THE INSTITUTE FOR SYSTEMS RESEARCH

ISR TECHNICAL REPORT 2009-9

Distributed Pruning Methods for Stable Topology Information Dissemination in Ad Hoc Networks

Kiran Somasundaram

The Institute for Systems Research

A. JAMES CLARK SCHOOL OF ENGINEERING

ISR develops, applies and teaches advanced methodologies of design and analysis to solve complex, hierarchical, heterogeneous and dynamic problems of engineering technology and systems for industry and government.

ISR is a permanent institute of the University of Maryland, within the A. James Clark School of Engineering. It is a graduated National Science Foundation Engineering Research Center.

www.isr.umd.edu

Distributed Pruning Methods for Stable Topology Information Dissemination in Ad Hoc Networks

Kiran K. Somasundaram, John S. Baras, Vahid Tabatabaee and Kaustubh Jain
Department of Electrical and Computer Engineering
University of Maryland,
College Park, MD 20770
Email: kirans@umd.edu, baras@umd.edu, vahidt@umd.edu, ksjain@umd.edu

***Abstract*—We propose a component based modelling and design strategy for the Neighborhood Discovery Component (NDC) and the Selector of Topology Information to Disseminate Component (STIDC) of proactive ad hoc routing protocols. The NDC design strategy focuses on limiting the detection time and removing time of network good and bad links respectively. The STIDC strategy makes sure that only stable links with long life time are selected as the topology information that will be broadcasted in network. We provide analytic performance analysis methods for both components and show that using the proposed methodologies we can significantly reduce the overhead of information dissemination in proactive routing protocols such as OLSR.**

I. Introduction

One of the key challenges and obstacles in deployment of proactive routing protocols such as OLSR [3] for Mobile Ad-hoc Networks (MANETs) is the overhead of topology information dissemination. In proactive routing protocols every node should have enough information to select its path or at least the next-hop to the desired destination. There are two steps in controlling the information dissemination overhead. The first step is to select an efficient mechanism for dissemination of information [2], and the second step is to use pruning methods for selection of topology information that will be disseminated. In this paper, we focus on the second step and propose a pruning methodology.

In [6], [7], a component based framework for modelling and design of routing protocols for ad hoc networks is proposed. This method provides a systematic approach that can be used in the study, analysis, design and optimization of routing protocols. In this approach, the routing protocol is modeled as a complex distributed system of systems. The complex system is divided into separate components that are divided into subcomponents themselves. The main objective of a component based design is *separation of concerns*, which is achieved by breaking the system into components that overlap as little as possible in functionality. The design principles, interfaces and performance metrics for each component should be specified so that each can be studied and designed separately.

The main components of a MANET routing protocol are: Neighborhood Discovery (NDC), Selector of Topology Information to Disseminate (STIDC), Topology Dissemination (TDC), and Route or Path Selection (RSC), Packet Forwarding (PFC). The two components that determine the volume and frequency of topology information that should be disseminated are NDC and STIDC. NDC determines the local mechanism that is used by each node for detection of neighbors in a dynamic network, and STIDC determines the mechanism which will be used for selection of local information to be broadcast to the network. In this paper we propose performance metrics and design strategies for these two components. We use the OLSR [1] routing protocol and illustrate our design methodology through extensions and generalizations of this routing protocol, as well as other routing protocols.

In mobile ad hoc networks (MANET), delay in detecting links and their failures is critical; hence we use these metrics as the main design guidelines for the NDC. The NDC design objectives are: (1) to detect "good" quality links with low delay, and (2) when quality of a detected link degrades, and is not acceptable, it should be removed from the detected link list with low delay too. Using these two main criteria, some of the detected links will have a short life time. Clearly these links are not good candidates to be broadcast in the network; the STIDC design mechanism is to select the most stable links for the network topology representation.

In sections II and III, we describe the function, performance model, design metrics and strategy for the NDC and STIDC respectively. In section IV we present and discuss simulation results.

II. Neighborhood Discovery Component (NDC)

Proactive routing protocols, such as *Optimized Link State Routing* (OLSR) [3], employ neighbor discovery methods to identify their local neighborhoods. Once this local neighborhood information is observed at every node, these link state routing protocols, broadcast a pruned version of the information across the network. A detailed discussion of these pruning methods is presented in [5] and [2]. The pruning methods are abstracted as the *Selector of Topology Information to Disseminate Component* (STIDC), which we will discuss in the next section. We refer the reader to [7] for a detailed exposition of models for these components.

