

# Network Lifetime Maximization of Sensor Network Based on Energy Aware Source Tree Routing

Shivshanker.P.Biradar<sup>1</sup>

Dept of ECE, LAEC, Bidar, Karnataka, India.

Email: shivdurga.biradar@gmail.com

Dr. T.S. Vishwanath<sup>2</sup>

Dept of ECE, BKIT bhalki, Karnataka, India.

Email:tsvrec1@rediffmail.com

## ABSTRACT

Wireless sensor network is a wireless network of autonomous sensor nodes where objective is to acquire data from different sensor nodes called sources in a central node called base station or server or sink. Several routing techniques are proposed for sensor networks. It includes shortest path routing, gossiping, flooding and so on. But Spanning tree is considered as good topology control mechanism for sensor network. Most of the literature emphasizes on generating the tree where sink is the root node. Such trees are called sink based tree. However sink based trees has maximum spreading. If congestion occurs at any node joining lower nodes to sink than sink cannot acquire data from nodes at lower level at tree. In order to avoid this problem, we propose a source based tree. Each of the source node generates a tree till sink. Therefore there are more than one tree present in the network and sink is connected to source through different paths. Each node upon getting a route request checks if it has maximum nodes connected to it. This list is called a MPR list. If it has maximum connectivity, it can acquire data from many nodes, aggregate them and forward it to sink. Thus our work consists of many trees from source nodes to sink where every intermediate node has maximum connectivity. Therefore the tree is most expanded. We further implement backup route creation to prevent sudden connectivity loss. Multiple paths from each source to sink ensures maximum data delivery. As power and energy loss is a major concern in such a network, we adopt an Energy aware tree creation to ascertain that all the intermediate nodes have maximum energy. Through results we prove that the proposed system performs better than sink based tree in terms of packet delivery ratio, throughput, low average energy consumption and elongated lifetime.

Keywords - Wireless Sensor Networks, Congestion Control, Source Based Trees, Sink Based Trees, OMNeT++ software.

Date of Submission: Aug 04, 2018

Date of Acceptance: Aug 21, 2018

## I. INTRODUCTION

What is the fastest rate at which we can collect a stream of aggregated data from a set of wireless sensors organized as a tree? We explore a hierarchy of techniques using realistic simulation models to address this question. We begin by considering TDMA scheduling on a single channel, reducing the original problem to minimizing the number of time slots needed to schedule each link of the aggregation tree. The second technique is to combine the scheduling with transmission power control to reduce the effects of interference.

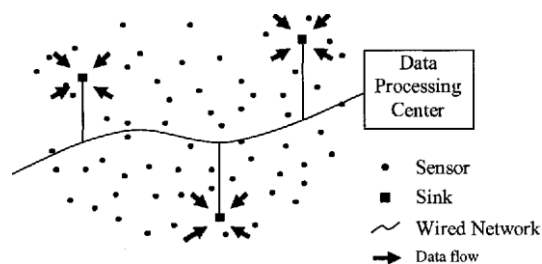


Fig.1 Wireless sensor network.

The maximum degree of the routing tree ensures prolong data delivery. Therefore we consider finally how the data

collection rate can be further enhanced by the use of degree-constrained routing trees. Considering deployments at different densities, we show that these enhancements can improve the streaming aggregated data collection compared to the baseline of single-channel data collection over non-degree constrained routing trees (Source Trees).

### 1.1 Spanning tree based algorithms for low latency and energy efficient data Routing and Congestion Control

Many wireless sensor networks (WSNs) employ battery-powered sensor nodes. Communication in such networks is very taxing on its scarce energy resources. Converge cast – process of routing data from many sources to a sink – is commonly performed operation in WSNs. Data aggregation is a frequently used energy-conserving technique in WSNs. The rationale is to reduce volume of communicated data by using in-network processing capability at sensor nodes. In this work, we address the problem of performing the operation of data aggregation enhanced converge cast (DAC) in an energy and latency efficient manner.

We present a source based variant of the Minimum Spanning Tree (MST) algorithm with multiple paths and compare with a variant of the Single Source Shortest Path Spanning Tree (SPT) algorithm.

