International Research Journal of Computer Science (IRJCS) ISSN: 2393-9842
Issue 11, Volume 4 (November 2017) www.irjcs.com

OBSTRUCTION CONTROL IN WSN BASED ON INTRUSION DETECTION SYSTEMS

Raghavendra Rao .B,
Research Scholar / Department of CSE,
Bundelkhand Institute of Engineering & Technology,
Bhundelkhand Univeristy, Jhansi
eliterbk@gmail.com

Dr.K.C.Roy
Director & Research,
Bundelkhand Institute of Engineering & Technology,
Bhundelkhand Univeristy, Jhansi

Manuscript History
Number: **IRJCS/RS/Vol.04/Issue11/NVCS10085**
DOI: 10.26562/IRJCS.2017.NVCS10085
Received: 27, September 2017
Final Correction: 30, September 2017
Final Accepted: 22, October 2017
Published: November 2017
Citation: Rao & K.C.Roy (2017). Obstruction Control in WSN Based on Intrusion Detection Systems. International Research Journal of Computer Science (IRJCS), Volume IV, 30-36. **doi://10.26562/IRJCS.2017.NVCS10085**
Editor: Dr.A.Arul L.S, Chief Editor, IRJCS, AM Publications, India

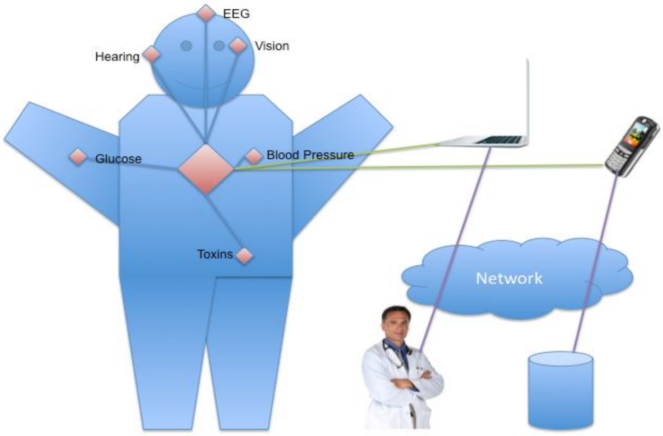
Abstract: WSN is a self organized network consisting of nodes. These nodes can have a small degree of movement due to medium on which they are deployed like body area networks where the sensors are fixed to body parts and the sensor exit movement due to actions of body like walking, running, sleeping etc. Since the nodes are moving, it is very difficult to maintain a reliable connection and when congestion happens in this network. In this work we propose a congestion control protocol based on the prediction of movement pattern of nodes and the data flow rate at movement stage to decide the congestion control strategy.
Keywords: WSN; IDS; Systems

I. INTRODUCTION

Recent application of WSN is in area of remote health monitoring. This application popularly called as WBAN works by coordination of multiple sensor fixed in various body parts. These sensors collect various parameters like blood flow rate, temperature, pulse rate etc and send the data to monitoring server which makes decision based on the data collected. In this kind of applications due to nature of medium the sensors exhibit various degree of movement. Say for various human activities like walking, swimming, sleeping, running the relative position of sensor changes and the routing path changes. Also the data generation rate various for different human activities and when congestion happens it is difficult to manage with traditional congestion control protocols which don't consider this varied degree of movement in Wireless Body area networks(WBAN).

Previously many studies have been made on congestion control in wireless sensor networks. Studies proposed solutions in different layers like physical, network and transport layers and also cross layers to detect and reduce congestion. Previous works can be classified into following types

1. Rate Based
2. Buffer Based
3. Priority Based
4. Cluster Based
5. Multipath routing based



Rate based algorithms estimates the number of flows from upstream nodes and modulates the rate of packet flow. Buffer based algorithms tune their transmission rate and time based on the buffer occupancy of nodes in the routing flow. Priority based algorithms assigns different priorities to flow and make their forwarding decisions based on the priority. Cluster based algorithms decentralize the congestion control to are of scope by clustering the network and manage congestion. Multi path routing algorithms divides and forwards packets across multiple path to reduce the congestion. In case of WBAN, even though nodes are having various degree of movement, the mobility is not completely random. In most deployments of this kind of sensor network, there is a pattern in the movement of nodes. Say the human user on which the WBAN is deployed has regular pattern like sleeping after eating, running is early morning etc. There is also a periodic behavior exhibited by the users, if this periodic movement can be used to predict the sensor position and based on it, the congestion control strategy can be formulated. In this work we use HMM to model the movement of WBAN and use the model result to predict the position of sensor and based on it congestion control is done.