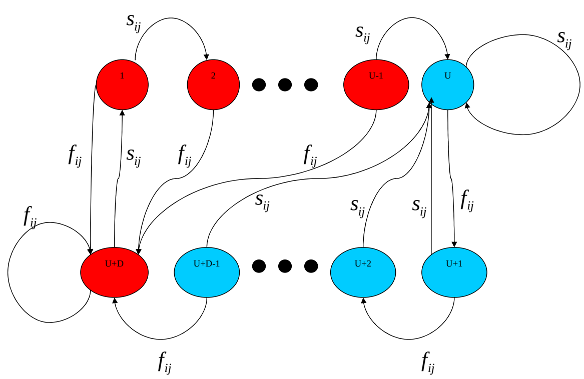

Fig. 1. FSM of neighbor detection mechanism

The neighbor discovery methods in ad hoc networks are usually driven by proactive HELLO packet broadcasts [3], [8]. The HELLO messages become susceptible to channel and contention losses. In order to combat these effects most of these protocols have been built with hysteresis to dampen the dynamics of the discovery (section 14 of [3]).

A. Performance Model:

A generalized state space model for these neighbor discovery methods has been introduced in [7]. When a station i receives a HELLO message from station j, that contains its own ID, consecutively for a neighbor Discovery Time (NDT), it will add station j to its neighbor list. Station i removes j from its neighbor list if it does not receive any HELLO message from j for the NHT period. These two periods form the hysteresis parameters of the NDC.

The Finite State Machine (FSM) for the neighbor discovery algorithm described is shown in Figure 1. Each station i in the network runs the FSM for every station j in its radio range. Whenever this station receives a HELLO packet from j it corresponds to a decision edge s, and when the HELLO packet is lost in transmission it corresponds to a decision edge f. Station i declares a unidirectional link $i \longrightarrow j$ if it is in any one of states from U to $U + D - 1$. Otherwise it declares a unidirectional link failure. This FSM then forms the *executable model* for the NDC in the system design.

The corresponding *performance model*, which captures the steady state behaviour of the FSM, can be obtained using Markov chain analysis. The inputs to this model are the probability of success (s_{ij}) and failure($f_{ij} = 1 - s_{ij}$) in the transmission of HELLO messages. The design parameters are U and D. A detailed analysis of these methods is presented in [7], [10].

B. Performance Metrics

Let π_k be the steady state probability that the NDC is in state k of the Markov chain. We can use the generalized global balance equations to derive the steady state probabilities. One of the main NDC performance metrics is the probability of detecting a directional link to node j from node i is:

$$q_{ij} = \sum_{k=U}^{U+D-1} \pi_k \tag{1}$$

and if we assume that the probability of successful transmission from i to j and from j to i are independent from each other, then the probability of a bidirectional link detection is:

$$p_{ij} = q_{ij}^2 \tag{2}$$

The design (or control) parameters for NDC are the U and D parameters that can be set to achieve the desired performance. We can consider a number of performance metrics for the NDC. For dynamic and mobile networks, delay in the detection of a neighbor and delay in removing a node from the neighbor list are important metrics. We can use Markov chain analysis techniques to approximate and/or compute lower and upper bounds for these parameters [10]. For the unidirectional links the link detection delay is the first hitting time of state U if we start from state $U + D$. Likewise, delay in removing a node from the neighbor list is the first hitting time of state $U+D$ starting from state U. These distributions are for unidirectional links; for bidirectional links the detection delay is the maximum of the detection times of two unidirectional links. Similarly, the removing delay for bidirectional links is the minimum removing delay of the two corresponding unidirectional links.

