Biologically Inspired Self Selective Routing with Preferred Path Selection*

Boleslaw K. Szymanski, Christopher Morrell,
Sahin Cem Geyik, and Thomas Babbitt

Department of Computer Science
Center for Pervasive Computing and Networking
Rensselaer Polytechnic Institute, 110 8th Street, Troy, NY 12180
{szymansk, morrec, geyiks, babbit}@cs.rpi.edu

Abstract. This paper presents a biologically inspired routing protocol called Self Selective Routing with preferred path selection (SSR(v3)). Its operation resembles the behavior of a biological ant that finds a food source by following the strongest pheromone scent left by scout ants at each fork of a path. Likewise, at each hop of a multi-hop path, a packet using the Self Selective Routing (SSR) protocol moves to the node with the shortest hop distance to the destination. Each intermediate node on a route to the destination uses a transmission back-off delay to select a path to follow for each packet of a flow. Neither an ant nor a packet knows in advance the route that each will follow as it is decided at each step. Therefore, when a route becomes severed by a failure, they can dynamically and locally adjust their routing to traverse the shortest surviving path. Preferred path selection reduces transmission delay by essentially removing back-off delay for the node that carried the previous packet of the same flow. The results reported here for both simulation and execution of a MicaZ mote implementation, show that this is an efficient and fault-tolerant protocol with small transmission delay, high reliability and high delivery rate.

Keywords: routing, wireless sensor networks, route repair, ant colony paradigm, link failure.

1 Introduction

Wireless sensors networks are composed of a large number of nodes equipped with radios for wireless communication, sensors for sensing the environment and

* Research was sponsored by US Army Research laboratory and the UK Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the authors, and should not be interpreted as representing the official policies, either expressed or implied, of the US Army Research Laboratory, the U.S. Government, the UK Ministry of Defense, or the UK Government. The US and UK Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

CPU's for processing applications and protocols. A significant number of wireless sensor networks consist of battery-powered nodes to be able to operate unattended. Such networks require autonomy of management (self-management), fault-tolerance, and energy-efficiency in all aspects of their operation. These properties are especially important for routing, since multi-hop communication is a primitive wireless sensor network operation that is fault-prone as well as energy-intensive. For instance, commonly observed in such networks are faulty (or, potentially subverted) nodes and transient and asymmetric links caused by wildly oscillating packet reception quality. Faulty nodes and transient links cause severe packet loss and spontaneous network topology changes[1, 2]. In terms of energy usage by sensor network node components, radio operation is typically the most costly, as evidenced by a study in [3] and typical hardware specifications given in [4].

A traditional approach to multi-hop routing is to use routing tables that indicate the neighbor to which a packet should be forwarded to reach a destination; prominent examples include AODV[5] and Directed Diffusion[6]. This fundamental approach, which emulates traditional wired network communication, naturally requires nodes to constantly maintain individual neighbors states (e.g., active or sleeping) to support routing decisions. In operating conditions typical for wireless sensor networks, such maintenance often requires significant overhead, especially if fault-tolerance is to be supported. Hence, providing efficient routing protocols that naturally accommodate and perform well in fault-prone conditions is still an open and formidable challenge and is therefore the subject of this paper.

This paper presents the biologically inspired family of Self Selective Routing (SSR) protocols[7], which has been extended with preferred path selection, introduced in this paper. In SSR, after a node currently possessing a packet transmits it, all nodes that receive it decide which one will forward it. This decision is made autonomously by each receiver based on their respective hop distances to the destination using a transmission back-off delay to resolve potential ties.

In this paper, we discuss two novel mechanisms used by SSR(v3), introduced here as compared to SSR(v2) presented in [7]: (i) an efficient and local repair of severed routes and (ii) preferred path selection. The first mechanism allows a node that detected no responders to its transmission broadcast to increase its hop distance to the destination. This increase enables the currently traveling packet to retrace a part of its path. In an effort to make the protocol more tunable, we have enabled the user to choose whether route repair occurs in each packet, or in each node. Repairing the packet increases the hop count only in the individual packet, and provides a temporary alternate route that is desired in the case of transient failures. This method of repair maintains the established topology of the network. Repairing the node increases the hopcount in the node, and provides a permanent change to the network's topology that is desired in the case of permanent failures. The second method introduced in this paper, allows

the node that forwards the current packet to select itself for forwarding the next packet in the flow with essentially no delay. This creates a protocol that is both delay efficient (minimal delay to forward a packet in a normal case) and robust (another node will forward a packet if the preferred node is down or has lost its link to the sender) at the same time.

