

$$I_C < P_R - (M_{CD} * 2)$$

$$x = P_{DC} / I_C$$

where P_R is the user-defined period when the receiver is turned on, P_{DC} is the user-defined duty cycle period and M_{CD} is the maximum clock skew of the embedded oscillator on the specks.

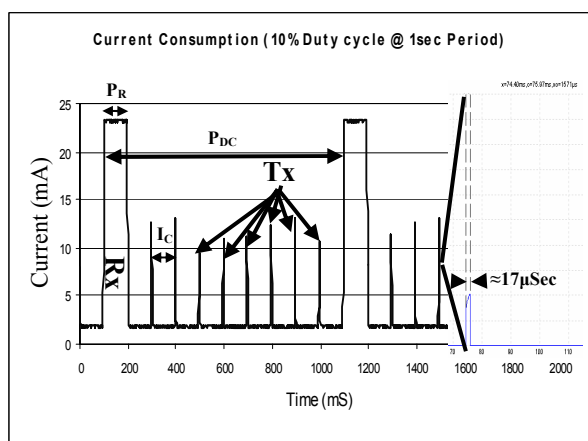


Figure 3. Current consumption of SimMAC running on ProSpeckz

The overhead required for retransmission is a function of P_R and P_{DC} , which would be reduced with a longer P_R and P_{DC} . Figure 4 shows the duty-cycling rate of the radio transceiver as compared to the intended duty-cycling rate, D_C . It is observed that when P_R is four seconds, the overhead induced by retransmission using SimMAC is negligible when the intended duty-cycle of the receiver is more than 5%.

4.2. Future work on the MAC layer

In the current version of SimMAC all specks are required to operate at a fixed duty cycling rate. However, this would present significant overheads when there is little or no traffic in the network. Therefore the ability for each speck to dynamically change its duty cycling rate based on local traffic requirements would be explored in the future. Different specks can then operate at different duty-cycling rates by keeping track of the values of P_R and P_{DC} of their neighbours.

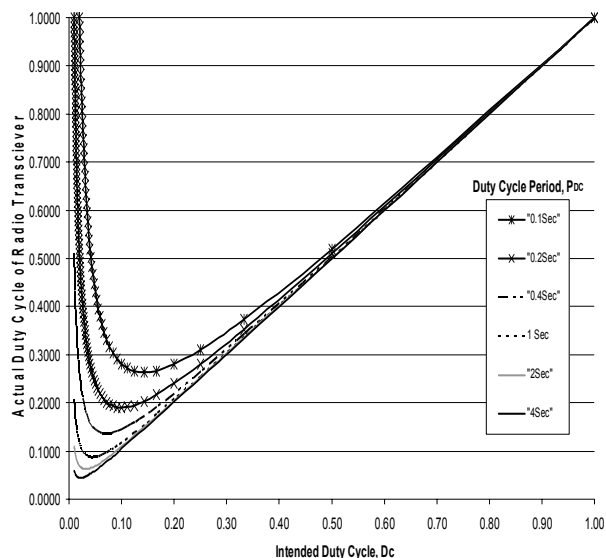


Figure 4. Comparison of actual versus intended duty-cycling using SimMAC

5. Network layer considerations and constraints

Given their small sizes, specks will be space constrained and only 2 Kilobyte of random-access memory (RAM) has been allocated for networking. This includes the memory space needed to buffer incoming/outgoing messages (the interface queues), the routing tables, and the one-hop neighbour list. This memory constraint and the properties of a specknet, as mentioned in Section 2, render many existing MANET and sensor network routing protocols unsuitable for specks. These algorithms can be grouped into the following categories:

a) *Proactive (table-driven) algorithms* – These algorithms [9] constantly maintain routing table(s) on each node by storing the route to any destination node in the network. The need to maintain these tables means that these algorithms are constantly exchanging data even though some routes would never be used. This overhead and the need to maintain huge network tables make such algorithms unsuitable for specknets, especially given their very limited memory.

