Social-Aware Routing for Wireless Mesh Networks

Shadi Saleh Basurra[1], Yusheng Ji[2], Marina De Vos[1], Julian Padget[1], Tim Lewis[3], Simon Armour [4]
[1]Dept. of Computer Science, University of Bath, UK
[2] Information Systems Architecture Science Research Division, National Institute of Informatics (NII), Tokyo, Japan
[3]Toshiba Research Europe Ltd, Bristol, UK
[4]Dept. of Electrical and Electronic Engineering, Bristol University, UK
Email: {S.S.A.Basurra, mdv, jap}@cs.bath.ac.uk, Tim.Lewis@toshiba-trel.com, kei@nii.ac.jp, Simon.Armour@bristol.ac.uk

***Abstract*—In wireless mesh networks (WMN), most routing algorithms use broadcast at some stage of the path discovery process, thereby taking up a large proportion of the network bandwidth. Intelligent rebroadcast algorithms aim to reduce this overhead by calculating the usefulness of a rebroadcast and the likelihood of collisions. Unfortunately, this introduces latency and breaks the rebroadcast chain, resulting in reduced reachability. In this paper we present our Social-aware Routing Protocol with Parallel Collision Guidance Broadcasting for WMN (SCG). It reduces rebroadcasting without a loss in reachability and without a significant increase in latency. Our claims are validated through simulations comparing our algorithm with existing protocols.**

I. Introduction

Wireless Mesh Networks (WMN) [4] consist of heterogeneous wireless devices acting as mesh routers (MR) or mesh clients (MC). The former, mainly stationary nodes with unlimited power supply, are responsible for establishing a network backbone of self-configuring and self-healing links. They are usually equipped with multiple wireless interfaces to provide connectivity to MCs and neighbouring MRs. The MCs have a single interface and establish ad-hoc connections with remote nodes.

WMNs are of academic and industrial interest due to their rapid deployability, scalability and low cost. However, WMNs' main bottleneck is path discovery for routing. With nodes joining, leaving and moving in the network, the paths need constant recalculation. Most existing routing protocols use broadcast at some stage to find routes, taking up a large proportion of bandwidth, creating loops and possibly introducing network delay [1].

In this paper we propose the Social-aware Routing Protocol with Parallel Collision Guidance Broadcasting for WMN (SCG). Routing is established using a new parallel collision guidance broadcasting technique [2], initiated by the network MRs. MRs allocate social tie labels to MC nodes to divide the network into communities. These labels denote long term social relationships that can be identified from contact frequency and duration between nodes. The concept of associating social ties to mobile nodes is not new in itself, but the purpose to which we put them here is novel. Pan Hui et al [5], used a labelling concept to design a social forwarding schemes in Delay-Tolerant Networking (DTN) to determine the node's future mobility pattern. The authors demonstrated a significant improvement in forwarding performance in terms of both delivery ratio and cost. We use social ties and their footprints as a mechanism to minimise control overheads in WMN. This is described in detail in Section III-D. Through simulation we show that our protocol provides fast routing with in most cases a reduction in control overheads.

Acknowledgements: This work is partially supported by Toshiba Research Europe Ltd, the Yemen Ministry of Higher Education and Scientific Research (MoHESR), Great Western Research (UK) and the National Institute of Informatics (NII) Japan.

II. Background and related work

Most routing protocols designed for mobile ad hoc networks (MANET) can also be applied to WMN because they have many features in common e.g. node mobility and a dynamic connection model etc. Yet, these protocols do not take full advantage of MR capabilities. MRs are typically more reliable and have more processing power and energy capacity and are static, whereas MCs do minimal routing to save energy [4]. Therefore, any routing algorithm for WMN needs to account for the capability and capacity differences of MRs and MCs. Two approaches that extend existing ad hoc routing protocols to allow for this are Multi-Radio AODV [11] and field-based routing (FBR) [12].