There are other protocols that, like SSR, route on the premise of avoiding neighbor state maintenance and letting receivers contend for forwarding packets. However, they all require geographical location information, which SSR does not. Three such protocols, GRAd[8], GRAB[9], and BLR[10] are not capable of a route repair. Other protocols, GeRaF[11], IGF[12], PSGR[13] and SIF[14] define eligibility regions for packet forwarding and therefore require detailed knowledge of geographical placement of currently active nodes which is difficult to obtain and maintain in wireless sensor networks.

2 Self Selective Routing

The SSR protocol has been inspired by the use of pheromones by the biological ants to mark paths to guide other ants to food sources without memorizing or prescribing a path explicitly[15, 16]. Accordingly, the SSR protocol consists of three phases: (i) an initial destination request flooding that finds the destination node, (ii) a destination reply flooding that establishes hop distances between each node and the given flow's destination, and (ii) data transmission proper.

The destination request phase corresponds to the initial search for food in which ant scouts randomly explore the environment. In the process, they mark the branching paths with pheromones, which will later guide the ant scout back to the home colony (retracing the path, an ant will follow the strongest marks as they were most recently visited on the way out). Packets sent in this stage are referred to as DREQ (Data Request) packets. The destination reply phase corresponds to a walk back to the colony by an ant that found a food source. Walking back, an ant will mark branches on the path home with pheromones to distinguish the return path from other, unused paths. Packets sent in that stage are called DREP (Data Reply) packets. This initial flooding could be done once at the sensor network deployment to all potential destinations (in wireless sensor networks there is often only one destination, the base station, making the initial two stages particularly simple). We used for this purpose the signal-strength aware flooding technique described in [17] which also provides more details on the these two initial stages. This paper focuses on data transmission stage itself.

2.1 Data Transmission in SSR

As shown in figure 1, the data transmission stage can be represented by a Finite State Automaton (FSA) that defines the input, actions and output generated in each state of a node in the network as it routes data (similar FSAs can be

defined for the destination request and reply stages). For example, when a node receives a packet that it has not seen before, it immediately moves into the *NEW* state, and depending on its input and status (e.g. data packet received by the destination, data packet received by a node closer to the destination then the sender, acknowledgment packet received, etc) the node transitions itself into the corresponding state and executes the associated actions (for clarity, not shown in the figure).

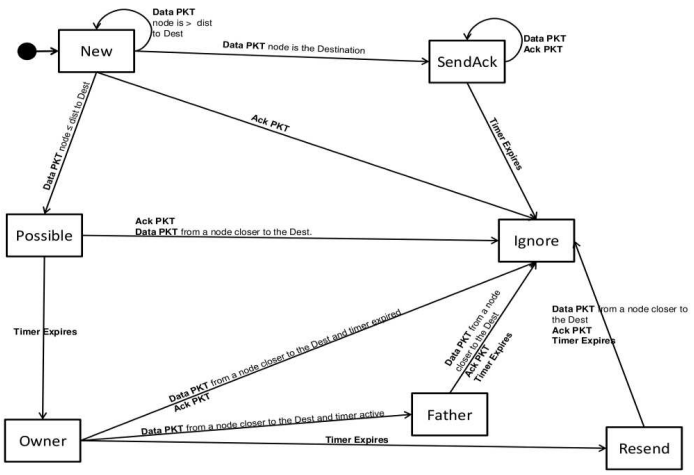

Fig. 1. State diagram for SHR-PP

When the source transmits a DATA packet, only neighbors that are closer to the destination than the sender will react. Depending on the reacting nodes proximity to the destination in relation to the sending node, it selects a transmission back-off delay. That delay is uniformly distributed between 0 and $\lambda/2$ if the reacting node is one hop closer to the destination. If the reacting node is more than one hop closer, the back-off delay is selected between $3\lambda/4$ and λ. This difference in back-offs ensures that the more reliable single hop closer neighbors have priority over the less reliable multiple hop closer neighbors. λ is a scaling factor that allows us to tune the probability of collision of the nodes' responses. If, during the back-off delay, a DATA packet is received from a node that is closer to the destination, the receiving node cancels the forwarding of the DATA packet and moves to the Ignore state. When the transmission back-off time expires, the node increments the packet's actual hop count by one, sets the expected hop count to its hop distance to the destination and then transmits the packet.