b) *Reactive (on-demand) algorithms* – These algorithms [9] would discover routes only when one is needed. Even though these algorithms do not suffer from overheads caused by maintaining redundant routes, the memory required to perform route discovery is still quite considerable as the ones which are discovered need to be stored either in a cache or in the route reply packets.

c) *Hybrid (proactive/reactive) algorithms* – An example of a hybrid algorithm is ZRP [10] which aims to combine the advantages of both proactive and reactive routing. Each node would use proactive routing for routing packets within a certain hop distance and revert to reactive routing for destinations further away. However, as the route cache used for performing reactive routing is still limited by the memory available on each node, this algorithm would be an unsuitable choice for specknets.

d) *Location-based algorithms* – As the name implies, location-based algorithms [11] make use of location information to assist in route discovery. The primary problem in the case of specknets is the requirement to determine location information. This is difficult given the small size of specks, low memory and power resources, and the lack of positioning systems such as GPS. Although algorithms using triangulation or graph prediction could possibly determine relative location or logical location of specks, these would be computationally expensive and result in increases to the network overheads significantly. Furthermore, there is a need to keep location information about the destination and neighbouring nodes for this type of routing algorithms, which further increases the memory requirement for each node.

d) *Data-centric algorithms* – Directed diffusion [12] is a data-centric algorithm which is typically used in sensor networks. Data generated by sensor nodes is named by attribute-value pairs and routes are established by positive and negative reinforcement. Direct diffusion would suit routing in sensor networks where the sensed data is the source for all interaction and the data items can be easily quantified and identified as routing tables are looked up by interests instead of node addresses. However, in the case of highly reprogrammable computational networks such as specknets, the quantification and identification of data items or interest may not be straightforward or even possible due to the sheer numbers of data item types and the flexibility required. Also, the latency and the reliance on broadcasting would create huge initial overheads that may not be supported by the limited memory on each speck.

5.1. Preliminary experiment

In order to demonstrate the importance of memory resource requirement on routing algorithms, a simulation was designed in Ns-2 to determine the amount of memory required for the interface queues to buffer incoming packets, using DSR [9] as an example routing algorithm.

The simulation scenario is based on a network of 100 wireless mobile nodes in an area of 200 x 200mm. Nodes would pause for 1 second on reaching a random destination before selecting a new destination using a random waypoint walk. The maximum speed that the nodes will travel at is 1 millimeter per second. The nodes will communicate using the radio with a range of 40mm and a data rate of 2Mbps. At any time, there could be a maximum of 50 pairs of nodes communicating with each other in a peer-to-peer fashion with packet sizes of 32 bytes. Figure 5 shows the length of the interface queue for each node required for different transmission rates for a simulated time of 60 seconds.

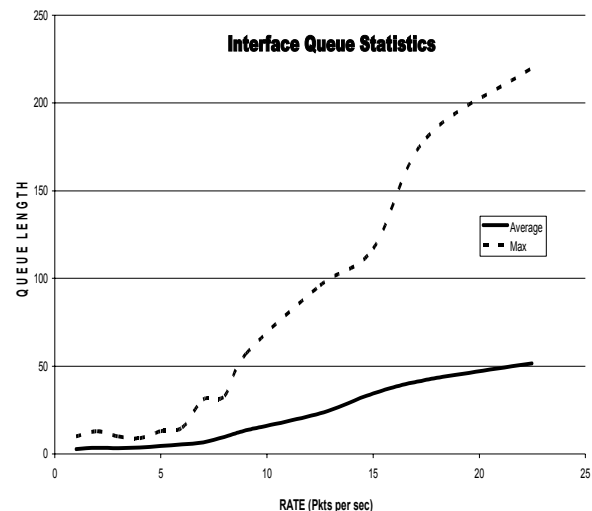


Figure 5. Simulation results showing the average and maximum interface queue length required

It was observed that in order to provide optimal performance for DSR, the memory needed for interface queues to buffer data packet could be quite large compared to the memory that is available in the specks. One point to note is that at certain hotspots in the network, there would be a requirement for a larger interface queue to sustain optimal performance as shown in the maximum queue size used by the simulation in Fig 5. Given that only 2 Kilobytes of memory is available on each speck, this could pose a possible problem when implementing DSR on specks. The same problem may occur for other “on-demand” routing algorithms, as all of them require global flooding to discover a route to a destination node. The simulation demonstrates the memory constraints that have to be considered when designing the network layer protocol for specknets.

