Adaptive Self-Configuration for Distributed Load in Sensor Networks

Conference Item

How to cite:

Copyright and Moral Rights for the articles on this site are retained by the individual authors and/or other copyright owners. For more information on Open Research Online’s data policy on reuse of materials please consult the policies page.
Adaptive Self Configuration for Distributed Load in Sensor Networks

#Mudasser Iqbal, Iqbal Gondal, Laurence S. Dooley
GSIT, Faculty of IT, MONASH University, Churchill, 3842 Australia
[#Mudasser.Iqbal, Iqbal.Gondal, Laurence.Dooley]@infotech.monash.edu.au

Abstract- The dynamic architecture of the network is monitored by exchanging so-called Network State Beacons (NSBs) between key network nodes. The Beacon Exchange rate and the network state define both the time and nature of a proactive action to combat network performance degradation at a time of crisis. It is thus essential to select and update the beacon exchange rate (FE) according to the variations in the load profile of the network. This paper presents a novel localized method that for selecting and updating the FE by adapting to the network load and energy constraints. The results indicating that the model reconfigures the network more effectively to achieve higher throughput as well as greater network integrity, with minimal resource overheads.

Keywords- High Density Ad-hoc Networks, Self Configuration, Distributed Adaptive Optimization, Beacon Exchange Rate

I. INTRODUCTION

Certain time critical and emergency scenarios require very robust, unattended and large scale communication infrastructures to support collaborative operational and signal processing activities. In such situations, ad hoc networks play a critical role in places where a wired backbone is neither available nor economical to build, such as law enforcement operations, battle field communications and disaster recovery. Recent advances in wireless communications and microelectromechanical systems have further extended the capabilities of ad hoc networks, through the development of miniature, low-cost sensors that possess sensing, signal processing and wireless communication capabilities.

To satisfy operational requirements, intermediate nodes called Parent Nodes (PN), which have relatively high resources, are used. These nodes are responsible for such tasks as in-network data processing, communication delay minimization and routing of PS nodes data to the Central Commanding Infrastructure (CCI). As these USN building blocks can fail due to unforeseen local or non-local factors, in order to maintain a minimum QoS for a USN, (which in this context is defined as lossless information delivery at minimal control traffic rates), PNs can be dynamically added or removed from the infrastructure. The non-administrated USNs must be self-monitoring and able to take proactive action to mitigate certain malfunctions before they actually occur.

Proactive network monitoring and reconfiguration requires maintaining the network state across the PNs at optimized instants to mitigate against prospective anomalies. The state profile is maintained by periodic exchanges of NSBs at a set Beacon Exchange Rate (FE). The rate of NSBs constitutes the additional load that the network must support in its reconfiguration activities. The load profile of the network is a key determinant of the network performance and typically defines the course of predictable anomalies in the network, such as the loss of connectivity due to energy shortage. Accurate and timely network information, including the estimated lifespan of key nodes and network load profile enables an effective and proactive strategy to be formulated to alleviate potential network impairments. It is therefore important to attempt to optimize these factors by considering the current network load and maximise throughput while concomitantly minimizing the risk of information loss due to node failures.

Previous work on self-configuring protocols has not focused upon investigating the role the beacon exchange rate plays in maintaining a QoS for the network. Gupta [1] and Chiasserini [2] have focused on energy-efficient, hierarchical modelling of sensor networks through dynamic configuration of the tree nodes. The success of their dynamic tree models is based on a problematic assumption that sensor nodes are able to connect to many PNs simultaneously. Cerpa in [3] emphasized the need for a high degree of synchronization between network components in order to correctly reconfigure. Policy-based and self-managing systems have been also considered, but these impose a high computational and storage requirement on individual sensing units.

This paper presents novel improvements to a proactive self-configuration model [4, 5] by significantly reducing overhead traffic while maintaining a guaranteed QoS. The research establishes bounds for selecting $F_E$ and develops a distributed adaptive model to dynamically update the rate in response to network load profile changes.

The remainder of the paper is organised as follows; Section II explores the underlying USN design and self configuration model, while Section III details the distributed load-adaptive $F_E$ selection model. Simulation results focusing upon the maintenance of QoS and reliability of configuration model are given in Section IV and conclusions are presented in Section V.

II. SENSOR NETWORK DESIGN & SELF CONFIGURATION

A. Network Design

The sensor network design approach described in [6] is based upon the optimal selection of PN density and location in a virtual hexagonal topology structured in autonomous clusters with each cluster headed by a PN. This approach is adopted to
achieve the best QoS by ensuring the availability of PN to a maximum number of PS nodes, while minimizing Grey Region (GR) areas (to reduce many-hop routing) and minimizing confusion/conflict zones.