Another important parameter (metric) that determines the overhead of update messages in the network is the rate of status change for bidirectional links in the network. We denote a bidirectional link by BU (bidirectional link up) and a unidirectional or lost link by BD (bidirectional link down). The probability of link status of link (i, j) changing from BU to BD is approximated by

$$p_{BU\to BD} = \frac{2 \cdot \pi_{U+D-1} f \cdot q - \pi_{U+D-1}^2 f^2}{p} \tag{3}$$

Here we have dropped the indices i,j for simplicity. Again, we have assumed the independence and identical probabilities of successful transmission over the directional links $i \longrightarrow j$ and $j \longrightarrow i$. The idea behind the approximation is that the link changes from BU to BD when at least one of the directional links makes a transition from the state $U+D-1$ to $U+D$ in the FSM, while the other one is in one of the link detecting states (blue states in Figure 1). Similarly, the probability of the link status changing from BD to BU is

$$p_{BD\to BU} = \frac{2 \cdot \pi_{U-1} s \cdot q + 2 \cdot \pi_{U-1}^2 s^2 - \pi_{U-1} s \cdot \pi_{U+D-1} f}{1 - p} \tag{4}$$

Using conditioning arguments, the *bidirectional link change rate* is given by

$$\lambda = (p_{BU\to BD} \cdot p + p_{BD\to BU} \cdot (1 - p))\, \lambda_H \tag{5}$$

where as before, p is the probability of detecting a BU link and λ_H is the rate of sending HELLO messages.

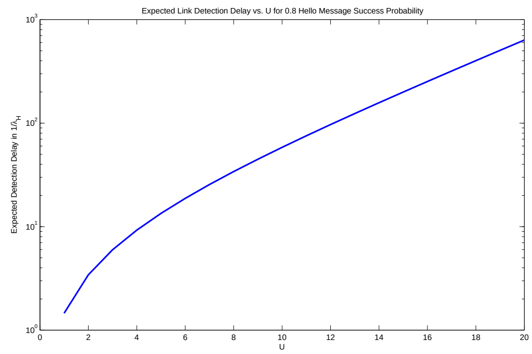

Fig. 2. Upper bound for Expected Detection Delay of Good Links

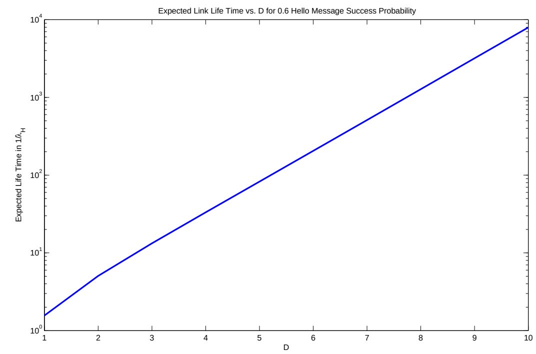

Fig. 3. Lower bound for Expected Life Time of Bad Links

Thus, λ_{ij} is the rate with which node i will detect changes in its bidirectional link to node j. This in turn will determine the rate with which the node i will generate Topology Change (TC)messages. An unstable link will have a lot of transitions between BU and BD which will increase the TC message payload. Hence, λ_{ij} can be used as a performance metric.

C. Design Strategy

In [7] a design mechanism to choose appropriate U and D to discover links with desired detection probability is given. While this does reduce the number of unstable links detected, it suffers from long detection delays that is not acceptable for dynamic networks. This is evident from the sojourn times analysis of the Markov chain model introduced in [10]. The analysis results for the sojourn time of the link detection is shown in Figure 2.

It is clear from figure 2 that the detection time grows exponentially with the parameter U. Thus the process of detecting only stable bidirectional links is time consuming and hence does not satisfy the primary objective of quick neighbor detection. We thus shift the onus of dealing with the problem of unstable links to the STIDC. We argue and show that this is a natural problem in this component based abstraction. In dynamic networks the primary objectives of NDC should be to limit the detection time of good links and at the same time the removing time of bad links.

Let us assume that for a good link the success probability is above 0.8, and for a bad link it is below 0.6. In figures 2, 3 the average link detection and life time delay for good ($s = 0.8$) and bad links ($s = 0.6$) as a function of U and D respectively are given. From these figures it is clear that by choosing $U = 2$ and $D = 2$ we can keep both metrics below 10 seconds.

