

Compressed Sensing based Protocol for Interfering Data Recovery in Multi-Hop Sensor Networks

Megumi Kaneko
Graduate School of Informatics
Kyoto University

Yoshida Honmachi Sakyo-ku, 606-8501 Kyoto, Japan

Khaldoun Al Agha
Laboratoire de Recherche en Informatique
Université Paris-Sud
91405 Orsay cedex, France

Abstract— We consider a multi-hop wireless sensor network that measures sparse events and propose a novel protocol based on Compressed Sensing (CS) as an alternative to traditional Media Access Control (MAC) scheduling and routing protocol. Instead of avoiding collisions, our CS-based protocol exploits interferences by superimposing the data measurements “over-the-air”, simultaneously received at any node. Thanks to our protocol design, each node is able to recover and forward only new data towards the sink. Our protocol achieves near zero reconstruction errors at the sink, while greatly reducing overhead and delays compared to conventional methods. These results reveal a new and promising approach to protocol design through CS.¹

I. INTRODUCTION

Recently, the groundbreaking theory of Compressed Sensing (CS) was developed, enabling a vector with correlated entries, i.e., that can be transformed into a sparse vector through a transformation basis, to be recovered with high accuracy from a few random projections onto another, incoherent basis [1]. Widely used in the domains of digital signal and image processing, CS is also a highly promising tool for improving the performance of wireless networks. Several works have exploited CS in wireless sensor networks with space-time correlated data as in [2][3][4][5]. By contrast, [6] considers a one-hop sensor network where multiple nodes transmit their measurement simultaneously to the sink. As sparse events are assumed, i.e., the number of measurements received at the sink is much smaller than the number of sensors, each measurement can be recovered from few observations with CS. A similar approach is taken for developing new multiple access schemes as in [7] or downlink scheduling [8].

In this work, we consider a multi-hop wireless sensor network and take the approach of [6]–[8] where the events to be reported occur sparsely. We develop a protocol based on CS and flooding that enables the sink to obtain and reconstruct the sparse measurement data with high accuracy without any heavy routing nor MAC protocol, by exploiting collisions. In a one-hop network, the interfering/superimposed data measurements due to packet collisions may be resolved by the CS-based algorithms in [6]–[8], but they pose major problems in a multi-hop network where every node forwards all received packets, as the number of interfering measurements may increase drastically, thereby causing the CS-based algorithms

to perform poorly due to loss of sparsity. Moreover, if different measurements of a same source but generated at different times are contained in the same packet, they would hardly be resolvable. Note also that our goal is radically different from the existing CS data recovery schemes in sensor networks: here, the generated events themselves are assumed to be sparse as in [6], which happens whenever only a few sensors are allowed to send significant data, such that these sparse data have no correlation between themselves. In that case, the existing CS based techniques as in [2][3][4][5] will not be applicable, since there is no data correlation to be exploited: each data would be simply forwarded to the sink by usual MAC and routing protocols, which are actually assumed in these works to guarantee collision-free transmissions. Instead, the novelty of our work is to *take advantage of CS techniques to enable efficient delivery of sparse data superimposed “over-the-air” by exploiting flooding and interferences*, as opposed to conventional collision-avoiding MAC scheduling and routing. Simulation results show that our CS based protocol achieves excellent data reconstruction at the sink, while greatly reducing overhead and delays compared to conventional protocols.

II. SYSTEM MODEL

We consider a multi-hop wireless sensor network with N nodes and one sink, forming a lattice as shown in Fig. 1. If a sensor S_n detects an event, the measurement is forwarded to the sink based on the proposed protocol. As in [6] which considers a one-hop sensor network, we assume digital transmissions with measurements $x \in \{-1, 1\}$, and that there may be up to K events occurring simultaneously within the whole network, but these events are considered to be sparse compared to the number of nodes, namely $K \ll N$.