After forwarding the packet, the node monitors the carrier to determine if the packet has been forwarded. Lack of forwarding causes retransmissions, and

finally route repair which is accomplished by increasing the node or packet's hop distance to the destination by 2 and retransmitting.

To promote reliable links, we introduced a preferred path selection, in which a node which forwarded the current packet will respond almost immediately to a transmission of a new packet in the same flow. To simplify processing, these nodes calculate their delay by dividing the regularly selected back-off delay by 625, while ensuring that it remains larger than the radio transition time. This results in a back-off delay between 20 and 160μs, given λ is 100ms. This minimizing of back-off delay ensures the node future self-selections, thereby stabilizing repeatedly traversed paths. In the ant pheromone model, as ants move over different paths, and the once strongly scented but now less used paths begin to fade, ants shift their routes to the paths that are most frequently used. In reference to the slow fading of the pheromone, we have chosen to not follow the biological inspiration literally. Instead, we restore the full range back off delay immediately after the preferred node fails to self-select, as such failure indicates that the recently used node is no longer reliable. Despite its simplicity, the effect of using the preferred path selection in SSR(v3) is very positive, as demonstrated in the section below.

3 Performance Evaluation

Using both the SENSE wireless network simulator [18] and MicaZ sensor motes [4], we performed a series of experiments to compare the performance of SSR(v2) with the newly designed SSR(v3). Additionally, in the case of simulations, both protocols were compared with a traditional routing protocol, AODV [5].

3.1 Simulations

We tested three different scenarios. The first one involved a single sink (base station) collecting data from many sources, which is a typical sensor network setting. The second scenario investigated transient failures, while the third one evaluated the performance of the protocols under permanent failures. In failure simulations, faults occurred with varying probabilities, while the sink network simulation evaluated the performance with a varying number of sources.

The simulation topography consists of an 8 unit by 8 unit terrain populated with 500 nodes placed randomly. Each node is stationary and has a single unit nominal transmission range. The wireless medium is simulated with the free space propagation model[20], and the radio modeled operation at 914 MHz with 1 Mb/s of bandwidth. Packet sizes were uniformly distributed around a mean of 1000 bytes and were sent at uniformly distributed intervals with a mean of 40 seconds. MAC broadcast was used in which a node senses the carrier and broadcasts only if no other transmissions are detected. The average hop distance between sources and their respective destinations is 7.8 hops.

Each simulation was executed eleven times, each time with a different random number seed for a simulation time of 3,000 seconds per seed. The same 11 seeds were used for all simulation sets. λ was set to 100ms for all simulations.

Single Sink Network In a wireless sensor network, using a single sink is common. For example, any network that contains a single base station is usually configured that way. Such configuration may result in heavy traffic congestion near the sink. Such congestion has the possibility of causing massive amounts of collisions, and could possibly stop the network from functioning at all. In sink network simulations, we varied the number of sources transmitting to a single sink from 10 to 100 to test the scalability of each protocol.

As is apparent in figure 2, a single sink network is where SSR(v3) shows its worth, and where AODV breaks under its limitations. The protocols' end-to-end delays were so drastically different, that a logarithmic scale was necessary to plot them together. As the density of sources increases from 70 to 100, which is 14% to 20% of the nodes in the network transmitting, AODV required approximately 100 seconds to transfer a packet from the source to the destination. Although SSR(v3) does increase its delay slightly, it still manages to keep that delay to under 0.1 seconds, even with 100 nodes transmitting. Clearly, the preferred path selection allows packets to move across the network quickly enough that a packet reaches the destination before the following packet is transmitted, thus avoiding any significant impact from congestion.