Most ad-hoc routing algorithms regardless of their degree of compatibility with WMN networks, use broadcasting at some stage to establish routes. Pure flooding guarantees high reachability and good routing time latency in low density networks. However, it also uses a lot of network capacity through redundant rebroadcasts. Smart routing algorithms aim to reduce the number of redundant rebroadcasts, but taken too far this may break the rebroadcast chain and critical intermediate nodes do not receive rebroadcasts, resulting in reduced reachability [1].

Many schemes, e.g. OLSR[6] and FBR[12], have been proposed for nodes to estimate neighbourhood density and trade off low broadcast redundancy with reachability, which in turn leads to the best possible network throughput, reachability levels and low broadcast latency. Most routing protocols designed for WMN see lowering broadcasting latency as a result of efficient broadcasting [13], but not as a protocol design objective. Our view is that both can be reduced by addressing them in the protocol design phase, especially in WMN networks with highly mobile MCs, where communications between nodes are short and moderately frequent.

978-1-4673-1881-5/12/$31.00 ©2012 IEEE

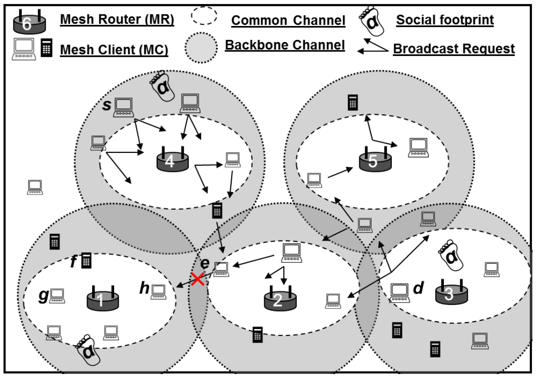

Fig. 1. Basic overview of the network structure

To validate our protocol we compare it to four popular existing protocols: Temporally-ordered routing algorithm (TORA), Dynamic Source Routing (DSR), Optimized Link State Routing Protocol (OLSR) and and Ad hoc on-demand distance vector (AODV), which we now briefly describe.

TORA's [9] main idea is to discover and build a directed acyclic graph (DAG) from source to destination, with edges indicating direct communication. TORA promises distributed loop-free and multipath routing. But, node convergence time can be lengthy and can produce temporarily invalid routes which further delays routing. TORA is designed to operate on top of the Internet MANET Encapsulation Protocol (IMEP)[3]. IMEP incorporates many ad-hoc network mechanisms by encapsulating routing packets into larger IMEP packets to improve the network performance.

The DSR protocol [7] uses broadcasting combined with local caches of existing routes. A major advantage in DSR is that route discovery and maintenance are performed on-demand since DSR does not broadcast hello message to sense neighbours. However since all packets carry the road map to their destination, this can cause delay and consume significant bandwidth in a high diameter network.

OLSR[6] is a point-to-point link-state routing protocol, with lower overheads than pure link-state protocols. This is achieved by selecting and using a subset of neighbour nodes, called multipoint relays (MPRs), to rebroadcast packets. However, MPR redundancy is needed to obtain reasonable synchronisation between link-state databases to prevent loops, generating overheads that reduce overall network performance.

AODV [10] reduces routing overheads by creating routes on-demand unlike table-driven protocols that keep lists of routes, such as TORA and OLSR. AODV can cope with dynamic networks, but frequent broadcasting can consume bandwidth and cause delay.

III. The SCG network routing protocol

Our protocol assumes that MRs are equipped with multiple wireless interfaces. One of the MR's non-overlapping radio channels is used to establish connectivity for self-configuring, self-healing links between MRs, forming the backbone channel. The other channels provide connectivity for the MCs and neighbouring MRs. Using the backbone wireless links, MRs pro-actively unicast/multicast MC IP lists with their associated MR addresses to maintain a partial network view. This partial network knowledge at MRs is based on the MCs' social ties (explained in Section III-C) and limits MRs' proactive behaviour, hence reducing overheads. The frequency of proactive data exchange is further constrained to only take place when a MC joins or leaves a MR.