The algorithm tries to construct an energy optimal tree. The nodes of these trees are scheduled for collision-free communication using a channel allocation algorithm. To achieve low latency, these algorithms use the b-constraint, which puts a soft limit on the maximum number of children a node can have in a tree. The tree obtained from energy minimizing phase of tree construction algorithms is restructured using the b-constraint (in the latency minimizing phase) to reduce latency (at the expense of increasing energy cost). The effectiveness of these algorithms is evaluated by using energy efficiency, latency and network lifetime as metrics.

## II. ANALYSIS

A shared, core-based tree is an efficient topology control solution in WSNs. Usually, such trees are build as spanning trees using the sink as the root and all nodes forward their data using this structure. Using the sink node as the root is optimum, since the sink in WSNs is, most of the times, a robust node that does not suffer from power limitation issues. Thus, this situation defeats the main disadvantage of shared, core-based tree, which is the fact that core nodes can become single points of failure. The creation of a sink-based tree is normally a simple procedure that begins after the network discovery phase. Nodes become part of this tree at the initialization of the network and maintain their position in it until topology changes happen.

In this work we consider that the following assumptions exist:

- Nodes communicate only with nodes that are one hop away.
- Dense placement of nodes achieves the coverage of the network.
- Nodes route packets only to the nodes that are one level higher than them.

### A. Existing System

Sink determines number of sources it wants to acquire data from. Then it generates a RREQ for every independent source. Nodes that have answered a RREQ are given more weight over singular nodes while forwarding the packet. Thus a directed graph is created that starts from sink and expands to every source. When sources start data forwarding to the sink node, intermediate nodes suffer buffer overflow in CDMA based system. As no child of a node has information about other children, packet collisions are unavoidable. Further in case of a single failure, entire tree needs to be constructed. Hence subsequent routing increases control packet which results in significant energy loss. In order to avoid this scenario, we propose a source tree based routing.

### B. Proposed System

We evaluate the impact of transmission power control, multiple paths, and routing trees on the transmission performance for raw-data converge cast..

Although the techniques of transmission power control and multi-path transmission have been well studied for eliminating interference and reducing congestion in general wireless networks, their performances for bounding the completion of data collection in WSNs have not been explored in detail in the previous studies. The fundamental novelty of our approach lies in the extensive exploration of the efficiency of transmission power control and residual energy based bound in obtaining the path.

The data collection rate often no longer remains limited by scheduling but by the topology of the network. Thus, in the final step, we construct network topologies with specific properties that help in further enhancing the rate. Our primary conclusion is that, combining these different techniques can provide an order of magnitude improvement for raw converge cast where many nodes sends data to a single node.

Proposed work is a greedy, tree-based, multi-channel protocol for data collection applications. It partitions the network into multiple sub trees and minimizes the intra tree interference and congestion by assigning different paths to the nodes residing on different branches starting from the top to the bottom of the tree. Figure shows the same tree given in Fig.2 which is scheduled according to tree for raw data collection. Frequency is defined as the rate at which a node can forward data to it's root. Here, the nodes on the left most branch is assigned frequency F1, second branch is assigned frequency F2 and the last branch is assigned frequency F3 and after the path assignments, time slots are assigned to the nodes.

### C. Sink-Based Tree Creation

The first phase of network establishment is network discovery. In this phase, the major task of every node is to discover its neighbor nodes in the network. A common method for network discovery is some sort of flooding algorithm, initiated and controlled by the sink. After the end of this phase each node maintains a routing table in which the IDs of the neighbors nodes are kept, as well as any other information requested by the algorithm's designer. An example of the end of initial phase is presented in Figure 3 in which the nodes are aware of all of their one-hop neighbors.

Tree based data forwarding

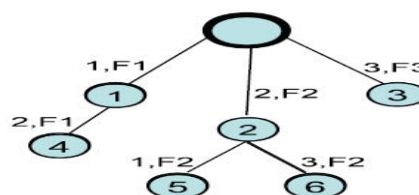


Fig.2 Packet transmission along Tree

Phase 2 of the tree creation begins from the sink and the target is to reduce the initial topology and place nodes in levels from the sink to the edge nodes. The process initiates with the sink broadcasting a "Hello" message with the level field set to 0. Nodes that receive this message update their level field to 1 and re-broadcast this message. Each node that receives a "Hello" message, updates its level field and broadcasts this message accordingly. If a node receives more than one "Hello" messages, it updates its level field based on the lower level. Thus, if its level changes to a lower value, it informs the nodes around it and the process iterates until all nodes are set in levels. Hence, a spanning tree is created. As shown in Fig. 4, creating a level-based, shared tree, leads to a "relaxed" topology (with a lower degree of connectivity). Nodes communicate with a lesser number of nodes, a fact that renders the network more capable to avoid and control congestion situations (either in the medium or in the buffers) while the source-to-sink packet time is reduced to the minimum, since a spanning tree is created.