II. RELATED WORK

In this section we survey the current congestion detection and control protocols. CODA [1] is an energy conserving and efficient control technique that is designed to solve congestion in the upstream direction i.e., the sensor to sink direction. It involves of two main schemes: 1) open loop hop by hop backpressure mechanism. 2) closed loop multisource regulation. The detection method in CODA is the receiver based congestion detection. It considers a combination of both present & past loading conditions of the current buffer occupancy in the receiver. If the occupancy exceeds the threshold value, then congestion is inferred. The node detecting the congestion will notify its upstream neighbors to reduce the flow by backpressure mechanism. CODA detects congestion based on queue length and wireless channel loading. It uses AIMD rate adjustment technique and jointly used end-to-end and hop-by-hop controls for regulation.

Event to Sink Reliable transport is a unique transport solution that is designed to achieve reliable event detection with minimum energy expenditure and congestion resolution [2]. This technique overcomes on of the disadvantages of CODA. ESRT works based on two parameters: Event reliability and reporting frequency. Event reliability is defined as the number of data packets received at the decision interval at the sink. The end-to-end data delivery services are regulated by adjusting the sensor report frequency. If the reporting frequency is too low, the sink will not be able to collect enough information to detect the events. But on the other hand, if the reporting frequency is too high, it endangers the event transport reliability. ESRT adjusts the reporting frequency such that the observed event reliability is higher than the desired value to avoid congestion. The congestion detection in ESRT is by local buffer level of the sensors nodes. The sensor node adds a congestion notification bit on the packet's header when congestion occurs. When the sink receives this CN bit, it knows that congestion has happened in WSN.

Congestion control for Multiclass Traffic (COMUT) is a framework that consists of a distributed and scalable congestion control mechanism. It is based on self organisation of networks into clusters. Each cluster is equipped with a sensor that is autonomously monitors congestion within its scope[3]. These networks are designed to support multiclass of traffic in WSN's. Each cluster is governed by a sentinel. These sentinel roles are assigned to sensors to proactively monitor the system and collect the event rates that is used to infer the combined level of congestion. The local traffic is reported by the sensors to the sentinel en-route a local broadcast system. The sensor rates per cluster are regulated by exchanging only small amounts of control information via regulator packets between the sentinel sensors alongside the flow path.

Congestion control for Sink to Sensors (CONSISE) [4] is a technique that works downstream i.e., from the sink to sensor direction. Conventionally, congestion happens in the sensor-to-sink direction but, the reverse is also possible.

The reasons are broadcast storm problem that refers to higher levels of collision that occurs on a series of local broadcast and reverse path traffic from sensors to sink. Congestion in the sensor-to-sink direction will not be rare if WSN is built over CSMA/CA type of MAC and flooding based routing protocol.

Priority based congestion Control protocol (PCCP)[5] is an upstream congestion control protocol in WSN which measures congestion degree as the ratio of packet inter arrival time to the packet service time. It is designed in a way that the data packets have a guaranteed weighted fairness so that sink can get different throughput from the sensor nodes but in a weighted way. PCCP is intended to improve energy-efficient and support traditional QoS in terms of latency, throughput and packet loss ratio. PCCP can be of three components: 1) Intelligent Congestion Detection (ICD). 2) Implicit Congestion Notification (ICN). 3) Priority-based rate adjustment.
In CCF[6] algorithm each node measures the average rate r at which packets can be sent from the node, divide the rate among the children nodes, adjust the rate if the queue is overloaded and propagate the rate downstream. It is designed to work with any MAC protocol in the data link layer and it exists in the transport layer. CCF uses packet service to deduce the availability of the service rate. It controls congestion in a hop-by-hop manner and each node uses exact rate adjustment based on its available service rate and child node number. It has two major disadvantages: The rate adjustment is based on packet service time which leads to low utilization as it has significant packet error rate. It cannot allocate the remaining effective capacity as it uses work conservation scheduling algorithm.