A. Route discovery in SCG

The basic idea behind SCG is that the source MR sends out Path Discovery Commands (PDCs) to the source and destination MCs which then fire off Path Discovery Broadcasts (PDBs) at the same time to find one another. As these messages propagate through the network, they mark the path they take so that when they collide, they can return to the originating node establishing the forwarding route as they go. To explain this more fully, consider the scenario in Fig 1. A MC_s in MR_4 wants to send data to MC_d in MR_3. If MC_s does not have a fresh route to MC_d, it contacts MR_4 to request the start of a path discovery process. Thanks to the event-driven data exchange between MRs, which is performed efficiently by the footprint mechanism (explained in Section III-C), MR_4 knows that MC_d is hosted by MR_3, and will also know the estimated end-to-end delay time to reach MR_3. This helps MR_4 to calculate the approximate timing to send the PDCs via the backbone and common wireless channels to ensure they arrive at MC_s and MC_d at the same time. The PDC message contains (i) the target IP address, (ii) the MRTL value (see III-B), (iii) the broadcast sequence number and (iv) the broadcast initiation time. Upon PDC receipt, MC_s and MC_d broadcast PDB messages in order to find one another. PDB rebroadcast continues at intermediate nodes (MRs and MCs via the common communication channel) until a positive routing collision occurs, that is when an intermediate node receives PDBs generated from both ends with identical broadcast sequence number and the source IP address of one PDB is the same as the destination IP of the other. This "positive" routing collision occurrs at MC_e in Fig 1. If a unidirectional route is required, only one Route REPly (RREP) message is generated and returns via intermediate nodes to MC_s to set half of the newly discovered forwarding path $\text{MC}_s \rightarrow \text{MC}_e$, while the forward path on the nodes $\text{MC}_e \rightarrow \text{MC}_d$ has been set by PDBs generated from MC_d. On the other hand, for a bidirectional route, two RREPs are generated and forwarded to MC_s and MC_d by the node at which the positive collision takes place.

B. Reduction of redundant re-broadcasts using MRTL

To further improve the SCG route discovery mechanism between distant nodes of two different MRs, a strategy similar to TTL in AODV [10] is used, but instead of hop numbers, we use a MR-To-Live (MRTL) counter. MRTL is the number of MRs a broadcast packet needs to cross before it is discarded, when the MRTL value is zero. MCs act as defence walls to protect their MR zones from unnecessary rebroadcasting. The MRTL value is maintained during the proactive data exchange between MR nodes, as MRs can readily identify the number of zones between themselves in the network.

C. Reducing MR backbone control overhead

Unlike routing algorithms that treat MRs as gateways to forward traffic between nodes, SCG uses MRs to coordinate the parallel collision guidance broadcasting, and encourages ad-hoc multihop/multichannel path set-up. Global knowledge of a MC's current location is required to implement the parallel collision guidance broadcasting. This is susceptible to scalability problems in large and highly mobile networks. To resolve this, we use a node's social ties to localize knowledge and distribute it across network nodes. This is achieved by dividing the network into smaller communities based on frequent interactions. Since SCG depends on MRs in coordinating the initiation of path discovery operations for MCs, MRs can easily monitor and detect highly interacting network MCs and label them with a common social tie number. This can also be determined by the MCs themselves.

A social-tie relationship is a unique label that groups nodes with common behaviour. This label represents a long-term relationship, and nodes can have multiple social ties. Any changes that occur to social tie members are not circulated through the entire network but just to the MRs of nodes involved. Hence, there are no unnecessary data transfers to uninterested groups, and a single point of failure associated with centralised knowledge is avoided.