*D. Source Based tree creation*

Source-based trees introduce a completely different routing concept in comparison with sink-based tree. The biggest difference lies on the fact that source-based trees, abandon the shared, core-based, philosophy and move to distributed solutions. Source-based trees are constructed on demand (when a node becomes a source node) and continue their operation until nodes stop being sources. Hence, when a node senses that is becoming a source node (receives analog data) it starts building its routing tree from itself (source) to the sink.

Building a proper and well tuned routing tree from a source to the sink is not a straightforward procedure as with shared, core-based trees. The reason lies on the fact that source-based trees do not "reduce" the topology as shared, core-based trees do. On the contrary, they incur significant overhead since every node is possible create its own tree, due to the fact that every node can become a source.

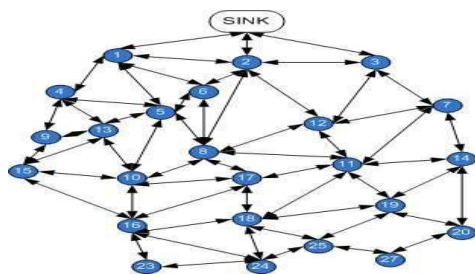


Fig.3 Initial Network Connectivity (before topology control algorithm applies)

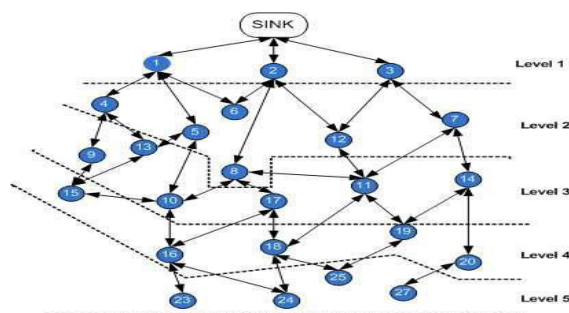


Fig.4 Phase 2 of Network Connectivity (after topology control algorithm applies)

**III. SOURCE-BASED TREES IN CONGESTION CONTROL ALGORITHM**

A normal question that arises, is why to employ source based trees for any reason, since sink-based trees are efficient while source-based trees present several disadvantages. Before answering this question we shortly present the ways that congestion is controlled in WSNs.

Generally two methods exists with which algorithm designers attempt to control congestion in WSNs. The first method is called "traffic control". Algorithms that employ the traffic control method, adjust the traffic that is injected to the network by the different sources, in order to cope with the capacity of the currently employed paths. [5], [6], [7], [8].

The second method is called "resource control". This method, the network takes advantage of the redundant deployment of sensor nodes in the field and employs the resources (buffer, power) of nodes that do not participate in the initial routing paths to forward data through them. Thus, sources do not reduce the rate with which they inject packets in the network and excess packets are forwarded through alternative or multiple paths [2], [9], [10]

*A Issues of Source-Based Trees*

To construct source-based trees which can fulfill their purpose we consider a small set of "critical" parameters:

**Location Awareness** – Nodes must be aware of their position in relation to the sink. Localization and positioning in WSNs is a subject that attracted a lot of research work [11] and solutions already exist in the literature. Hence, if a node is aware of its position then it selects as neighbors only nodes that are closest to sink than itself. For example in Fig. 5 node 27 will not be selected by node 25, due to its position further away from the sink, even though it is a level 2 node.

**Higher Level Connection Availability** – A node is kept on the tree only if it is connected to at least one node at a higher level than itself. For example in Fig. 5 node 14 will remain out of the tree since it is not connected to a node at a higher level.