B. Self Configuration Core Protocol
The network design defines the initial configuration of the sensor network for best QoS with the communication and connectivity model for the PN and PS nodes described in [5]. In the steady state network operation, the model can handle a number of irregularities including: a) increased traffic load leading to congestion and packet losses causing loss of information, b) decreased energy resources increasing the risk of PN failure, c) sudden failure of a PN due to local or non-local disasters and d) addition of new PNs.

To address these various scenarios, a Self-Configuration Protocol is employed [5], with the key element being the continual local geographically monitoring of the network state.

Network State Management: In order to monitor the network for impairments and malfunctions, it is crucial to maintain the state of the network. This state profile can be maintained in both distributed and centralized manners. For this purpose, NSBs are exchanged amongst the PNs throughout the network at the F_S rate. The exchange of NSBs between neighbouring PNs defines the local state of the network at each cluster in terms of network load, remaining energy, remaining life of the PN and the PN availability. The rate of exchange and method of propagation of NSBs are the key factors in defining the nature, time and effectiveness of any proactive action. The following section discusses these two factors in presenting a model that achieves superior performance in terms of more effective energy consumption and reliable data transmission.

III. BEACON PROPAGATION AND EXCHANGE RATE

A. Beacon Propagation
NSBs are exchanged by the neighbouring PNs in the whole network to maintain the state. As stated earlier, network state can be maintained either centrally or in a distributed fashion. For central network monitoring, the NSBs from each cluster head must be propagated to the CCI and the rate of exchange which in turn connects to the CCI for sending NSBs and receiving F_S updates.

Fig. 1 illustrates the centralized propagation scheme. For connected networks, it is possible to declare a PN as a head node to minimize the long range communication with CCI to one PN only to maintain the network state centrally. This head node also works as a gateway of the PN network to external world. The head PN periodically sends aggregated state information to CCI and takes input from the CCI for what F_S should be maintained and which reconfiguration is to be undertaken. This information is routed throughout the network by adopting one of geography-based ad hoc routing strategies. For this purpose, the GEAR [7] protocol is employed which is a recursive data dissemination protocol for wireless sensor networks. GEAR is selected for F_S propagation because of its proven performance in highly dense wireless sensor networks, while consuming minimum energy. Fig. 1B illustrates partially connected and a centralized network state management scheme in which a hybrid interconnectivity approach is employed, with each cluster headed by a PN which in turn connects to the CCI for sending NSBs and receiving F_S updates.

The inherent decentralized structure and high density of USNs reduces the importance of CCI for steady state operation and reconfiguration of the network. Fig. 1C illustrates decentralized clusters, with each cluster headed by a PN and each cluster maintains its local state and own F_S tuned to the requirements of that cluster. Subsection III-B details F_S selection and updating methodology. This distributed state management scheme prevents the need to have inter-cluster communication for NSB propagation, except appointing Associate Parent Nodes (APNs) [5]. These APNs are routing nodes for multi-hop linking between clusters, thereby forming a connected network within a decentralized one. Under steady state situations, these links remain inactive and clusters keep their operations isolated from each others, except mobile source localization and surrogate tracking. This idea of cluster activity optimization in isolation is based on decentralized and self configuring pheromone based communication in ants and termites while they locate food sources or build-up meters high mounds [8]. Simulations in Section IV show the comparative performance of the different propagation techniques discussed above in terms of their impact on network life and data transmission reliability.

Since the cluster heads define the backbone of the network and their life is crucial to the overall life of the network, in this work, the role of cluster head is randomly rotated among all cluster nodes to ensure the network energy resources drain evenly thereby protecting the network from experiencing non-uniform impairments. To assign nodes to the cluster heads in an energy efficient way, the usual minimum transmission power criterion is not employed because of its excessive communication and processing overheads. Instead, the node assignment is optimized to maximize the lifetime of the entire network [2], which is given by:

\[ L = \sum_{i=1}^{N} L_i \]  

(1)

where \( L_i \) is the network life time for a given energy for all clusters; \( S_i \) is the set of cluster heads while \( L_i \) is the lifetime of a single cluster head, defined by:-
where \( E_0 \) is the initial energy available at cluster head, and the two denominator terms respectively represent the power consumption contributions due to the output transmit power and cluster-head transmitting/receiving activity.