III. Selector of Topology Information to Disseminate Component (STIDC)

Many routing strategies inspired from protocols used in wired networks were shown to perform poorly when applied to networks with mobile stations [9]. Thus there is a need to develop protocols to take the dynamic nature of topology into account. One such effort was OLSR [3]. OLSR is a proactive link state algorithm in which every station in the network floods pruned local link state information across the network. This is needed because, unlike in wired networks, the topology here is dynamic and flooding the entire link state information can create *broadcast storms* [2]. Thus link state algorithms must include appropriate pruning methods to broadcast only significant links. This functionality in link state routing algorithms in our framework [6], is abstracted as the *STIDC*. This component based abstraction is discussed in detail in [6], [7]. In this paper we present novel methods to develop the functional and performance models of the STIDC.

We observe that a good STIDC should have the following properties:

1) The amount of link information that is flooded across the network should be sufficiently low.
2) The reduced topology as viewed by every host in the system, must preserve the shortest paths for routing.

Though OLSR pruning guarantees the second condition, it fails to reduce the link information broadcast storm in a dynamic network. This is because the *MPR* selection algorithm [9], employed in OLSR, tries to prune local information on a static topology without considering its time-varying nature. The greedy MPR selection algorithm (based on second-order neighbor coverage) is likely to choose unstable links. This is because the nodes which are farthest from the host are likely to be chosen as MPRs because they tend to have a larger coverage. These MPRs usually have poor signal reception and hence the detected bidirectional links are more likely to be unstable.

We show that this problem can be resolved if we choose a metric that considers the dynamics of these links. A good metric to consider is the *bidirectional link rate* introduced in subsection II-B. This leads to a formulation which helps us to interpret the STIDC as a solution to a constrained optimization problem.

A. Functional Model

As with traditional pruning algorithms we assume that every host h in the network knows its local neighborhood (the first-order (one-hop) neighbors) $\mathcal{N}^1(h)$ and the second-order(two-hop) neighbors $\mathcal{N}^2(h)$. With this local view every host attempts to solve the following optimization problem to obtain the set of neighbor links that it should flood across the network:

$$\min \sum_{j \in \mathcal{N}^1(h)} \lambda_{hj} \quad \cdots Problem(h)$$

such that shortest paths from h to every $i \in \mathcal{N}^2(h)$ are preserved.

In effect this problem formulation attempts to find the optimal set of stable links which preserve the shortest paths in the global view of the network as a dynamic graph. This property is shown in subsection III-C. As a result any solution to $Problem(h)$, viewed globally, indeed satisfies the general properties of a STIDC described above. Unfortunately $Problem(h)$ turns out be an NP-Hard problem as shown in the next subsection.

B. Computational Complexity and Greedy Approximation

In this subsection we show that $Problem(h)$ is an NP-Hard problem in the local neighborhood. We present a parsimonious transformation of the well known NP-Complete *set cover* problem to $Problem(h)$. Let $\mathcal{U}$ represent the universe of elements. Let $\mathcal{S} = \{S_1, S_2, \cdots, S_N\}$ be the set of subsets of the universe. Associated with each subset is a weight $w_k, 1 \leq k \leq N$. The optimal set cover is to find a subset of $\mathcal{S}$, $\mathcal{C}$ which covers all the elements in the universe and also minimizes the cost $\sum_{S_k \in \mathcal{C}} w_k$. Consider the following polynomial time transformation. Let us create a fictitious host h. For each element s_i in $\mathcal{U}$ create a node i which represents the second-order neighborhood $\mathcal{N}^2(h)$. For each of the sets in $S_k \in \mathcal{S}$ create a node k which represents the first-order neighborhood $\mathcal{N}^1(h)$. For each $k \in \mathcal{N}^1(h)$, let the *link change rate* be $\lambda_{hk} = w_k$. The incidence between $\mathcal{N}^1(h)$ and $\mathcal{N}^2(h)$ is then created as follows. There exists an edge between $k \in \mathcal{N}^1(h)$ and $i \in \mathcal{N}^2(h)$, if the set S_k contains the element s_i. A feasible solution to $Problem(h)$, in this local view, would essentially solve the optimal set cover problem. The constraint of maintaining all the shortest paths to $\forall i \in \mathcal{N}^1(h)$, ensures that all the elements in $\mathcal{U}$ are covered by the one-hop neighbors which form the cover $\mathcal{C}$. This would be the optimal cover because it minimizes $\sum_{k \in \mathcal{N}^1(h)} \lambda_{hk}$, which is the same as minimizing the covering cost $\min_{S_k \in \mathcal{C}} \sum_{S_k \in \mathcal{C}} w_k$.