**Number of nodes kept in neighbor table** - In dense placements this number could be very big and will create significant overhead problems to the network. Thus, a maximum number of neighbors must be set. According to [12] a value of six neighbors seems to be optimal. Also a minimum number of two must exist. If a node is connected to less than two higher level nodes, it must add in its neighbor table nodes which are at the same level but nearest to the sink. If such nodes do not exist, then this node should not be considered as a good neighbor for data forwarding.

Employing these parameters in the construction of source based routing tree all nodes are connected to upstream nodes and each node provides at least two alternative paths. Nodes that do not comply with the parameters discussed above are not part of this routing tree as shown in Fig.5. Of course, due to the dynamic nature of WSNs, is possible that a node which is not at the routing table the present moment, to become part of it, in case of topology changes.

Also, although Fig. 5 seems even simpler than sink based tree topology, sink based tree topology is a global, shared topology while source based tree (Fig. 5) concerns just one source node, out of possible, all nodes in the network. This means that if many sources exist in the network there could also exist many different trees that share the same nodes.

#### IV. EVALUATION

To compare the performance of source based against sink based trees, concerning resource control algorithms for congestion control in WSNs, we performed a series of simulations.

##### A. Network Model

We assume that there is a WSN, where nodes are initially deployed uniformly. One sink is located at a specific point in the network. Each node is considered that is aware of its position in relation to the sink. In our network deployment we consider that all nodes (except the sink) are identical and “carrier sense multiple access with collision avoidance” (CSMA/CA) is employed as MAC protocol. Moreover, in order to compare the performance of source and sink based trees when the network faces congestion, we employed as a resource control reaction mechanism, the alternative path creation scheme of Hierarchical Tree Alternative Path (HTAP) algorithm [9].

HTAP is a resource congestion control algorithm in WSNs and consists of four schemes:

- Topology control
- Hierarchical Tree Creation
- Alternative Path Creation
- Handling of Powerless (dead nodes)

We stress that we have employed just the

alternative path creation scheme, which is described below.

Alternative path creation runs when the buffer of a node starts filling with a rate with which buffer occupancy is increasing. In this case each node runs locally a lightweight congestion detection (CD) logarithm. When the buffer occupancy reaches a specific value, CD algorithm starts counting the rate with which packets are reaching the node. Since each packet is identified by the Node ID in its packets header, CD algorithm is aware of the nodes that are transmitting packets through this node, as well as, for their data rate.

##### B. Simulator setup

For our simulations we employed the OMNeT++ simulator. The radio propagation and transmission models are given by:

$$P_{rec,ideal}(d) \leftarrow P_{transmit}(1/(1=d^\gamma)) \quad \text{-----(1)}$$

Where,  $2 \leq \gamma \leq 4$

$$P_{rec}(i,j) \leftarrow P_{rec,ideal}(d_{i,j})(1+a(i,j))(1+\beta(t)) \quad \text{-----(2)}$$

Where  $P_{transmit}$  is the signal strength at the transmitter and  $P_{rec,ideal}(d)$  is the ideal received signal strength at distance  $d$ ,  $a$  and  $\beta$  are random variables with normal distributions  $N(0, \sigma_a)$  and  $N(0, \sigma_\beta)$ , respectively. A node  $j$  can receive packets from node  $i$  if  $P_{rec}(i, j) > \Delta$  where  $\Delta$  is the threshold. In our simulations we use the following default simulator parameters:

- $\sigma_a = 0.5$
- $\sigma_\beta = 0.03$
- $P_{error} = 0.05$
- $\Delta = 0.1$

The rest of the parameters reflect Mica-Z node specifications, the most important of which are presented in Table 1.

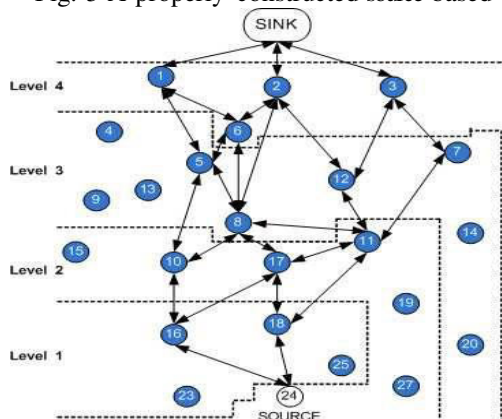
Table 1: Simulation Parameters

Max Data Rate (kbps)	128
Transmission Power (dbm)	2
Receive Threshold (dbm)	-74
Transmission Current (mA)	17.4
Receive Current (mA)	19.7
Fragment Size (bit)	1024
Buffer Size(Bytes)	512K
MAC layer	CSMA/CA