Given the lattice structure of the network, each node only communicates with its closest neighbors at distance d , so there may be 2, 3 or 4 neighbors depending on the node’s location. For example, a packet sent by node S_1 in Fig. 1 will be received by nodes S_2 and S_6 only, while a packet sent by node S_7 will be received by nodes S_2 , S_6 , S_8 and S_{12} . For simplicity, it will be assumed that nodes take turns for being in transmit or receive mode: if at time t , a node at position (i, j) in the grid is in transmit mode, then at time $t + 1$ it will be in receive mode while its four neighbors in positions $(i \pm 1, j)$, $(i, j \pm 1)$ will be in transmit mode. Nodes are assumed to be synchronized as in [6]. Channels are considered

¹This work was supported by the Grant-in-Aid for Scientific Research no. 23760334, Ministry of Education, Science, Sports, and Culture of Japan.

to be Additive White Gaussian Noise (AWGN), and all links between neighbors at distance d have the same gain. That is, assuming fading and path loss effects would not affect the protocol much if the channel gains from each neighboring node are known at the receiving node, as assumed in [6].

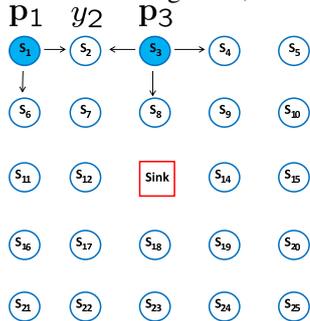


Fig. 1. Multi-Hop Wireless Sensor Network Model (1)

III. PROPOSED PROTOCOL

To exploit "over-the-air" data aggregation, we consider a routing based on a simple flooding procedure whereby upon packet reception, each node broadcasts this packet locally until it reaches the sink. However, unlike the single-hop case, there are major issues to be resolved in this multi-hop setting. If each node simply forwards all received packets, the number of superimposed measurements will increase drastically, leading to poor CS recovery due to loss of sparsity in each packet. Moreover, if each source node is assigned a signature sequence as in the single-hop case, it will not be possible to distinguish whether the multiple received packets are due to self-interference, i.e., they are copies of a same original packet, or if they originate from the same source but at different times.

To alleviate these issues, the proposed protocol is designed as follows. First, the length of the signature sequences is increased so that each sequence simultaneously codes over ID space and time space, i.e., each measurement is associated with a unique sequence $\mathbf{a}_{n,l}$, vector of size M , that identifies the pair (origin node ID S_n , number of hops l). Here S_n is the sensor that originally detected this measurement, and l is its time stamp parameter, defined as the number of hops elapsed by forwarding the packet bearing this measurement, counted from its origin node up to its current location in the network.² We have $\mathbf{a}_{n,l} \in \{-1, +1\}^M$, given by a pseudo-random signature sequence generated by Rademacher random variables of probability $1/2$. Thus, fixing a maximum delay for a packet or time to live L in number of hops, which depends on the network size, there are NL sequences $\mathbf{a}_{n,l}$ of size M , where we set $M < NL$ for reducing overhead consumption. DS-CDMA type of modulations may be used for allowing the superposition of synchronously received signatures. Note that synchronization here is not such a stringent assumption as the considered application typically requires low data rates. Sequences $\mathbf{a}_{n,l}$ are gathered into matrix \mathbf{A} of size $M \times NL$, whose column $q = (n-1)L + l$ contains $\mathbf{a}_{n,l}$. In practice, these sequences may be generated by a Hash function with known

²Among NL , two equal or opposite sequences may be chosen from the total of 2^M . But as $NL \approx 300 < 2^M \approx 10^{10}$ here, such events rarely occur.

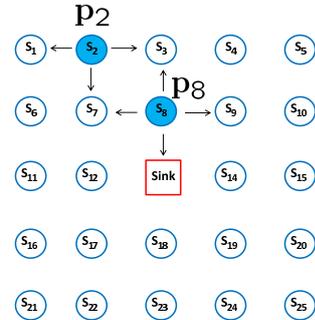


Fig. 2. Multi-Hop Wireless Sensor Network Model (2)

seeds at all nodes and hence may be assumed to be known without additional overhead, even if only a subset of these are actually used in the network, i.e., \mathbf{A} may group all possible sequences. For notational convenience, let \mathbf{x}_{ALL} denote the NL -length sparse vector where $\{\mathbf{x}_{\text{ALL}}\}_{(n-1)L+l} = x_{n,l}$ which groups all the data measurements over the whole network to be reported with all possible hop counts, i.e., $\mathbf{x}_{\text{ALL}} = [x_{1,0} \dots x_{1,L} \dots x_{n,l} \dots x_{N,0} \dots x_{N,L}]^T$ where the $L+1$ first terms store the data of source node S_1 with hop counts $l=0 \dots L$ and similarly, the $L+1$ last terms for source node S_N with $l=0 \dots L$. Thus, each node S_k receives the M -length vector y_k in (2), giving an under-determined system of equations where the NL -length vector of unknowns should be recovered with only M observations. CS theory stipulates that \mathbf{x}_k can be recovered accurately given the recovery condition $M \geq cK \log(N/K)$ where c is a constant, provided that \mathbf{x}_k is sparse or has a sparse representation in some transform domain [9].