EB works in similar fashion to CODA. It uses congestion control in tree routing structure to all data sources to a sink. It uses the hop-by-hop backpressure mechanism. EB works in three steps: 1) each node calculates the average rate at which the data packets can be sent. 2) The node then divides the average data rate in to the number of children nodes to give the per-node data packet generation rate and adjusts the rate if the buffer is overflowing. 3) The node then compares the data rate of two children nodes with the parent nodes. The smaller rate among the two values is propagated such that data sources do not send packets beyond the minimum rate supported by the nodes along the path to the sink. SenTCP[7] is a transport protocol that uses open loop hop-by-hop Congestion Control. It has two distinct features that it adopts while detection. It detects congestion using local Congestion degree and uses hopby-hop for control[4]. The features include: 1) SenTCP conjointly uses average local packet service and average local packet inter-arrival time. These features determine the current local congestion degree in each intermediate sensor nodes. They effectively help to differentiate the reasons for packet loss and delay in wireless communication. 2) Each intermediate node issues a feedback signal backward and hop-by-hop control. This signal carries buffer occupancy ratio and local congestion degree. These parameters are used to adjust the sending rate of the neighbouring nodes in the transport layer[1][10]. SenTCP realizes higher throughput and good energy efficiency since it reduces packet dropping by hop-byhop. The major disadvantage of SenTCP is that it guarantees no reliability.

Pump slowly and Fetch Quickly (PSFQ) [8] control protocol aims at distributing data from sink-to-sensors i.e., it belongs to the downstream reliability guarantee. PSFQ is a mechanism that is proposed for reprogramming a group of sensors. PSFQ is based on slowly injecting packets into the network "pump operation" and performing aggressive hop-by-hop recovery in case of packet loss "fetch operation" and selective status reporting "reporting operation". The disadvantages of PSFQ include: 1) since it uses hop-by-hop recovery, it requires more buffer space. 2) The transmission of data packets is relatively slow in operation and hence there is large delay in the system. 3) PSFQ cannot detect a loss of single packets individually as it uses NACK signals for indication and the entire block is re transmitted upon request. 4) It cannot be used in the forward direction and does not address packet loss due to congestion.

III. PROBLEM DEFINITION

Given a wireless body area networks with N sensors located at different parts of body. The N sensors relative position changes due to various movements of body like walking, sitting, sleeping, running etc. The sensor sense parameters in their deployed area and forward via multi hop to a monitor server. The multi hop routing path varies every time due to relative position change of sensors and different sensors have different sampling rate. Due to different rate and sensor position change congestion happens in network. In this work we propose a effective solution to solve this problem of congestion.

IV. PROPOSED SOLUTION

The proposed solution consist of following parts

1. Prediction of link quality
2. Devising of congestion free path
3. Rate control to limit the congestion

Prediction of Link Quality

Body movement consists of various stages like walking, running, jumping, sleeping, sitting, falling etc.

The sensors position map for various stages is different and the volume of traffic from sensors at these stages is also different, so the first step in modeling the link quality is to predict the next stage. Here deviate from the existing works, instead of predicting link quality over a time for entire network, we split prediction over a time for different stages. For each stage, the sensor position map is determined and initially for some time intervals SH routing is used and link quality matrix is formed. The link quality is measured in terms of interference of the link. Over a time period the link quality matrix for different stages are formed and ARIMA model is employed to detect further link quality for that stage. For detection of stage, a tri axial accelerometer is fixed in one sensor of body area networks. From the sensor readings, over every time window W, following features are calculated.

M_{mag}, V_{mag} – mean of magnitude and angle
M_{ang}, V_{ang} – variance of magnitude and angle
E_{mag},E_{ang} – entropy of magnitude and angle

These features are mapped to different stages and a neural network is trained. The neural network will have 6 input neurons for the corresponding six features and 6 output neurons for six corresponding stages [walking, running, jumping, sleeping, sitting and falling]. Further stages can also be added by increasing the output neuron and training for that stage. The features captured every W window period for a total time of T, say 1 day is used to train the neural network. The neural network is trained using back propagation algorithm. We use three layer back propagation network with the number of hidden layer neuron calculated as 2*input neuron + 1.

With neuronal network, the stage can be detected for every time window W, and from the sequence of stages, the next stage is predicted using HMM algorithm. HMM is trained with 6 states with one 1 state for 1 stage. 1 day sequence of stage is used to train the HMM using viterbi algorithm. From this HMM interval, the stage for next time window W can be predicted. The link quality is measured in every stage over a period of time interval and using past observations, ARIMA model is trained to predict link quality for next time period. To model the link quality , we measure the packet success ratio in the links.

Packet success ratio = No of success transmission / Total number of Packet attempted.