##### C. Results & Discussion

An initial number of 40 nodes, are uniformly placed on a square grid over an area of 400x400m. All nodes communicate only with the nodes that are one hop always. In this simulation series we have programmed every source to transmit constantly 100 packets/s while we increase the number of sources. Using these simulation settings we expect to analyze the performance of the network when the data load is increasing.

Fig. 5 A properly constructed source based tree



Results in Fig.6 indicate that as the number of sources in the network increases, the source-based tree topology is more able to increase the packet delivery ratio at sink node compared with the sink-based topology. This is because when the sources increase, the sink-based topology provides a limited number of alternative paths to each node. But whereas source based trees provides multiple paths to sink from each source node to divert the traffic.

Results in fig.7 indicate that as the number of sources in the network increases, the source-based tree topology is more able to increase the sink throughput compared with the sink-based topology. The reason lies on the fact that when the sources increase, the sink-based topology, which is a shared topology, provides a limited number of alternative paths to each node. Thus, when congestion occurs and the capacity of the routing tree is exceeded, alternative paths are limited and the throughput reduces.

Results in fig 8 indicate lifetime of source based routing tree is better compared to sink based routing tree. Lifetime indicates the duration after which a node dies in the network once the simulation starts. Lifetime of sink based trees goes on decreases with increase in number of source nodes due to limited number of alternative paths.

Result in fig.9 indicates as the number of sources increases in network then latency of source based routing tree is better compared to sink based routing trees. Since source based trees provide multiple paths from each source node to sink which avoids the congestion problem in network.

Results in fig.10 indicates as number of sources increases in the network then average energy remaining of source based routing tree is better because it avoids congestion by providing multipath from each source node to sink. This multipath avoids congestion and retransmission of packets.

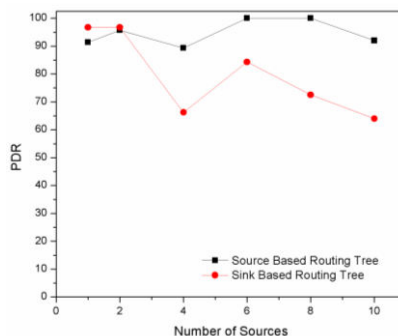


Fig.6 A graph of number of sources versus packet delivery ratio (PDR) of source and sink based routing trees.

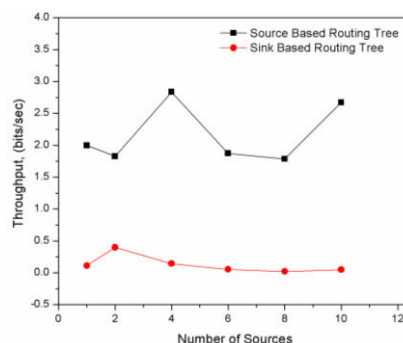


Fig.7 A graph of number of sources versus sink throughput (Tp) of source and sink based routing trees.

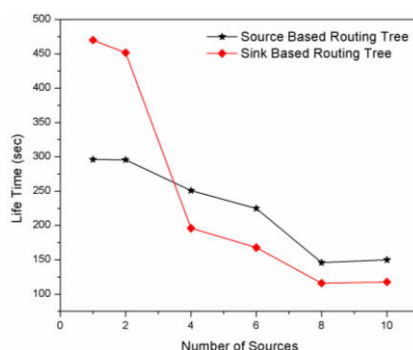


Fig.8 A graph of number of sources versus lifetime of source and sink based routing trees.

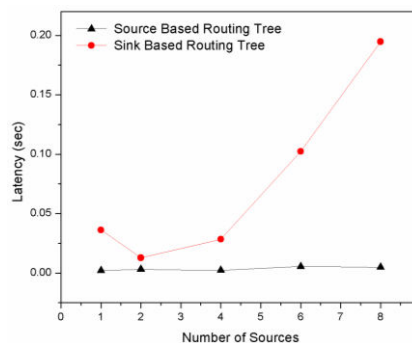


Fig.9 A graph of number of sources versus latency of source and sink based routing trees.