B. Beacon Exchange Rate (\( F_E \))

The random beacon exchange rate \([4]\) does not reflect the true state of the network. The selection of the exchange rate based on the load profiles of the clusters provides a better picture. This profile can be established either centrally or in a distributed fashion. The rationale is to keep tuning the exchange rate throughout the network with respect to the level of network activity. If the network undergoes a high load scenario, the energy profiles of PNs will degrade quickly. In this situation, the network state is highly dynamic and beacons must be exchanged more frequently, but at a rate that consumes the least additional energy by optimally adapting to the new load profile of the network and maintaining the actual state of the network across all clusters.

In calculating the rate \( F_E \) two bounds need to be set, the lower being defined by the minimum rate with which the NSBs must be exchanged to maintain the network state even in the case of significantly lower network load. The upper bound limits the maximum value of \( F_E \), exceeding which places extra load on the network due to very frequent NSB exchanges and, actually may result in redundant NSBs being observed and propagated \([4]\). The total load on the network for higher bound of \( F_E \) is:

\[
\eta = U_{\text{Total}} \left[ \frac{T_F}{F_{E_{\text{max}}}} \right] \mathcal{U} \text{ where } U_{\text{Total}} = \int_{t=0}^{T_F} \sum_{i=0}^{C_i} U_i \tag{3}
\]

where \( U_{\text{Total}} \) is the total load on the network within a given time interval \( \{T_i, T_f\} \); \( U_i \) is the load on PN \( i \) at time \( t \), \( \mathcal{U} \) is the extra load caused by one proactivity, \( F_E \) is number of clusters and \( \eta \) is the total load on the network, including the load caused by proactivity, at time \( T_F \). The second term in (3) is the load caused in this interval by proactive activities. Given the extra load \( (U_i) \) policy factor \( k, \eta \) defines the upper bound of \( F_E \) satisfying:-

\[
\eta \leq \left(1 + \frac{k}{100}\right) U_{\text{Total}} \tag{4}
\]

i.e. \( F_{E_{\text{max}}} \) must maintain \( \eta \) within the allowed extra \( k\% \) load.

The relationship between the lower bound \( (F_{E_{\text{min}}}) \) and minimum required update resolution \( (T_R) \) is given by:

\[
F_{E_{\text{min}}} \leq T_R \tag{5}
\]

However, if \( F_{E_{\text{min}}} \ll T_R \) then redundant NSBs may be propagated, resulting in significant overhead proactivity actions. Conversely, if \( F_{E_{\text{min}}} \) is greater than \( T_R \), the NB propagation will be less frequent than required so there is a probability that at times the network will be under-stated, a condition where the actual picture of current network state is not available. To avoid these two extremes of redundancy and under-stateness, \( F_{E_{\text{min}}} \) needs to be optimized. Consider the following relationship:

\[
d = T_R - F_{E_{\text{min}}} \tag{6}
\]

The optimal lower bound of \( F_E \) must be as close to \( T_R \) as possible so it minimizes the lower bound optimization factor \( d \). This operational zone describes the optimal range for the selection of lower bound that would keep network state safely normal thereby avoiding the two extreme conditions. The relationship between the network state \( (\delta) \) and lower bound optimization factor \( (d) \) is given by:

\[
\delta = d^3 (p)^{-1} \tag{7}
\]

where \( p \) is a tuning factor, whose value depends upon the resolution of updating \( (T_R) \). The operational zone is defined by:

\[
-2d \leq \delta \leq 2d \tag{8}
\]

Fixing the upper and lower bounds of \( F_E \) is greatly influenced by two design parameters, namely the extra allowable network load \( U_i \) and the network state update resolution \( T_R \), which are conditionally dependant on each other. This dependency states that for a particular \( U_i \) there is a minimum \( T_R \), and vice versa, beyond which the update resolution starts placing an additional load on the network that is not permitted. To numerically define this relationship, the extra load \( W \) introduced by \( F_{E_{\text{min}}} \) is given by:

\[
W = \left[ \frac{T_F}{F_{E_{\text{min}}}} \right] U_{\text{Total}} \tag{9}
\]

In order to conform to the design-policy:

\[
W \leq 0.1k \times U_{\text{Total}} \tag{10}
\]

\[
\Rightarrow k \geq 0.1W \times (U_{\text{Total}})^{-1} \Rightarrow k_{\text{min}} = 0.1W \times (U_{\text{Total}})^{-1} \tag{11}
\]

Equation (11) defines the minimum value of \( k \) that can be used while allocating the extra load for a particular update resolution \( T_R \). Conversely the maximum value of \( k \) is not linearly dependent on the update resolution, but rather it is defined by the required lifetime of the network. From (2), the new lifetime of a single cluster head, after incorporating proactive activities is given by:

\[
L_{\text{new}} = E_i (\alpha c_i + f(n_i) + p(W))^{-1} \tag{12}
\]

where the term \( p(W) \) represents the contribution to power consumption due to extra load introduced by cluster head. \( \lambda \) for proactive activities. From (10), if:

\[
W = 0.1k \times U_{\text{Total}} \tag{13}
\]

then \( k_{\text{max}} \) should be selected so that \( L_{\text{new}} \geq L_{\text{req}} \) with \( L_{\text{req}} \) as the required life of the network chosen by the network designer.