Sensor position map provides the location of sensors. The sensor position map is calculated for different stages. This sensor position map will be used to find the route in the corresponding stage.

Procedure 1: Procedure for Building Training Set for Stage Detection
1. Launch a stage Activity (S) for a Time period T (Say Sitting etc)
2. Capture the accelerometer value(amplitude, angle) for Time Period T
3. Divide the period T in W size intervals.
4. For each W interval calculate Mmag, Vmag, Mang,Vang, Emag,Eang
5. Add Mmag,Vmag,Mang,Vang,Emag,Eang=> S to training data set
6. Do step 1 to 5 for each stage. (walking, sleeping etc)

Procedure 2: Procedure for Neural Network Training
1. Create Neural Network (6,13,6) – 6 input, 13 intermediate , 6 output
2. For each record in training set (created using Procedure 1) training Neural Net
3. Retrain Neural Network till the Error is less than 0.01 or maximum iteration of 1000.

Procedure 3: Stage Detection
1. Capture accelerometer value (amplitude, angle) for W interval
2. Calculate Mmag, Vmag, Mang,Vang, Emag,Eang for the W interval data.
3. Invoke on Neural Network (created by Procedure with Mmag, Vmag, Mang,Vang, Emag,Eang
4. Return the stage for corresponding output neuron triggered.

Procedure 4: Link Quality Collection
1. Collect Average Link quality at every pair of links at every W interval
2. Detect stage S for W interval (using Procedure 3)
3. Store the Average Link Matrix into Queue of Stage S.
4. Do this for a Time interval T

Procedure 5: Link Quality Prediction Training
1. For each Stage S do step 2 and 3.
2. Collect all Average Link Matrix collected in Time interval T
3. Build ARIMA model for each pair of link over all the data collected in time interval T

Procedure 6: HMM Training
1. Collect Sequence of stage in every W interval for 1 complete day
2. Create 6 states [corresponding to each stage]
3. Traverse the Sequence of Stage over the 6 states and calculate transition probability [Viterbii algorithm]

Devising of congestion free path

The stage of construction of congestion free path works only for the time period W and if a node has further data to send even after interval of W, the path must be re computed.

Procedure 7: Prediction

1. Invoke HMM model (learnt from Procedure 6) to get Predicted Stage
2. Invoke on ARIMA model for stage S for each pair of Link to get predicted Link Quality

Once the link quality is predicted for each link in network, the congestion free path must be constructed in an on demand manner. When a node wants to send packet stream to sink, it will send RouteReq to node with link quality above a threshold. Every node which receives the RouteReq will check if the rate of data packet can be accommodated and will forward RouteReq to node whose predicted link quality is above a threshold and only if it can accommodate the rate of data transfer. Target node receiving the RouteReq will send RouteRes to the source node. When the RouteRes is received at intermediate node, they forward only if the rate of data packet can be accommodated. When the RouteRes is received at the source node, it will select the sink which is close to it.

When sending the data packet through the path selected, the data packet has a field to mark first packet, intermediate packet or last packet. When the first packet marker is set, all the intermediate nodes reserves the rate and it clears based on the two conditions

1. Data packet with last packet marker is set
2. Intermediate packets are not received in a timeout period.

Each sensor node has varied data rate based on the applications running on it. So sensor nodes better know the rate. In RouteReq, we add a parameter rate and the intermediate nodes broadcast RouteReq and Route Res only if they are able to satisfy the data rate requested. In each sensor node there is a queue for buffering the data packets from each node and the size of queue is allocated based on the rate requested. If the nodes sends packet exceeding their rate, then packets would be dropped only for that node and it will not affect all other nodes traffic through it.

Rate control to limit the congestion

Sometimes due to processing at node, the queue reserved for a flow may get full. For each queue, expected packet service time is calculated when allocated and when nodes packet service time is continuously higher than expected packet service time, than one of two decisions is taken

1. If the buffer memory for queue enlargement is available, then queue size is increased and the expected packet service is updated.
2. If the buffer memory for queue enlargement is not available, the data packet or ACK packet to the sender is marked with reduce rate field to reduce the data rate.

By this rate adaptation is done and the congestion is avoided.