Any MR can assign a social tie number to MCs exhibiting high communication frequency in order to form a social group. Hence, in the event that a MC joins or leaves a MR, only those MRs that host the social group members receive the event notification. ACK messages are used to provide reliable delivery of these notifications. To implement the social tie, we use the concept of a social footprint.

A social footprint represents the social ties of a group of MCs, and consists of the addresses of the MRs that host the social tie group members. Footprint instances of a particular social tie help the group members to identify all MRs that host them. Only one footprint instance is required at each MR to inform one or more MCs that belong to the same social tie group. All footprint instances that represent a social tie, distributed across the network MRs, must be consistent at all times i.e. they must hold identical content for the protocol to perform efficiently with minimum redundant rebroadcasts.

Footprints are made volatile by associating a timer with each. When the timer expires, the associated footprint is deleted. A footprint timer can also be reset, such as when the host MR receives hello from the MCs that belong to the same social tie as the footprint.

D. Illustrating social tie and footprint usage

Let us consider the following illustrative scenario of the usage of footprints. We use FP_n to mean a footprint of social tie n. Using the mesh network Fig 1, MC_s in MR_4, MC_g and MC_f in MR_1 and MC_d in MR_3 have been recognised and classified by these MRs to share a social tie α due to frequent communication. If the link between MC_s and MR_4 fails due to MC_s moving or channel interferences, etc., MR_4 is triggered to multicast a notification message to MR_3 and MR_1 about this change. Note that $\mathrm{MR}_{2,5}$, do not receive/process this update message because they do not host any MCs with the social tie α. In this case, the traffic is reduced by about 40% compared to that for pure flooding.

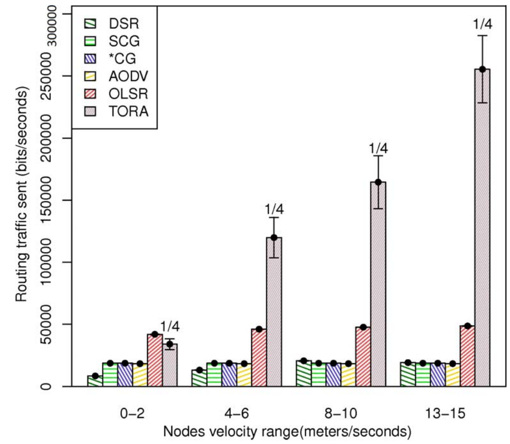

Fig. 2. Routing Traffic Sent (bits/seconds)

Since MC_s was the last MC to benefit from FP_α stored at MR_4, FP_α never gets refreshed and is eventually deleted from MR_4. During the absence of MC_s, FP_α is kept synchronised and updated at MR_4 until the expiry of its associated timer. We also keep a copy of FP_α at MRs (with the same expiration time) and in mobile nodes (MC_s in this case). Hence, when that moving node comes in range of a new MR with no FP_α knowledge, the MR uses the copy stored in the newly arrived node, if not expired. Else, the new MR uses the expired copy to direct a unicast request to MR_1 for a synchronised copy of FP_α to avoid broadcasting requests for FP_α through the backbone channel. The reason we unicast a request to MR_1 for a copy of FP_α, instead of using the one in the arriving node MC_s, is that a FP_α copy from a mobile node may not be updated due to moving off network. Also, because MR_1 hosts a large number of MCs with the α social tie, the FP_α timer is likely to be reset frequently as a result of (local) MC communication.

If FP_α contains multiple MRs, multicast requests can occasionally be performed to request multiple FP_α copies to guarantee a reliable response, and for validation consistency. For protocol efficiency, it is critical to check that FP_α copies received from different MRs are identical. If not, the protocol calls for synchronization between all/some MRs with FP_α.

IV. Experimental plan and simulation results

We use OPNET [8] to simulate the AODV, TORA, DSR, OLSR and SCG protocols on a 1 km^2 grid with 50 nodes, of which about 20% act as mesh routers. For statistical reliability, we performed simulation runs with 5000 random seeds which each lasted 3600 seconds. The MR units have unlimited battery power and no mobility. They operate two non-overlapping channels via 802.11 interfaces with 11Mbps data rate.