5.2. Future work on the network layer

A novel protocol - Speckzone Based Routing Protocol (SBaRP) - is currently being designed to enable specks to route data under stringent memory requirements. SBaRP is a hybrid routing algorithm that utilises the advantages of both proactive and reactive protocols to enable routing in specknets. It is similar to ZRP, but unlike ZRP where each node has its own zone defined and maintained based on hop distances, SBaRP defines Speckzones by maintaining the number of nodes within each zone to a certain bound. Thus each Speckzone is shared by a collection of Specks where each Speck can only belong to one Speckzone. Routing within the Speckzone is done proactively whereas routing to Specks in other Speckzones would be done reactively.

As the number of Specks within each Speckzone is bounded, the routing table for proactive routing is also bounded and hence the memory space needed to store the proactive routing table can be defined. For communications between each SpeckZone, the routes which are reactively discovered are stored in distributed routing tables maintained by all Specks within the Speckzone.

SBaRP is currently being simulated on Ns-2 and is being implemented on the ProSpeckz. Further experiments and simulations would then be carried out to determine the performance of utilising the distributed memory model for the routing tables as proposed in SBaRP.

6. Conclusions

Specknets is currently in the early stages of its development, but a number of new challenges have been identified for communication and networking. These include the requirement for the wireless medium to be designed for specks with minute footprint at the physical layer, the need for an unsynchronised power-aware MAC layer and the ability for data routing to be carried out by the network layer in face of both extremely limited memory space and power onboard each speck. These constraints should present the research community with new challenges to develop novel designs and protocols which is unprecedented in the networking requirements for sensor networks and MANET.

7. References

- [1] D.K. Arvind, K.J.Wong, "Speckled Computing: Disruptive Technology for Networked Information Appliances", Proc. IEEE International Symposium on Consumer Electronics (ISCE'04), September 2004, pp. 219-223
- [2] Barrett, T.W., "History of UltraWideBand (UWB) Radar & Communications: Pioneers and Innovators", Proc. Progress in Electromagnetics Symposium 2000 (PIERS2000), Cambridge, MA, July 2000
- [3] Ns-2 simulation, www.isi.edu/nsnam/
- [4] IEEE Standard for Information Technology, IEEE Std 802.15.4-2003, 2003
- [5] The Infrared Data Association, www.IrDA.org
- [6] P. Havinga and G. Smit, "Energy-efficient TDMA medium access control protocol scheduling", Proc. Asian International Mobile Computing Conference (AMOC 2000), November 2000, pp. 1-9
- [7] S. Singh and C. Raghavendra, "PAMAS: Power aware multi-access protocol with signalling for ad hoc networks", ACM SIGCOMM Computer Communication Review, July 1998, pp. 28(3):5-26
- [8] W. Ye, J. Heidemann, and D. Estrin, "An energy-efficient MAC protocol for wireless sensor networks", Proc. 21st Conference of the IEEE Computer and Communications Societies (INFOCOM), volume 3, June 2002, pp. 1567-1576
- [9] E. Royer, C.K Toh, "A Review of Current Routing Protocols for Mobile Ad-Hoc Networks", IEEE Personal Communications, April 1999, pp. 46-55
- [10] Z.J. Haas and M.R. Pearlman, "The Zone Routing Protocol (ZRP) for Ad Hoc Networks", Internet Draft, draft-ietf-manet-zone-zrp-02.txt, June 1999
- [11] Y.C Tseng, C.S Hsu, "Location-Aware Routing and Applications of Mobile Ad Hoc Networks," The Handbook of Ad Hoc Wireless Networks, CRC Press, 2003, Chpt. 18
- [12] C. Intanagonwiwat, R. Gocinda, D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks", in Proc. Of the Sixth Annual International Conference on Mobile Computing and Networks (MobiCOM 2000), August 2000