Then, the proposed algorithm works as follows:

1. If source node S_j in transmit mode detects a new event with measurement value $x_{j,0} \in \{-1, +1\}$ initialized by $l=0$, it transmits the packet of M bits $p_j = x_{j,0} \mathbf{a}_{j,0}$, which is received by all its neighbor nodes $k \in \mathbf{N}(S_j)$.³ For example, assuming S_1 and S_3 to be source nodes in Fig. 1, at time $t=0$, packet $p_1 = x_{1,0} \mathbf{a}_{1,0}$ is received by S_1 's neighbors S_2, S_6 and packet $p_3 = x_{3,0} \mathbf{a}_{3,0}$ by S_3 's neighbors S_2, S_4, S_8 .

2. At a time t , if sensor node S_k is in receive mode, it receives signal \mathbf{y}_k containing all simultaneous transmissions from its neighbor nodes $n \in \mathbf{N}(S_k)$. The received signal at node S_k is

$$\mathbf{y}_k = \sum_{n \in \mathbf{N}(S_k)} p_n = \sum_{j,l} x_{j,l} \mathbf{a}_{j,l} + \mathbf{z}_k, \quad (1)$$

where $x_{j,l}$, the measurement with source node S_j is included in packet p_n if it was actually forwarded by node S_n . Vector \mathbf{z}_k of size M , denotes the AWGN. Using matrix \mathbf{A} defined above, (1) is equivalently reformulated as

$$\mathbf{y}_k = \mathbf{A} \mathbf{x}_k + \mathbf{z}_k, \quad (2)$$

where \mathbf{x}_k is the vector of size NL whose i^{th} component is

$$\{\mathbf{x}_k\}_i = \begin{cases} \{\mathbf{x}_{\text{ALL}}\}_i & \text{if } \{\mathbf{x}_{\text{ALL}}\}_i = x_{j,l} \text{ is received by } S_k, \\ 0 & \text{otherwise.} \end{cases}$$

In the example of Fig. 1, note that $S_j = S_n$, i.e., the source nodes and transmit nodes coincide since we describe Step 1,

³If a new event was detected during the node's receive mode, it will store the measurement and send it in the next transmit slot with hop count $l=1$.

so the received signal at receiver node S_2 is written $\mathbf{y}_2 = \mathbf{A}\mathbf{x}_2 + \mathbf{z}_2$, where $\mathbf{x}_2 = [x_{1,0} \ 0 \dots 0 \ x_{3,0} \ 0 \dots 0]^T$.

3. Receiver node S_k decodes each measurement contained in the received signal \mathbf{y}_k , using CS principles. In particular, this problem can be formulated by ℓ_1 - ℓ_2 optimization as

$$\tilde{\mathbf{x}}_k = \arg \min_{\mathbf{x}_k} \frac{1}{2} \|\mathbf{y}_k - \mathbf{A}\mathbf{x}_k\|_2^2 + \lambda \|\mathbf{x}_k\|_1, \quad (3)$$

where Lagrange Multiplier λ makes a trade-off between representation error and sparsity of the solution. The main computational effort of the proposed scheme arises from this step (3), solved by the Iterative Shrinkage-Thresholding Algorithm (ISTA) as it only requires very low complexity [9].

4. The reconstructed vector $\tilde{\mathbf{x}}_k$ is renormalized so that each component belongs to $\{-1, +1\}$.

5. After decoding the received measurements at node S_k , their respective (source ID, time stamp)-sequence $\mathbf{a}_{j,l}$ could be identified. Node S_k then compares the sequence $\mathbf{a}_{j,l}$ of each newly decoded measurement to the sequences of previously received ones that are stocked in its local table $\mathbf{Q}(S_k) = \{\mathbf{a}_{q,l'} \text{ received by } S_k, 0 \leq l' \leq L\}$.