Having defined the exchange rate bounds, \( F_E \) is initially selected \((F_{E_{\text{init}}} \) for \( t = 0 \) to be equal to the lower bound \((F_{E_{\text{min}}}) \) and is updated dynamically according to the changes in the load profile of the network. This rate is then periodically updated to \( F_{E_{\text{init}+t}} \) using the following linear stochastic feedforward process:

\[
F_{E_{\text{init}+t}} = F_{E_{\text{init}}}(1 + 0.1\lambda) \tag{14}
\]

where \( \lambda \) is the parameter used to update the current exchange rate depending upon the change in load profile of the network, which is given by:
Simulations were carried out to evaluate the performance of the network with $F_E$ selection and communication model implemented in both centralized ($CF_E$) and distributed ($DF_E$) ways for different load profiles and PN malfunctions. Table I details the complete simulation environment parameters. Network Throughput, Overhead Control Traffic and Synchronization Maps were used to analyse the performance of the system under various environmental parameters. These metrics are defined in the following paragraphs:

- **Throughput**— It is defined as:
  \[
  \text{Throughput} = \frac{\text{Energy Consumed}}{\text{Packet Loss} + c}
  \]
  where $c=0.01$ is a constant chosen to avoid division by zero. This metric is calculated in order to quantify the benefit of proactiveness at the cost of extra energy consumption.

- **Control Overhead**— The number of control packets generated over the life of the network. Since the amount of control traffic, generated by our model, is defined by the $\lambda$ curves, this metric compares the performance of employing different curves for different load profiles in terms of minimization of overhead traffic.

- **Synchronization Map**— A two dimensional graphical interpretation of instants (and $F_E$ maintained) when the NSBs are exchanged to synchronize the network state information across the PNs in the backbone. The map helps to analyse graphically the cause of lesser overhead traffic generated by a particular $\lambda$ curve. This helps in the selection of a particular $F_E$ update method (value of $v$) for a specific load profile.

The following subsections analyse each of these performance metrics in detail.

### A. Network Throughput

Fig. 3 shows the aggregate energy drainage profile of the PNs, which reveals a minimum increase of up to 10-15% in the energy consumption for Hybrid technique over the no-$F_E$ technique. Fig. 4 illustrates that proactiveness improved overall throughput for all types of beacon exchanges, with the Hybrid method maintaining the best QoS for an additional 25% of the network operation time compared to the no-$F_E$ scenario.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area under Surveillance</td>
<td>Open irregular terrains of dimensions 25000m²</td>
</tr>
<tr>
<td>Deployment Topology</td>
<td>Random for both PS &amp; PN nodes</td>
</tr>
<tr>
<td>CH Comm. Range (m)</td>
<td>50-450</td>
</tr>
<tr>
<td>Density of PS nodes</td>
<td>200-500</td>
</tr>
<tr>
<td>Density of PN nodes</td>
<td>30-50</td>
</tr>
<tr>
<td>$F_E$ Implementation</td>
<td>No, Centralized, Distributed</td>
</tr>
<tr>
<td>Sampling Rate</td>
<td>5 Sec</td>
</tr>
<tr>
<td>Control Packet Size</td>
<td>2 kbits</td>
</tr>
<tr>
<td>Network Activity Time (mins)</td>
<td>80 Sequential experiments, 900-1800 seconds time for each run</td>
</tr>
<tr>
<td>Power Consumption (mW)</td>
<td>Tx 14.88, Rx 12.50, Idle 12.36, Sleep 0.016</td>
</tr>
</tbody>
</table>