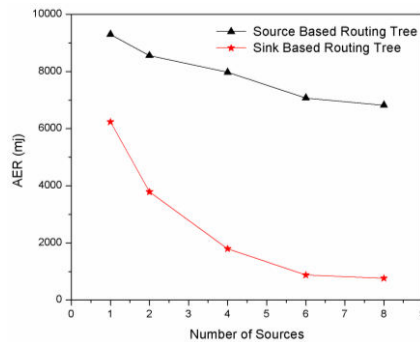


Fig.10 A graph of number of sources versus average energy remaining (AER) of source and sink based trees.

## V. CONCLUSION:

Simulation results shows that Tree-based topology control algorithms can be used in WSNs. Source-based trees is more efficient when the data load is heavy, since they provide more routing paths from each node and also provides good throughput, packet delivery ratio, average energy consumption, latency and lifetime of the nodes compared to sink based routing trees. One of the important point to be noted from this work is that one can carefully tune the source based trees that can provide an efficient topology control solution for specific applications.

## REFERENCES

- [1] J. N. Al-karaki and A. E. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey," *IEEE Wireless Communications*, vol. 11, pp. 6–28, 2004.
- [2] J. Kang, Y. Zhang, and B. Nath, "TARA: Topology-Aware Resource Adaptation to Alleviate Congestion in Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 18, no. 7, pp. 919–931, 2007.
- [3] V. Vassiliou and C. Sergiou, "Performance Study of Node Placement for Congestion Control in Wireless Sensor Networks," in *New Technologies, Mobility and Security (NTMS), 2009 3rd International Conference on*, dec. 2009.
- [4] C. Sergiou and V. Vassiliou, "Energy utilization of HTAP under specific node placements in Wireless Sensor Networks," in *European Wireless (EW), 2010*, pp. 482–87.
- [5] A. Woo and D. E. Culler, "A Transmission Control Scheme for Media Access in Sensor Networks," in *Proceedings of the 7th annual International Conference on Mobile Computing and Networking (MobiCom'01)*. New York, NY, USA: ACM Press, 2001, pp. 221–235.
- [6] Y. Sankarasubramaniam, O. Akan, and I. Akyildiz, "ESRT: Event-to- Sink Reliable Transport in Wireless Sensor Networks," in *MobiHoc '03: Proceedings of the 4th ACM International Symposium on Mobile Ad Hoc Networking & Computing*. New York, NY, USA: ACM, 2003, pp. 177–188.
- [7] C. T. Ee and R. Bajcsy, "Congestion Control and Fairness for Many-to- One Routing in Sensor Networks," in *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*. New York, NY, USA: ACM, 2004, pp. 148–161.
- [8] C.-Y. Wan, A. T. Campbell, and L. Krishnamurthy, "P.S.F.Q: A Reliable Transport Protocol for Wireless Sensor Networks," in *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless Sensor Networks and Applications*. New York, NY, USA: ACM Press, 2002, pp. 1–11.
- [9] C. Sergiou, V. Vassiliou, and A. Pitsillides, "Reliable Data Transmission in Event-Based Sensor Networks During Overload Situation," in *WICON '07: Proceedings of the 3rd International Conference on Wireless Internet, Austin, Texas, October 2007*, pp. 1–8.
- [10] C. Sergiou and V. Vassiliou, "DAIPaS: A Performance Aware Congestion Control Algorithm in Wireless Sensor Networks," in *Telecommunications (ICT), 2011 18th International Conference on*, may 2011, pp. 167–173.
- [11] J. Hightower and G. Borriello, "Location Systems for Ubiquitous Computing," *Computer*, vol. 34, pp. 57–66, August 2001.
- [12] N. Li, J. Hou, and L. Sha, "Design and Analysis of an MST-based Topology Control Algorithm," in *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications*. IEEE Societies, vol. 3, 2003.
- [13] Sergiou, C.; Vassiliou, V. "Tree-forming schemes for overload control in Wireless Sensor Networks", *Ad Hoc Networking Workshop (MedHoc-Net), 2012 The 11th Annual Mediterranean*, On page(s): 61 – 66.