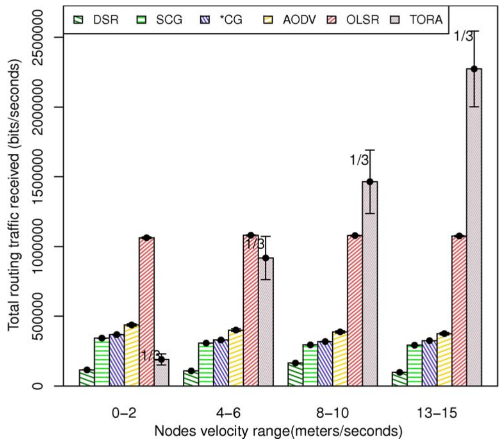

Fig. 3. Routing Traffic Received (bits/seconds)

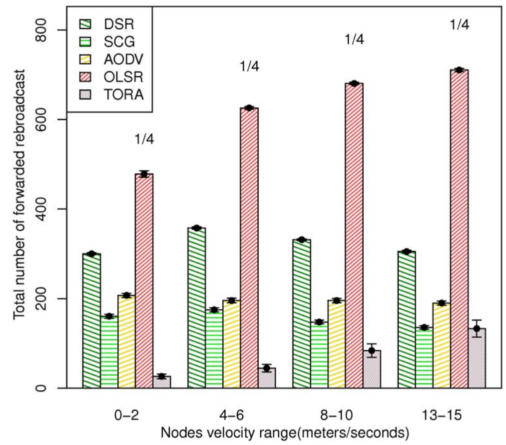

Fig. 4. Routing Broadcast Retransmission

We use the standard Random Waypoint mobility model to handle MC motion. MC speed values are uniformly distributed over 0-15m/s. In all scenarios a source node generates traffic to five defined MCs. To introduce social awareness, we assume that each node has a label informing MRs of its group. We fix the number of social groups at five, each contain 20% of MC, and labels are randomly assigned among MCs at start of each simulation.

The Fig 2 shows the total traffic sent by DSR, SCG, OLSR and TORA protocols for operations such as neighbour sensing and reactive/proactive path discovery procedure etc. The TORA bar is scaled by 1/4 for legibility. To quantify the impact of our footprint, the *CG bars in Fig 2 and 3 respectively, represent the total sent/received overheads of SCG when the footprint mechanism is disabled. It shows clearly that TORA has the highest overhead in bits/seconds. In general, however, routing control overheads increase in frequency and size linearly with speed. DSR and AODV show very little increase because of their reactive nature. SCG produces slightly higher overheads than AODV. Despite its pro-active behaviour, SCG overheads are lower. This can be attributed to the footprint and MRTL mechanisms that reduce redundant rebroadcasts during path discovery (See Fig 4).

Fig 3 shows the total average routing traffic received by the various protocols in bits/second, not counting the cost of the unicast data traffic. The TORA bar is scaled by 1/3 in Fig 3 for legibility. DSR, SCG and AODV, in this order, produce less routing traffic, hence deliver the highest throughput, while TORA has the greatest routing overhead. Note that overheads are higher for TORA and OLSR at higher velocities. Due to the reactive nature of DSR, SCG and AODV, overheads stabilise at different velocities. Interestingly, despite the proactive nature of OLSR and high control overheads, MPRs help keep overhead consistent over a range of velocities.