- if there exists a sequence $\mathbf{a}_{q,l'}$ in $\mathbf{Q}(S_k)$, $q = j$, $l' \leq l$: as fading is not assumed, any new data from S_j should have a smaller hop count $l < l'$ unless the new data packet that followed the shortest path was lost, which rarely occurs in this setting. So with high probability, this data was already received. Thus, S_k discards the decoded data $\tilde{x}_{j,l} \mathbf{a}_{j,l}$, from the set of data to be forwarded next.
- if there exists a sequence $\mathbf{a}_{q,l'}$ in $\mathbf{Q}(S_k)$, $q = j$, $l' > l$: although both share the same origin node S_j , this is a new data as it has a smaller hop count than the one stocked. Therefore, S_k will forward the decoded data $\tilde{x}_{j,l} \mathbf{a}_{j,l}$.
- in all other cases, S_k will forward $\tilde{x}_{j,l} \mathbf{a}_{j,l}$.

For example, in Fig. 1, node S_2 reconstructs $x_{1,0}$, $x_{3,0}$ from \mathbf{y}_2 based on Steps 3 and 4. Both measurements will be forwarded since they are the first ones to be received, but the local table is updated to $\mathbf{Q}(S_k) = \{\mathbf{a}_{1,0}, \mathbf{a}_{3,0}\}$.

6. For all data $\tilde{x}_{j,l} \mathbf{a}_{j,l}$ to be forwarded, the hop count is incremented to $l + 1$. All decoded measurements as well as sequences in $\mathbf{Q}(S_k)$ for which $l > L$ are discarded. Then, S_k superimposes all the remaining data to be forwarded into a packet p_k composed of M bits, $\mathbf{p}_k = \mathbf{A}\tilde{\mathbf{x}}_k = \sum_{j,l} \tilde{x}_{j,l} \mathbf{a}_{j,l}$, and forwards it. In Fig. 2, among all even nodes sending at $t = 1$, S_2 sends $p_2 = \tilde{x}_{1,1} \mathbf{a}_{1,1} + \tilde{x}_{3,1} \mathbf{a}_{3,1}$, which is received by S_1, S_3, S_7 . Based on Step 5, S_1 discards $\tilde{x}_{1,1}$ and S_3 , $\tilde{x}_{3,1}$. Similarly, S_8 sends $p_8 = \tilde{x}_{3,1} \mathbf{a}_{3,1}$, received by the Sink.

7. The Sink runs the reconstruction algorithm for all incoming packets during the session of duration T_{out} . If multiple versions of the same data are received, diversity gain may be achieved. The measurement time is given by the first received packet: if $\tilde{x}_{j,l}$ is received for the first time, then its time is given by $T-(l+1)$, where T is the actual time at the Sink in hop counts. Given the procedure of Step 5, it is very unlikely that multiple versions of a given measurement with different hop counts are simultaneously received at the Sink for the first time. Thus, if several measurements from a node are received,

they will correspond to different measurements of that node. That is, if the Sink receives $\mathbf{p}_k = \tilde{x}_{j,l} \mathbf{a}_{j,l} + \tilde{x}_{j,m} \mathbf{a}_{j,m}$, with $l < m$ and has a local table $\mathbf{Q}(\text{Sink}) = \{\mathbf{a}_{j,l'}\}$, then most probably $l \leq l' < m$, i.e., $\tilde{x}_{j,m}$ is the same as $\tilde{x}_{j,l'}$, but received through a longer route in the network. Then, $\tilde{x}_{j,l}$ will be considered as a new measurement with time $T-(l+1)$ and $\tilde{x}_{j,m}$ as a replica of $\tilde{x}_{j,l'}$, providing diversity.⁴

IV. NUMERICAL RESULTS

In a network of 49 nodes within a regular 7 by 7 grid, K sources generate measurements. These sparse generation events occur at random times during each session, e.g., they may occur simultaneously. Each session ends after time out T_{out} in hop counts, at which point the measurements collected by the sink are evaluated. The Signal-to-Noise Ratio (SNR) for each link is fixed to 30 dBs. Each simulated point is averaged over 300 sessions. Note that here only one measurement is generated by a source node per session, in order to focus on the reconstruction performance of the measurements as in [4], while measurement times can be retrieved as described in Step 7. Thus, we evaluate the reconstruction error averaged over all sessions as in [2][4], defined for one session as $\epsilon = \frac{\|\hat{\mathbf{x}} - \mathbf{x}_0\|_2}{\|\mathbf{x}_0\|_2}$, where \mathbf{x}_0 is the vector of size N containing the original measurements for all N sensors, i.e., its corresponding components are in $\{-1, +1\}$ if sensor S_j is an origin node, and zero otherwise, while $\hat{\mathbf{x}}$ denotes the reshaped vector of collected measurements at the sink at the end of a session.