### C. Implementation Method

$F_E$ selection and tuning model defined by the boundary equations (3)–(6) and update equations (14) and (15) is implemented in both centralized (Fig. 1A, 1B) and distributed (Fig. 1C) fashion. In case of centralized control, the CCI governs the calculation of $F_E$ which is then communicated to the cluster heads for implementation. Also the cluster heads send the updates on network state to CCI and, in response, receive necessary reconfiguration instructions. The load parameter in the equations would refer to the total load on the network while the life would be defined as the sum of the lives of all cluster heads. When the model is decentralized, the clusters behave as autonomous network regions and calculation of $F_E$ is devolved at cluster head level. Also the state of the network is managed locally and communicated to other clusters only in the case of anomalies through routing nodes. This methodology makes the structure self-sufficient in its operation not depending upon external communication infrastructures (such as CCI) for $F_E$ tuning and reconfiguration. Simulation results in the following section further quantify these arguments and provide a performance comparison of the model.
value leads to less energy consumption but also less confidence in maintaining QoS for a longer time period, so mandating proper selection of $F_E$ within the bounds. Also the unstable throughput in the case of large $F_E$ in Fig. 4B shows the understated condition where the value of $F_E$ fails to keep the network state updated so proactive action of the model is unable to counter possible impairments in time.

**B. Control Overhead**

To quantify the impact of the $\lambda$ curves on the overhead control traffic required for maintaining the network state across all PNs, the network was tested under various load profiles, with Fig. 5A showing increasing, normal and random load profiles applied on the network. From earlier theory, the logistic change in $F_E$ was developed to support only network load changes by a factor of two. It was anticipated that for random load changes, the straight line ($v = 1$) would perform better, and this is borne out in Fig. 5B where by inducing an equivalent change in $F_E$ as the load changes keeps the network well informed about the network state, with lower control traffic than that of the logistic change. Conversely, when the network load underwent smooth changes from start to finish, it was found that inducing logistic change helped conserve bandwidth by minimizing control traffic as observed in the increasing and normal load profiles in Fig. 5B.

The reason for the improved performance of the logistic curve is found to be that for increasing and normal load profiles, the logistic curve updates exchange rate at a higher value (i.e. lower beacon exchange rate) earlier than the straight line. This earlier updating reduces traffic and conserves overall network bandwidth. Similar trends are observed in the case of random loads where the straight line performed better. It was found that the straight line established higher $F_E$ values quicker than the logistic curve which also reduced network traffic more quickly and for a longer period, making it suitable for random load situations.

**C. Synchronization Maps**

To investigate the superior performance in different load situations of one $\lambda$ curve over the other, an analysis of the exchange rates that are maintained by the two curves throughout the operation of network was performed. Fig. 6 shows the synchronization maps for the curves $v = 1$ and $v = 3$, for the load profiles in Fig. 6A. Each map shows the actual $F_E$ maintained on the horizontal axis, while the vertical bars show the time instant when these rates are synchronized between all PNs. Examining the maps for increasing and normal load profiles, it is observed that the logistic curve updates the exchange rate to a higher value, which means both earlier and less frequent exchanges of beacons, than the straight line. This earlier update reduces traffic and conserves overall bandwidth. A similar occurrence was observed in the case of random load where the straight line performed better.

---

**Fig. 3:** Energy consumption profile for implementation methods of $F_E$

**Fig. 4:** Comparison of throughput for various implementation methods of $F_E$ for (A): 10 Sec $F_E$, (B): 40 Sec $F_E$

**Fig. 5 (A):** Increasing, Normal and Random Loads applied on the network to test the comparative performance of lambda curves, (B): Amount of control traffic generated by various load profiles for different $\lambda$ curves

**Fig. 6:** Synchronization Maps for the two $\lambda$ curves when applied on increasing, normal and random load profiles. The numbers on the horizontal axis are the actual $F_E$ (in seconds) maintained by the two curves for an active network.
(Fig. 6B), while the map in Fig. 6C shows that the straight line establishes higher $F_E$ values earlier than the logistic curve, so reducing network traffic sooner and for a longer period of time making it suitable for random load situations. Another observation relates to the increased control traffic in the case of normal load. The synchronization maps of Fig. 6B reveal that this is due to the curves maintaining the exchange rate closer to the initial rate posing a requirement of tuning the initial $F_E$ within the lower and upper bounds defined previously.

V. CONCLUSIONS

This paper has presented a new beacon exchange rate selection and tuning technique for centralized and distributed load based methods. Both analytical and simulation results have shown that adapting the $F_E$ according to the load variations provided a significant performance improvement in terms of network throughput, that helped in proactively handling network malfunctions, including node failure and overload. The results also confirmed the model’s stability in terms of inducing logistic changes in $F_E$ for a normal network load profile which adapts to load changes in such a way that network synchronization requests are minimized. The proposed model was found to be very robust with more than 70% of component devices observed connected through development of multi-hop routes in a sensor network.

REFERENCES