This parallel with the set-cover problem suggests to use the popular greedy approximation algorithm employed for the set cover problem [4]. Algorithm 1 is carried out by every host h in the network. The algorithm outputs the set of stable neighbors $\mathcal{C}$ which solve $Problem(h)$. Each host uses a set of relays (for example, those obtained from the Topology Dissemination Component [7]) to broadcast any topology changes of the stable link set $\{(h,j), \forall j \in \mathcal{C}\}$.

Algorithm 1 Greedy Approximation algorithm for $Problem(h)$

Cover $\mathcal{C} = \emptyset$
Removing the essential cover
for all $i \in \mathcal{N}^2(h)$ **do**
  $E_i = \{j_1, j_2, \cdots, j_d\} \leftarrow$ set of vertices $\in \mathcal{N}^1(h)$ which shares an incidence with i.
  if $|E_i| == 1$ **then**
    $\mathcal{C} = \mathcal{C} \cup \{j1\}$
  end if
end for
$\mathcal{R} = \mathcal{N}^2(h)$
for all $j \in \mathcal{C}$ **do**
  for all $i \in \mathcal{N}^2(h)$ **do**
    if $(i,j) \in E$ **then**
      $\mathcal{R} \leftarrow \mathcal{R} \backslash \{i\}$
    end if
  end for
end for
$N^1(h) \leftarrow \mathcal{N}^1(h) \backslash \mathcal{C}$
Greedy selection
while $\mathcal{R} \neq \emptyset$ **do**
  $D_j = \{i1, i2, \cdots\}$ be the set of vertices $\in \mathcal{R}$ which shares an incidence with $j \in N^1(h)$.
  $c_j = \frac{\lambda_{hj}}{|D_j|}$
  Assign c_j to each element $i \in D_j$. This is the cost in covering i.
  $j^* = \arg\min_{j \in N^1(h)} c_j$
  $\mathcal{R} \leftarrow \mathcal{R} \backslash S_{j^*}$
  $N^1(h) \leftarrow N^1(h) \backslash j^*$
  $\mathcal{C} \leftarrow \mathcal{C} \cup \{j^*\}$
end while

For each $j \in \mathcal{N}^1(h)$, the influence of j in the second order neighborhood $\mathcal{N}^2(h)$ is defined as the set of neighbors of j in $\mathcal{N}^2(h)$. Mathematically the *influence* is given by $I(j) = \{i \in \mathcal{N}^2(h) : (j,i) \in E\}$, where E is the edge set in the local view of the host h. Let us denote the size of the maximal influence $|I^*| = \max_{j \in \mathcal{N}^1(h)} |I(j)|$. We denote by Λ^{Greedy}, the cost achieved by using the greedy algorithm and Λ^{Opt}, the optimal cost.

Theorem 3.1:

$$\frac{\Lambda^{Greedy}}{\Lambda^{Opt}} \leq H(|I^*|)$$

where $H(n) = \sum_{i=1}^{n} \frac{1}{i}$ is the n^{th} harmonic number.

Proof: The proof is based on lemma 11.9 (Chapter 11 of [4]) which states that the total covering cost of any influence

$$\sum_{i \in I(j)} c_i \leq H(|I(j)|)\lambda_{hj}$$

Let us suppose that the optimal stable neighbor link set is $\mathcal{C}^*$. Then

$$\begin{aligned} \Lambda^{Optimal} &= \sum_{j \in \mathcal{C}^*} \lambda_{hj} \\ &\geq \frac{1}{H(|I^*|)} \sum_{j \in \mathcal{C}^*} \sum_{i \in I(j)} c_i \\ &\geq \frac{1}{H(|I^*|)} \sum_{i \in \mathcal{N}^2(h)} c_i \\ &= \frac{1}{H(|I^*|)} \sum_{i \in \mathcal{C}} \lambda_{hj} \end{aligned}$$

■

C. Global properties of the algorithm

Theorem 3.2: The shortest path between any pair of stations i and j is preserved in every host h's global view.