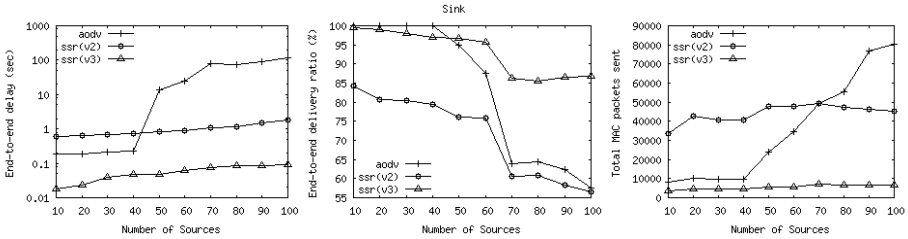

Fig. 2. Transmission delay, delivery ratio, and total MAC packets sent in the case of a single sink network for three compared protocols: AODV, SSR(v2) and SSR(v3)

SSR(v3) is also superior in terms of delivery ratio. As sources increase to 100, SSR(v3)'s delivery ratio decreases to near 90%, while AODV's drops to nearly 55%. The reasons are the same as described earlier, where AODV succumbs to the congestion around the sink node, while SSR(v3) is fast enough to avoid significant congestion. Also in total MAC packets sent, SSR(v3) manages to use less than 10% of the packets that AODV uses at 100 sources.

Failure Simulations The failure sensitivity of SSR's route repair routine can be tuned by adjusting the number of retransmissions by the forwarding node required to invoke route repair. By increasing this value, SSR can be successfully employed in a network with a high rate of transient failures, but maintains performance in a network with a high rate of permanent failures. In our tests, two retransmissions were required to invoke route repair. Since a packet transmission interval is 40 seconds, a node failure lasting less than 80 seconds on average would not change the route from the source to the destination. As mentioned earlier, the protocol is also tunable, because route repair can be executed temporarily on individual packets, or permanently on the nodes.

Transient Failures There are several possible causes for transient node failures, such as error-prone links, power management induced duty cycles, or excessive packet collisions. Of these, the duty cycle induced failures are the least disruptive since they may be coordinated with the networking protocol. The presented simulation results are based on a random transient failure model, so they exaggerate the effect of duty cycles on the protocols. In the transient failure simulations, each node was assigned a mean active time and a mean sleep time. The sum of these two times was fixed at 200 seconds. The time spent in each mode was distributed exponentially about the mean value.

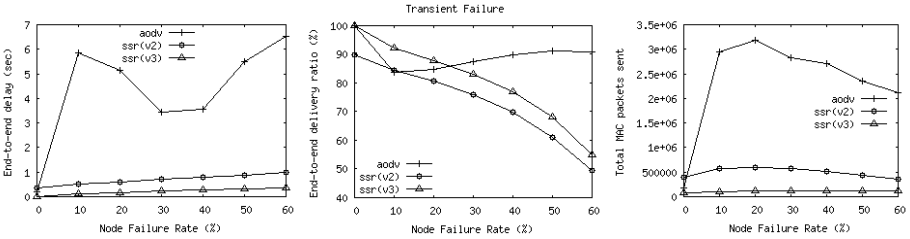

Fig. 3. Transmission delay, delivery ratio, and total MAC packets sent in the case of transient failures for three compared protocols: AODV, SSR(v2) and SSR(v3)

As seen in figure 3, AODV has the worst transmission delay that increases significantly with the transient failure rate. SSR(v3) has by far the smallest delay of the three protocols, with a factor 10 advantage over AODV for the most failure prone case. SSR(v3) has lower delays than AODV for all cases in which transient failures are present. Both SSR(v2) and SSR(v3) only slightly increase the incurred transmission delay when the transient failure rate is growing.

In terms of delivery ratio, AODV is the best, dropping from 100% in a reliable case to 90% for 60% transient failure rate. SSR(v3) delivery ratio drops from 100% to 55% over the same region while SSR(v2)'s is slightly lower, dropping

from 90% to 50%. However, AODV requires a much larger number of MAC packet transmissions than either SSR(v2) or SSR(v3). This is because to find a new path, AODV's route repair algorithm initiates a new route request phase, causing a flood of packets from the point at which the route is severed. AODV uses over 30 times more packets than SSR(v3). Hence, by implementing a simple replication scheme, in which each packet in SSR(v3) is sent 3 times, we could bring the SSR(v3) delivery rate into a range that is more comparable with AODV, while still keeping the number of MAC packets 10 times lower. The impact of this huge difference in packets required will show itself primarily in the energy consumption of the protocols.

Permanent Failures In the permanent failure model, each node had a random chance of failing. Nodes that fail had their failure start time uniformly distributed over the simulation time. In this scenario, trends observed for transient failures continue but are less pronounced.