V. RESULTS

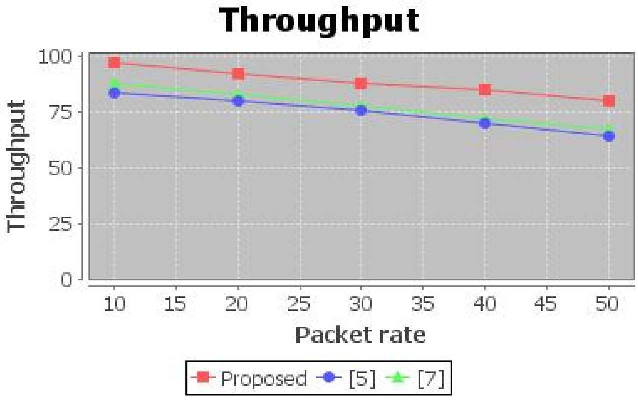
To test the performance of proposed solution we implemented the solution on Jprowler wireless sensor network simulator. The simulation was conducted with following parameters

Parameters	Values
Number of Nodes	100 to 200
Communication range	100m
Area of simulation	1000m*1000m
Packet Rate	10 to 40 packet per second
Simulation time	30 seconds
Interface Queue Length	50
MAC	802.11
Movement Speed	5 m/ sec

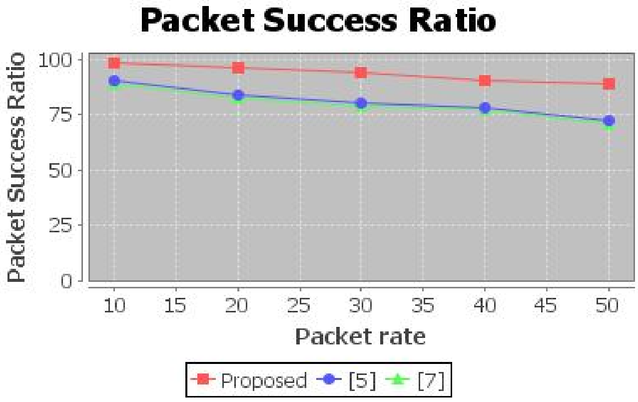
At each time interval of 5 sec, 10 nodes generated packet with rate of 10 to 40 as configured and maintained the traffic rate for 5 sec. The proposed solution is compared with priority based congestion control scheme mentioned in [5] and senTCP mentioned in [7]. We measured following parameters

1. Throughput
2. Packet Success ratio
3. Packet Delay
4. No of congestion points

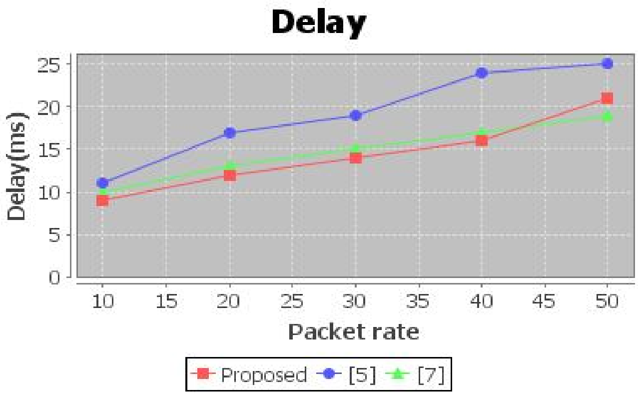
Throughput is measured as the number of packets received at target nodes. Packet success ratio is the ratio of number of packet received successfully at target nodes to the number of packets sent. Packet delay is the end to end delay for packet traversal from source to target node. A node is congestion point if its interface queue occupancy is more than 90%. Throughput is calculated by increasing the packet rate from 10 to 50 insteps of 5 and results are plotted below. From the results we see that proposed solution achieves better throughput than other solution.



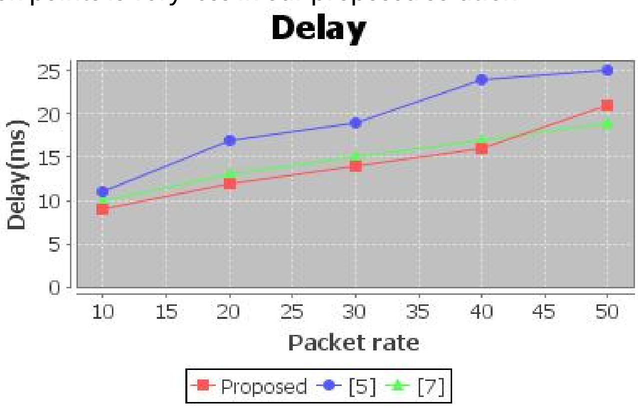
Packet success ratio is calculated by increasing the packet rate from 10 to 50 insteps of 5 and results are plotted below. From the results we see that proposed solution achieves better success ratio, the reason being the number of lost packets reduced due to better management of packet queues.