Fig 4 shows that OLSR generates the most pure flooding, but a large proportion of this rebroadcast traffic is topology control (TC) messages. The OLSR bar is scaled by 1/4 for legibility. The results show that TORA produces the least pure flooding packets, although the route creation process was set during the simulation to be initiated on-demand. Additionally,

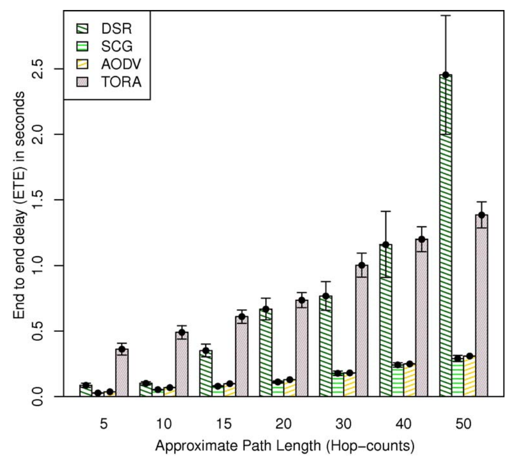

Fig. 5. Route Discovery Delay

a large number of routes are discovered and maintained in a proactively distributed fashion by the IMEP on which TORA is designed to operate. The graphs also show that DSR is the second costliest broadcast forwarding protocol. This is expected since DSR depends on flooding to find paths. Nodes using SCG forward fewer broadcasts than AODV due its stochastic overheads control feature and MRTL. For TORA and OLSR, rebroadcast volume increases with node velocity, while SCG and DSR exhibit the opposite behaviour. We attribute this to physical disconnection due to interference in SCG, AODV and DSR.

The experiments reported in Fig 5 were carried out differently. Instead of varying node velocity, it is the number of nodes (MR and MC) making up routes between a source and a destination that are varied, ranging from 5 to 50 in length. The nodes are configured to remain stationary. Node density is set to remain approximately constant. This allows testing the route discovery delay time (in seconds) in a controlled environment, minimising most issues associated with mobility. Simulated channel conditions and its associative issues, like interference, still exist. Fig 5 indicates that, in general, the route discovery time is proportional to path length. TORA suffers the longest delay in most path lengths, but this delay increases gradually.

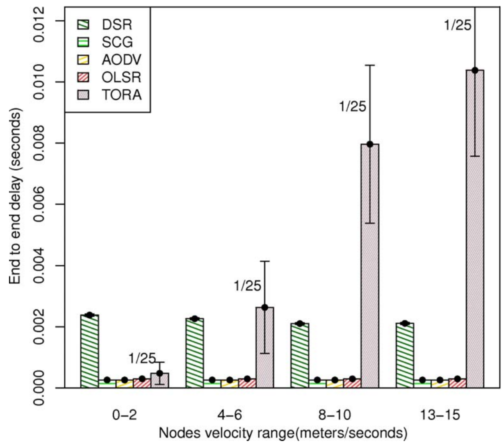

Fig. 6. Network Delay (seconds)

SCG proves to be the fastest in all cases. Interestingly, DSR starts quickly, but later shows a sharp increase in latency. DSR's wide confidence interval indicates uncertaintly in the prediction of DSR routing delay, and this increases with path length.

Fig 6 shows the network delay i.e. the average end-to-end delay for packets sent across the network with mobile nodes. Results are similar to those in Fig 5. However, DSR managed to stabilise the route discovery latency at various speeds. On the other hand, TORA shows the greates latency in highly mobile networks.

V. Discussion

Based on our experiments, we can conclude that the SCG protocol manages to maintain low overheads and delays in most cases. Reducing the time required for routing is an obvious advantage of parallel broadcasting of the source and destination nodes and the usage of social ties. The SCG collision guidance technique can be viewed as a new stochastic broadcast control mechanism.

The DSR protocol performed well in low-mobility environments in term of overheads as shown in Fig 4. This is due to the absence of periodic/proactive behaviour. However, due to DSR's on-demand nature, it suffers substantially from network latency and path discovery delays. DSR performance degrades rapidly with increasing mobility which causes considerable routing overhead due to the source-routing mechanism, and to the accumulated path information stored in packet headers that must be processed fully by intermediate nodes. This routing overhead and latency is directly proportional to the path length. Interestingly, OLSR sends the largest number of pure flooding messages. A high number of the rebroadcasts are topology control (TC) messages. MPRs pro-actively flood TCs to build the necessary topology information base, thus using the generated overhead to provide stability and ease of distribution under various network velocities as shown in Fig 3.