Proof: Let us suppose the shortest path hop count between two stations S and D is k. For $k \leq 2$ the proof is trivial (the local view gives all the paths). For $k > 2$, let us suppose that the shortest path is $S \rightarrow j_1 \rightarrow j_2 \cdots \rightarrow j_k \rightarrow D$ in the original communication graph G. Since the pruning algorithm preserves all shortest paths in the local view, there would be at least a replacement path $S \rightarrow j_{r1} \rightarrow j_2$. Thus $j_{r1} \rightarrow j_2 \rightarrow j_3 \cdots \rightarrow j_k \rightarrow D$ is a path of length $k-1$. If we apply the above argument recursively it is trivial to show that that shortest path is indeed preserved in the global view. ■

D. Performance model

Performance models for components of an algorithm help analyze its performance without resorting to packet level simulations. The performance models should generalize and capture the properties of all such component system realizations. Furthermore, the performance metrics of the algorithm should be derivable from the performance metrics of the components. A good performance model and its analysis for NDC is carried out in [7], [10]. In this paper we develop a performance model for STIDC based on reduced Monte-Carlo methods and this necessitates the addition of a new output parameter λ_{hj} for the NDC. These component models help in parametric analysis of the composition of the NDC and the STIDC. The Markov chain methods introduced in subsection II-B for NDC give good estimates of the steady state bidirectional link statistics. These performance outputs from the NDC serve as the inputs to the performance model of the STIDC. Figure 4 illustrates the relation between the two performance models.

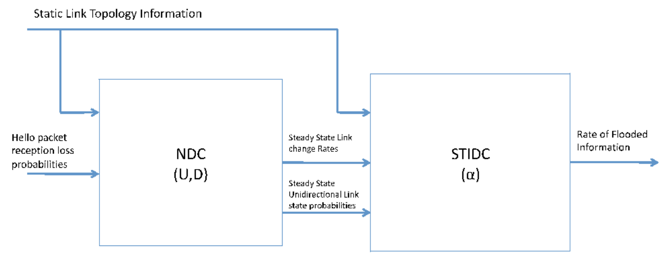

Fig. 4. Composition of Performance Models

While NDC performance models provide local metrics of the graph topology, the set cover algorithms that couple the state spaces of the FSMs of every station provide a global metric of the graph topology. To perform an exhaustive Monte-Carlo simulation to estimate the link-flooding overhead would be an overkill. Instead we extract a reduced Markov model (of the coupled Markov models of the NDCs) shown in Figure 5 which accurately captures the average link change rate (however, the sample paths are *not* identical). This reduced Markov chain can be used to drive a Monte-Carlo simulation to approximate the amount of link information to broadcast.

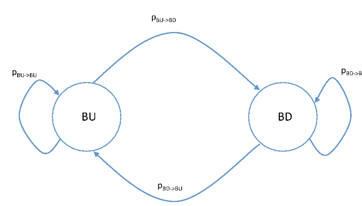

Fig. 5. Reduced Markov Chain which captures the average change rate

In order to obtain a generalized performance model for the STIDC, we claim that the STIDC algorithm attempts to solve the following optimization problem

$$\min \sum_{j \in \mathcal{N}^1(h)} \lambda_{hj}^{\alpha}$$

such that shortest paths from h to $\mathcal{N}^2(h)$ are preserved.

When $\alpha = 0$ it captures the OLSR's MPR selection algorithm and when $\alpha = 1$ it reduces to our rate sensitive set-cover problem. Thus (U, D) and α form the tuning parameters for the components NDC and STIDC respectively.

IV. Results and Discussion

To illustrate the power and efficiency of performance analysis using component methods we choose the network example shown in Figure 6. The values on the edges indicate the symmetric loss probabilities for the HELLO messages. These Bernoulli processes induce a random process on the graph. The neighbor discovery protocol tries to detect bidirectional links based on this process which evolves over the edges. The neighbor discovery protocol depending on the programmed hysteresis parameters (U, D) dampens the bidirectional link discovery. The neighbor discovery process can be interpreted as a filtering process of the original stochastic process. The STIDC algorithms run on this filtered process and choose a set of *significant links* that needs to be flooded. The manner in which nodes perceive the significance of links depends on the tuning parameter α. The tuning factor $\alpha = 0$ myopically chooses links to nodes that have maximal local coverage. However $\alpha = 1$ chooses cheap nodes (whose links do not change often) that also give a good coverage.