Packet delay is calculated by increasing the packet rate from 10 to 50 in steps of 5 and the results are plotted below. From the results we see that the proposed solution has comparatively higher delay the reason being queuing delay in nodes



We measured the number of congestion points by varying the packet rate from 10 to 50 in steps of 5 and the results show that number of congestion points is very less in our proposed solution.



VI. CONCLUSION

In this work, we have proposed a congestion control mechanism which is able to predict the link quality for various movement stages using HMM Model. The model was able to predict the movement accurately and with rate control clubbed with congestion free path decided based on the link quality, the quality of service of the network is improved in terms of packet delivery ratio and delay.

REFERENCES

1. Chien- Yin Wan, Shane B. Eiseaman, Andrew T. Champbell, "CODA: Congestion Detection and Avoidance".
2. Ozgur B. Akan, Ian F. Akyildiz, "Event-to-Sink Reliable Transport in wireless Sensor Networks". IEEE/ACM transactions on Networking. Vol. 13, No.5, October 2005
3. Kyriakos Karenos, Vana Kalogeraki, Srikanth V. Krishnamurthy. "Cluster-based Congestion control for sensor networks".
4. Ramuja Vedantham, Raghupathy Sivakumar, Sueng – Jong Park, "Sink-to-Sensors Congestion Control Strategy".
5. Mohammad Hossein Yaghmaee, Donald Adjeroh. "A New priority Based Congestion Control Protocol for Wireless Multimedia Sensor Networks". IEEE 2008.
6. Swastil Brahma, Mainak Chatterjee, Kevin Kwiat," Congestion Control and Fairness in Wireless Sensor Networks".
7. C. Wang, K.Shoraby, B.Li, "SenTCP: A hop-byhop Congestion Control protocol for Wireless Sensor Networks" in IEEE INFOCOM 2005, USA, March 2005.
8. C. Y. Wan, A.T. Campbell, "PSFQ: A reliable transport protocol for Wireless Sensor Networks" in proceedings of ACM WSNA'02, September 28, 2002, USA.
9. K.Chenand K.Nahrstedt. EXACT: An Explicit Rate-based Flow Control Framework in MANET (extended version). Technical Report UIUCDCS-R-2002-2286/UILU-ENG-2002-1730, Department of Computer Science, University of Illinois at Urbana-Champaign, July 2002.
10. K.Chen and K.Nahrstedt. Limitations of Equation-Based Congestion Control in Mobile Ad Hoc Networks. In ICDCSW '04: Proceedings of the 24th International Conference on Distributed Computing Systems Workshops W7: EC, pages 756–761, 2004. Http://DOI:10.1109/ICDCSW.2004.1284118.
11. K.Chen, K.Nahrstedt, and N. Vaidya. The Utility of Explicit Rate-based Flow Control in Mobile Ad Hoc Networks. In WCNC '04: Proceedings of the IEEE Wireless Communications and Networking Conference, volume 3, pages 1921–1926, Mar. 2004.
12. K.Chen, Y.Xue, and K. Nahrstedt. On Setting TCP's Congestion Window Limit in Mobile Ad Hoc Networks. In ICC '03: Proceedings of the IEEE International Conference on Communications, Anchorage, Alaska, May 2003. http://DOI:10.1109/ICC.2003.1204525
13. C.de M.Cordeiro, S. R. Das, and D. P. Agrawal. COPAS: Dynamic Contention-Balancing to Enhance the Performance of TCP over Multi-hop Wireless Networks. In Proceedings of the 10th International Conference on Computer Communication and Networks (IC3N), pages 382–387, Miami, FL, USA, Oct. 2002.
14. R.de Oliveira and T. Braun. A Delay-based Approach Using Fuzzy Logic to Improve TCP Error Detection in Ad Hoc Networks. In WCNC '04: Proceedings of the IEEE Wireless Communications and Networking Conference, volume 3, pages 1666–1671, Mar. 2004.
15. R.de Oliveira and T. Braun. A Dynamic Adaptive Acknowledgment Strategy for TCP over Multihop Wireless Networks. In INFOCOM '05: Proceedings of the 24th Annual Joint Conference of the IEEE Computer and Communications Societies, volume 3, pages 1863–1874, Mar. 2005. http://doi:10.1109/INFCOM.2005.1498465