As seen in figure 4, as the number of node failures increase, the transmission delay also increases while the delivery ratio generally decreases. SSR(v3) achieves the lowest and most stable transmission delay of all three protocols. Even at 60% failure rate, its delay is only slightly increased compared with its delay in the reliable network, and is nearly 10 times better than that of AODV. Although SSR(v3) delivery ratio is not 100% as is AODV, it still shows a 16% improvement over SSR(v2), and stays at or above 96%. This improvement arises because any node that tends to get entangled in external collisions will not be able to forward packets consistently and therefore sooner or later it will be replaced in SSR(v3) by a node that can, if such a node exists.

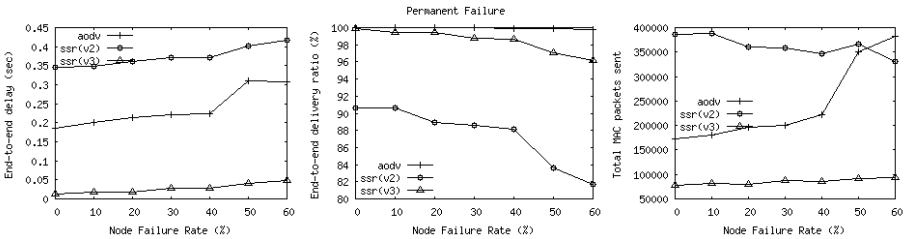

Fig. 4. Transmission delay, delivery ratio, and total MAC packets sent in the case of permanent failures for three compared protocols: AODV, SSR(v2) and SSR(v3)

Again, the most significant difference between AODV and SSR arises in MAC packet sent. As failures increase, the number of packets required for AODV to maintain 100% delivery begins to quickly increase, while SSR(v3) maintains practically the same number for all failure rates. Hence, for the same reasons

as discussed in transient failure simulations, the ratio of the numbers of MAC packets used increases from an initial factor of 2 to a factor of 5 for the 60% permanent failure rate.

SSR's approach to route repair is clearly more local and efficient, as evidenced by the plots. It should also be noted that under SSR(v2) and SSR(v3), the path lengths and number of packets per hop remain nearly constant over the range of permanent and transient failure rates. This demonstrates that priority-driven opportunistic behavior of these protocols is highly accommodative to potentially disruptive duty cycles and node failures.

3.2 Implementation on MicaZ Motes

We have implemented the new SSR(v3) protocol on MicaZ motes [4] using TinyOS version 1.1.7 to compare performance of this implementation with the implementation of SSR(v2) on the same hardware [19]. In the implementation, we used B-MAC with acknowledgments disabled to provide link layer functionality. DATA packets of 29 bytes were sent for 12.5 min at a rate of 5sec/packet in an indoor environment. The radio power was set to -21dBm and a distance of 1m provided a reliable delivery rate. However, with moderate probability some long distance transient links also formed. Both compared protocols used the same λ of 22ms.

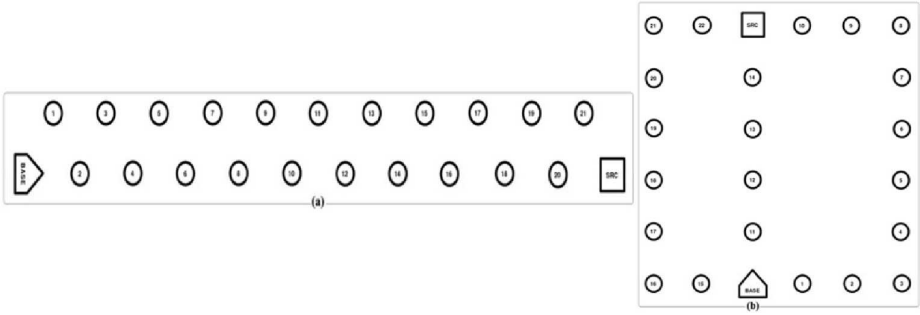

Fig. 5. (a) Double line topology, (b) Route repair topology. Nodes have reliable connections with their closest neighbors and transient connections with others. The base indicates the direction in which all motes are oriented.

SSR(v2) was compared to SSR(v3) on two topologies. Double line topology, shown in figure 5(a), has two motes at each hop eligible to forward the packet. Route repair topology from figure 5(b), contains three unequal length and disjoint paths: a short, medium and long one. With these topologies, we tested the repair capabilities of each protocol. During testing we blocked motes 12 and 13 in the network by placing a metal container over the motes after the first 5 minutes of the test.