TORA's performance is worst in most cases. One reason is the redundant invalid routes to destinations due to link failures or network partitioning. Using invalid routes introduces extra delays in discovering and setting valid-directed routes. TORA appears to suffer high intrinsic overheads due to its dependence on the generic IMEP layer, but these could perhaps be ameliorated by a custom network layer.

VI. Summary and future work

We have presented a Social-aware Routing Protocol with Parallel Collision Guidance Broadcasting (SCG). SCG improved the route discovery mechanism through on-demand parallel collision guidance broadcasting in WMN. Our protocol reduces overhead associated with mesh client (MC) related updates exchanged between mesh routers (MR) via social knowledge and volatile footprints. Moreover, SCG minimises redundant broadcasts by: (i) positive collisions occurring through the parallel broadcast from the source and destination nodes; (ii) the MR-To-Live (MRTL) technique, which is the number of MRs a broadcast crosses before it is discarded.

We assume static conceptual clustering of MCs in the network, which can symbolise the long social relationship between nodes. More experiments are needed to study the effect of group size. We are planning to enhance this social tie feature in future by allowing MRs to detect and associate social ties to MCs (in real-time) based on frequency and duration of communications. Also, we plan to test the SCG with various MR mesh topologies to investigate if the topology affects performance.

References

[1] Mehran Abolhasan, Tadeusz Wysocki, and Eryk Dutkiewicz. A review of routing protocols for mobile ad hoc networks. *Ad Hoc Networks*, 2(1):1 – 22, 2004.

[2] S.S. Basurra, M. De Vos, J. Padget, T. Lewis, and S. Armour. A zone-based routing protocol with parallel collision guidance broadcasting for manet. In *ICCT*, pages 1188 –1191, November 2010.

[3] M. S. Corson, S. Papademetriou, P. Papadopoulos, V. Park, and A. Qayyum. Internet manet encapsulation protocol (imep). Internet Draft 01, 1999.

[4] Richard Draves, Jitendra Padhye, and Brian Zill. Routing in multi-radio, multi-hop wireless mesh networks. In *In ACM MobiCom*, pages 114–128. ACM Press, 2004.

[5] Pan Hui and Jon Crowcroft. How small labels create big improvements. In *Pervasive Computing and Communications Workshops, 2007. n*, pages 65 –70, March 2007.

[6] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *IEEE INMIC 2001*, pages 62 – 68, 2001.

[7] D. Johnson, Y. Hu, and D. Maltz. The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4. Experimental Protocol for the Internet community RFC 4728, Network Working Group, 2007.

[8] *OPNET Modeler Documentation*, 2011. www.opnet.com.

[9] V.D. Park and M.S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *INFOCOM '97*, volume 3, pages 1405 –1413 vol.3, April 1997.

[10] C.E. Perkins and E.M. Royer. Ad-hoc on-demand distance vector routing. WMCSA '1999. Second IEEE Workshop, pages 90–100.

[11] Asad Amir Pirzada, Marius Portmann, and Jadwiga Indulska. Evaluation of multi-radio extensions to aodv for wireless mesh networks. In *Proceedings of the 4th ACM international workshop on Mobility management and wireless access*, pages 45–51. ACM, 2006.

[12] Technische Informatik Und, Rainer Baumann, Martin May, and Bernhard Plattner. Field based interconnection of hybrid wireless mesh networks. Technical report, Swiss Federal Institute of Technology, 2006.

[13] Brad Williams and Tracy Camp. Comparison of broadcasting techniques for mobile ad hoc networks. In *MobiHoc '2002*, pages 194–205. ACM.