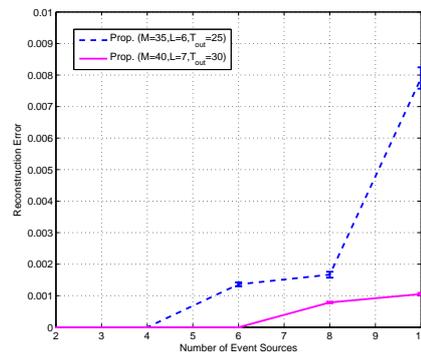


Fig. 3. Proposed protocol: reconstruction error with 95% confidence intervals.

Fig. 3 shows the average reconstruction error of our protocol for $K = \{2, \dots, 10\}$, for sequence length $M = \{35, 40\}$, $L = \{6, 7\}$ and $T_{\text{out}} = \{25, 30\}$. We observe that in both cases, excellent reconstruction is achieved even for larger values of K which imply lower sparsity in the number of superimposed data. By increasing T_{out} and M , more diversity gain and reconstruction accuracy are achieved, respectively.

As explained in the introduction, in the absence of data correlations to exploit, benchmark CS-based recovery algorithms as in [2][4] reduce to conventional forwarding protocols such as AODV routing and MAC scheduling that guarantee no collisions. Therefore, we compare the proposed protocol

⁴If $\mathbf{Q}(\text{Sink}) = \{\mathbf{a}_{j,l'}, \mathbf{a}_{j,m'}\}$ and $l' < m' \leq l < m$, $\tilde{x}_{j,l}$, $\tilde{x}_{j,m}$ may be equal to $\tilde{x}_{j,l'}$ or $\tilde{x}_{j,m'}$. Such cases rarely occur given the sparsity, and as packets with $l \geq L$ are discarded. But if they do, ambiguous packets are discarded.

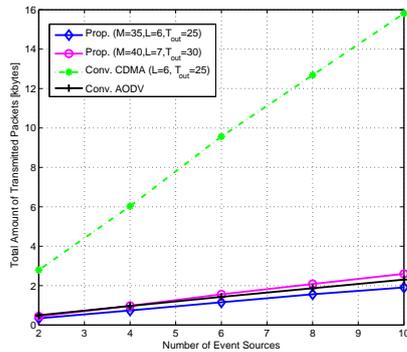


Fig. 4. Average required amount of transmitted packets per session [kbytes] with two reference algorithms: *Conventional AODV (Conv. AODV)* based on a simplified AODV routing with MAC scheduling, and *Conventional CDMA (Conv. CDMA)* based on the same flooding procedure as the proposed one but using NL-length sequences to guarantee orthogonality.⁵ In *Conv. AODV* based on [10], during each session, only the sources having measurements to transmit proceed to route discovery towards the sink, by flooding its neighbors with a Route Request packet of 24 bytes. The flooding continues until reaching the sink, which selects the shortest path route over which the Route Reply packet of 20 bytes is forwarded to the source. All other nodes in the same path memorize their route towards the sink, i.e., no route discovery procedure will be necessary when they become sources in turn. Once the route is fixed, each packet is forwarded to the sink. For simplicity, each packet will be forwarded sequentially by an ideal MAC scheduler (TDMA) that guarantees no collisions. The packet format is based on the IEEE 802.15.4 standard. Performance is only compared in terms of overhead and delivery delays, as both reference algorithms provide near-zero reconstruction error under this system model, given the absence of interference for the first one, and the orthogonal codes for the second. As overhead and data (origin node ID and measurement) are merged in the proposed and *Conv. CDMA* protocols, we evaluate the total required amount of transmitted packets for reconstruction.