As shown in table 1, in double line topology experiments, SSR(v3) provided a large improvement in delivery rate, more than halving the percentage of lost packets in SSR(v2). It also achieved a modest improvement in the end-to-end delay compared to SSR(v2). On route repair topology both protocols performed equally well.

Table 1. Experimental results for double line and route repair topologies.

	Double line		Route repair	
	SSR(v2)	SSR(v3)	SSR(v2)	SSR(v3)
Packets Sent	246	277	110	117
Packets Received	1070	1279	304	317
Packet Ratio (rec/sent)	4.33	4.61	2.74	2.69
Delivery Rate	47.3%	77.3%	77.3%	74.9%
End-to-end Delay	209 ms	174 ms	117 ms	122 ms
Average Hop Count	7.26	7.07	5.11	5.15

To better understand these results, we plotted the time versus delay of each successfully transmitted packet in both topologies for SSR(v3) (see figure 6). Initially, packets frequently followed different length paths showing transient nature of links in the experiment and therefore decreasing the effectiveness of the preferred path selection. However, later on, the nodes with stable link tend to persist longer on path used for transmission, increasing the advantage of SSR(v3) over SSR(v2). The failure of nodes 12 and 13 in the middle of a run (around packet 160) on route repair topology prevents this effect from occurring, resulting in similar performance of both protocols.

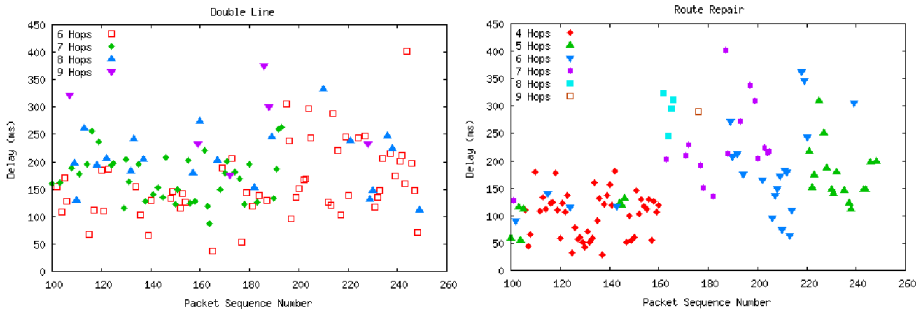

Fig. 6. Packet sequence number versus delay for SSR(v3) executed over the two topologies

In the current implementation, both SSR(v2) and SSR(v3) allowed longer but transient links to win self-selection. On the first glance, this seems to be beneficial as such links may decrease the number of hops needed to reach destination. However, closer inspection reveals that such links may increase the chance for retransmissions because the long links have relatively small probability of being overheard by the sender when they respond and transmit a packet towards the destination.

4 Conclusion and Future Works

In this paper, we have presented SSR(v3), which naturally accommodates fault-prone sensor network routing conditions and takes full advantage of the properties of the broadcast communication primitive of such networks. SSR provides seamless route repair in cases of permanent or transient failures of nodes or links. The preferred path selection introduced here allows the packet to traverse not only the shortest path to the destination, but also the most reliable one. It also preserves SSR(v2)'s ability to use other links if the preferred link is down. The resulting significant decrease in the transmission delay and increase in delivery ratio address the most important weaknesses of SSR(v2).

In future work, we intend to extend the SSR family of protocols to address issues of mobility and energy efficiency, both of which are common in some wireless sensor network applications. While SSR(v3) may currently accommodate mobility, it is not yet explicitly optimized for it. Mobility shortens the time over which hop distance tables remain valid. To retain SSR's autonomic behavior, we are researching how to efficiently update these tables based on local observations of node movement. SSR can already accommodate topology changes caused by energy-efficient topology control algorithms, such as ESCORT [21]. However, explicitly incorporating a topology control algorithm into SSR is still a challenge, as it requires ensuring that the algorithm is not so aggressive that it overcomes SSR's ability to find eligible forwarders for every packet.