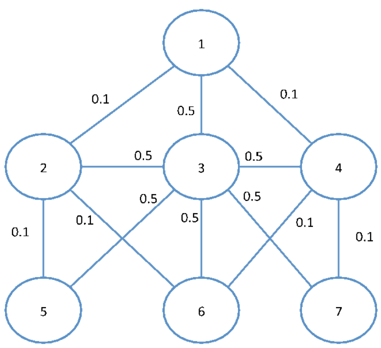

Fig. 6. Example network

For $\alpha = 0$, which corresponds to the default OLSR MPR selection algorithm we see a lot of link changes that are flooded. However when we use the new selection criteria based on our greedy algorithm, we observe a significant reduction in the flooding information. In Figure 7 we have obtained the average number of links flooded for varying U and D. We compare the performance for two values of α (0 and 1).

In Table IV we obtain the average number of links flooded for α varying in [0,1], in steps of 0.2, while $U = 2$ and $D = 2$ are fixed. The rates shown are normalized with respect to the HELLO transmission rates. As expected by changing α we observe that we can control the average number of links broadcast. Thus, the parameters U, D and α can be tuned to operate at various tradeoff points.

V. Conclusions

In this paper we have developed the performance model for the STIDC and presented a design strategy to optimize the amount of topology information broadcast across the network.

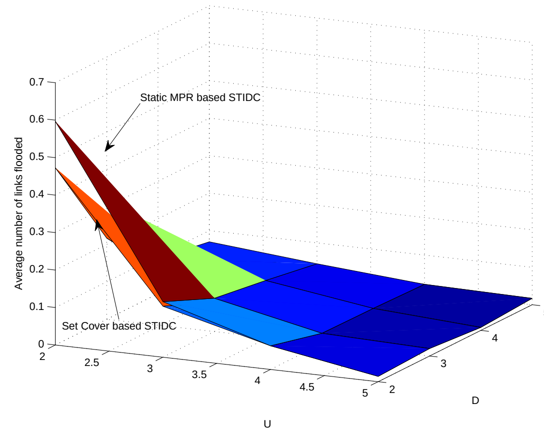

Fig. 7. Flooding performance of STIDC for varying U and D

α	0	0.2	0.4	0.6	0.8	1.0
Average number of links broadcasted	0.625	0.561	0.561	0.561	0.561	0.561

TABLE I
Flooding performance for different α

Currently we are studying the effects of the component parameters for large networks.

References

[1] Qayyum A.and Viennot L.and Laouiti A. Multipoint relaying for flooding broadcast messages in mobile wireless networks. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences (HICSS'02)*, volume 9, page 298, 2002.
[2] Williams B. and Camp T. Comparison of broadcasting techniques for mobile ad-hoc networks. In *Proc. ACM International Symposium on Mobile Ad-Hoc Networking and Computing (MOBIHOC)*, 2002.
[3] T. Clausen, P. Jacquet (editors), C. Adjih, A. Laouiti, P. Minet, P. Muhlethaler, A. Qayyum, and L.Viennot. Optimized link state routing protocol (OLSR). RFC 3626, October 2003. Network Working Group.
[4] Kleinberg J. and Tardos E. *Algorithm Design*. Addison Wesley, 2005.
[5] Wu J. and Dai F. A generic distributed broadcast scheme in ad hoc networks. *Trans. on Computers*, 53(10):1343–1354, Oct. 2004.
[6] Baras J.S. and Huang H. Component based routing: A new methodology for designing routing protocols for MANET. In *Proceedings 25th Army Science Conference*, 2006.
[7] Baras J.S., Tabatabaee V., Purkayastha P., and Somasundaram K. Component based performance modeling of wireless routing protocols. In *Proc. IEEE ICC Ad Hoc and Sensor Networking Symposium.*, 2009.
[8] McGlynn M. and Borbash S.A. Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks. In *ACM Int. Symp. on mobile ad hoc networking and computing*, 2001.
[9] Jacquet P., Laouiti A., Minet P., and Viennot L. *Performance of multipoint relaying in ad hoc mobile routing protocols*. Springer, February 2004.
[10] V. Tabatabaee, J. S. Baras, P. Purkayastha, and K. Somasundaram. Component based modeling and design of wireless ad-hoc routing protocols. Technical report, ISR, Univ of Maryland, 2008.