For our protocol, the required amount of transmitted packets for reconstruction is $B_{Prop} = M P_{Prop}^{total}$ [bits], where P_{Prop}^{total} is the total number of packets transmitted within each session, including all the forwarded ones due to flooding. Similarly, for *Conv. CDMA*, we have $B_{CDMA} = N L P_{CDMA}^{total}$, with $L = 6$. Both P_{Prop}^{total} and P_{CDMA}^{total} are obtained by simulations and averaged over all sessions. Fig. 4 shows that our protocol requires a comparable amount of packets as *Conv. AODV*, and even less when small errors are allowed ($M=35$). Compared to *Conv. CDMA*, the gain of our protocol goes up to 90%.

Finally, we evaluate the average delivery delay required by each protocol, expressed in number of hops for each source measurement in Fig. 5. As in the proposed and *Conv. CDMA* protocols, each measurement is generated at a random time in a session, and all packets are received at the Sink during

⁵As the aim is to show the pure benefit of the proposed CS-based data recovery, only basic routing schemes are considered as in the references. Its performance may be further improved by optimized flooding.

T_{out} , it takes at most T_{out} hops to gather all measurements. Thus, Fig. 5 shows T_{out} divided by the number of source measurements, which gives an upper bound on the actual average per measurement delay of both protocols. The delay upper bound of the proposed protocol equals that of *Conv. CDMA* if minimal errors are allowed, while near-zero error may be achieved with a slightly higher delay. Compared to the delay of *Conv. AODV*, the delay for our protocol decreases rapidly as K grows, down to 75%. Thus, the proposed protocol provides near-zero reconstruction errors, while achieving large overhead and delivery delay savings against conventional protocols.

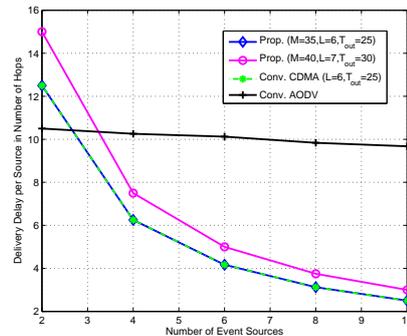


Fig. 5. Average delivery delay in number of hops per source measurement

V. CONCLUSION

We have proposed a novel CS-based protocol enabling non-scheduled, simultaneous transmissions for delivering sparse measurement data to the sink in multi-hop wireless sensor networks. Our protocol achieved low reconstruction errors, bringing significant savings in terms of overhead and delay, as compared to conventional routing and MAC scheduling strategies, as well as CDMA and flooding based protocol. The protocol will be extended to more general systems, with various topologies and asynchronous nodes.

ACKNOWLEDGMENT

We address our deepest gratitude to Assoc. Prof. Kazunori Hayashi, Kyoto University, for his invaluable comments.

REFERENCES

- [1] D.L. Donoho, "Compressed Sensing," *IEEE Trans. on Info. Theory*, vol. 52, no. 4, pp. 1289–1306, April 2006.
- [2] G. Quer et al., "On the Interplay between Routing and Signal Representation for Compressive Sensing in Wireless Sensor Networks," in *ITA Workshop*, 2009.
- [3] C. Luo, F. Wu, J. Sun, C.-W. Chen, "Compressive Data Gathering for Large-Scale Wireless Sensor Networks," in *ACM Mobicom*, 2009.
- [4] X. Wang et al., "Compressed Sensing Based Random Routing for Multi-Hop Wireless Sensor Networks," in *IEEE ISIT*, 2010.
- [5] L. Xiang, J. Luo, A.V. Vasilakos, "Compressed Data Aggregation for Energy Efficient Wireless Sensor Networks," in *IEEE SECON*, 2011.
- [6] J. Meng, H. Li and Z. Han, "Sparse Event Detection in Wireless Sensor Networks using Compressive Sensing," in *CISS*, 2009.
- [7] R. Mao and H. Li, "A Novel Multiple Access Scheme via Compressed Sensing with Random Data Traffic," in *IEEE WCNC*, 2010.
- [8] S.R. Bhaskaran, L. Davis, A. Grant, S. Hanly, P. Tune, "Downlink Scheduling using Compressive Sensing," in *ITW*, 2009.
- [9] M. Elad, *Sparse and Redundant Representations*. Springer, 2010.
- [10] C. Perkins, E. Belding-Royer and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," <http://www.ietf.org/rfc/rfc3561.txt>, IETF, RFC 3561, July 2003.