References

1. A. Woo, T. Tong, D. Culler: Taming the underlying challenges of reliable multihop routing in sensor networks. Proc. ACM SenSys03, ACM Press, New York, 2003, pp. 14-27.
2. J. Zhao, R. Govindan: Understanding packet delivery performance in dense wireless sensor networks. Proc. ACM SenSys 03, ACM Press, New York, 2003, pp. 1–13.
3. G. Anastasi, A. Falchi, A. Passarella, M. Conti, E. Gregori: Performance measurements of motes sensor networks. Proc. 7th ACM Intern. Symp. Modeling, Analysis and Simulation of Wireless and Mobile Systems, ACM Press, New York, 2004, 174-181.
4. Crossbow Technology, Inc., http://www.xbow.com
5. C. Perkins, E. Belding-Royer, S. Das: RFC 3561-ad hoc on-demand distance vector (AODV) routing. http://www.faqs.org/rfcs/rfc3561.html

6. C. Intanagonwiwat, R. Govindan, D. Estrin: Directed diffusion: a scalable and robust communication paradigm for sensor networks. Proc. ACM MobiCom, ACM Press, New York, 2000, pp. 56-67.
7. J.W. Branch, M. Lisee, B.K. Szymanski: SHR: Self-Healing Routing for wireless ad hoc sensor networks. Proc. Intern. Symp. Performance Evaluation of Computer and Telecommunication Systems SPECTS'07, SCS Press, San Diego, 2007, pp. 5-14.
8. R. Poor: Gradient routing in ad hoc networks. http://www.media.mit.edu/pia/Research/ESP/texts/poorieeepaper.pdf
9. F. Ye, G. Zhong, S. Lu, L. Zhang: Gradient broadcast: a robust data delivery protocol for large scale sensor networks. ACM Wireless Networks, 11(2) (2005).
10. M. Heissenbttel, T. Braun, T. Bernoulli, M. Waelchli: BLR: beaconless routing algorithm for mobile ad hoc networks. Computer Communications Journal, 27(11) (2004).
11. M. Zori, R.R. Rao: Geographic Random Forwarding (GeRaF) for ad hoc and sensor networks: multihop performance. IEEE Trans. Mobile Computing, 2(4) (2003) 337-348.
12. B. M. Blum, T. He, S. Son, J.A. Stankovic: IGF: a robust state-free communication protocol for sensor networks. Technical Report CS-2003-11, University of Virginia, Charlottesville, 2003.
13. Y. Xu, W.-C. Lee, J. Xu, G. Mitchell: PSGR: priority-based stateless geo-routing in wireless sensor networks. Proc. IEEE Conf. Mobile Ad-hoc and Sensor Systems, IEEE Computer Society Press, Los Alamitos, 2005.
14. D. Chen, J. Deng, P.K. Varshney: A state-free data delivery protocol for multihop wireless sensor networks. Proc. IEEE Wireless Communications and Networking Conf., IEEE Computer Society Press, Los Alamitos, 2005.
15. O. Cordon, F. Herrera, T. Stutzle: A review on the Ant Colony Optimization Metaheurstics: Basis, Models and New Trends. Mathware & Soft Computing, 9 (2002).
16. S. Koenig, B. K. Szymanski, Y. Liu: Efficient and Inefficient Ant Coverage Methods. Annals of Mathematics and Artificial Intelligence, 31(1-4) (2001) 41-76.
17. G. Chen, J. Branch, B.K Szymanski: Local leader election, signal strength aware flooding, and routeless routing. 5th IEEE Intern. Workshop Algorithms for Wireless, Mobile, Ad-Hoc Networks and Sensor Networks WMAN05, IEEE Computer Society Press, Los Alamitos, 2005.
18. G. Chen, J.W. Branch, M. Pflug, L. Zhu, B.K. Szymanski: SENSE: a wireless sensor network simulator. Advances in Pervasive Computing and Networking, Springer, New York, 2004, pp. 249-267.
19. K. Wasilewski, J. Branch, M. Lisee and B.K. Szymanski: Self-healing routing: a study in efficiency and resiliency of data delivery in wireless sensor networks. Proc. Conference on unattended Ground, Sea, and Air Sensor Technologies and Applications, SPIE Symposium on Defense & Security, Orlando, FL, April, 2007.
20. T. S. Rappaport: Wireless Communications: Principles and Practice. Prentice Hall, 1996.
21. J.W. Branch, G. Chen, B.K. Szymanski: ESCORT: Energy-efficient Sensor network Communal Routing Topology using signal quality metrics. Proc. 4th Int. Conf. on Networking, LNCS, Springer-Verlag, Berlin, vol. 3420, 2005, pp. 438-448.