THE INSTITUTE FOR SYSTEMS RESEARCH

ISR TECHNICAL REPORT 2010-3

Distributed Topology Control for Stable Path Routing in Mobile Ad Hoc Networks

Kiran Somasundaram, Kaustubh Jain,
Vahid Tabatabaee, John S. Baras

Distributed Topology Control for Stable Path Routing in Mobile Ad Hoc Networks

Kiran K. Somasundaram, Kaustubh Jain, Vahid Tabatabaee, John S. Baras
Institute of Systems Research and Department of Electrical and Computer Engineering
University of Maryland, College Park
College Park, MD 20742
Email: kirans@umd.edu, ksjain@umd.edu, vahidt@umd.edu, baras@umd.edu

***Abstract*—In this paper, we introduce the stable path topology control problem for routing in Mobile Ad Hoc Networks (MANETs). We formulate the problem as a constrained multi-agent optimization problem with only local neighborhood information. We develop and prove local pruning strategies that solve this problem. We also introduce the notion of distorted pruning, which offers a systematic method to trade path stability off against the hop count metric. Finally, we quantify the performance of our pruning algorithms using several simulation scenarios.**

***Index Terms*—long lived paths; lexicographic optimality; distributed pruning;**

I. Introduction

Topology control in MANETs has been a topic of active research in the recent years. A number of topology control mechanisms have been proposed for various purposes, including connectivity, energy-efficiency, throughput and robustness to mobility [1]. In particular, a number of topology control algorithms, both centralized and distributed, that are aimed to reduce the *broadcast storm problem* have been developed [2]. *Broadcasting* in a network refers to the process by which a packet sent from one station reaches all other stations in the network. Broadcast of link states is commonly used in MANET routing protocols to maintain routes [3], [4]. However in these mobile networks, link states are very dynamic, and consequently a large number of broadcast packets are sent across the network. This problem is referred to as the *broadcast storm problem* [2].

Topology control mechanisms such as those in *Optimized Link State Routing* (OLSR) reduce the broadcast storm problem by localized pruning of links on static graphs. However, MANETs are dynamic graphs, and these static graph approaches are limited in solving the broadcast storm problem. Although there are a number of metrics that capture the link dynamics, very few algorithms use these link metrics for topology control in MANET routing. Even those that do are only heuristic, which do not offer proof guarantees for the reduced topology and routing [5].

One important metric for routing in MANETs is *path longevity* or *path stability* [6]. In this paper, we refer to it as path stability. Although path stability has been studied for many reactive distance vector schemes [6], [7], there is little work that addresses topology control for stable paths in link state routing. We introduce a new topology control mechanism which guarantees stable path routing: *Stable paths topology control* is a mechanism to reduce the initial topology to reduce the broadcast storm and at the same time guarantees that the stable paths for routing (unicast) from every host to any target station are preserved in the reduced topology. Topology control for stable paths has a two-fold advantage: First, these long lived paths are *cheaper to maintain* (as they are less likely to change). Second, it offers the higher layer traffic long lived paths and consequently yields *improved throughput performance*. However, there is a tradeoff between the stable path and the shortest path routing. It has been shown that the stable paths in a mobile wireless network usually tend to be longer in hop count, and hence have more path delay [8].

The main contributions of this paper are the following. We introduce the *stable path topology control* problem. We formulate this problem as a *constrained distributed multi-agent optimization problem*, where the agents include all the stations in the network and have access to only local neighborhood information. We introduce the notion of *loop freedom* for this problem and present sufficient conditions to achieve it. We also develop a mathematical framework to systematically trade path stability off against hop count by introducing a *distorted pruning* construct. We prove that distorted pruning yields shorter hop count paths. We also show that we can recover OLSR's topology control mechanism as a special case of this distorted pruning. We emphasize that our goal is not to engineer link stability metrics, but to develop a general framework for the stable path topology control problem that can make use of available link stability metrics.

Our stable path topology control mechanism can be implemented with minor modifications to OLSR's neighbor discovery and topology selection mechanism. In this paper, we compare the performance of our topology control mechanism with that of OLSR using OPNET simulations [9]. For these simulations, we modified the default code of the OLSR model in OPNET to implement our topology control algorithm.

This paper is organized as follows. In Section II, we summarize the link and path stability metrics available in the literature. In Section III, we introduce the mathematical notation that we follow in this paper. In Section IV, we formulate the stable path topology control problem and develop perfect pruning algorithms to solve it. In Section V, we extend the perfect pruning algorithms to distorted pruning algorithms. Finally in Section VI, we quantify the performance of our pruning algorithms using several simulation scenarios. In this section, we also explain why OLSR and its variants are not capable of solving the stable path topology control problem.

II. Existing Stability Metrics and Related Work

The majority of the routing protocols proposed for MANETs, both reactive and proactive, are mechanisms that use *hop count* as the metric for route selection [10]. However, the communication links in a MANET are vulnerable to frequent breakage due to mobility and channel erasures [11], [12], [13]. Hence, schemes based merely on hop count, which are inherently insensitive to the stability of the paths, have shown poor performance [14]. This limitation has inspired a number of protocols that use *link stability* as a metric for routing.

Perhaps, the earliest MANET protocol to use link stability metric for routing is the *Associativity Based Routing* (ABR) scheme [15], which uses an *associativity* threshold that predicts the stability of a neighboring station. It assumes that the neighbors that remain associated beyond this threshold are less likely to move away and hence form stable neighbors or links. *Signal Stability based Adaptive routing* (SSA) [16] is another link stability based routing protocol that uses signal strength and location information from the neighboring stations to estimate stability of the links. *Route-lifetime Assessment Based Routing* (RABR) [17] is an extension to SSA that uses thresholding of link ages to choose routes. Mobility prediction was suggested in [11] to improve unicast and multicast routing protocols for MANETs. This scheme uses GPS location information to estimate the residual lifetime for links.

In [12], the authors present a simulation study of the empirical distribution of link lifetimes for various mobility models [18]. From these empirical distributions, they also derive a method to compute the residual life time distribution for these models. The study reveals that there are strict thresholds only beyond which the residual lifetimes exhibit a positive correlation with the link age. Another statistical characterization of link lifetimes is presented in [6]. Their simulation results show that the longer lifetime paths tend to have longer length (in hop count), and hence, there is a clear tradeoff between path stability and path delay. The *Stability and Hop-count based Routing algorithm* (SHARC) [8] identifies this tradeoff and combines the link stability metric and the hop count to find short paths (in terms of hop count) that also have good stability.

Another simulation study presented in [19] shows that the path life is inversely related to the maximum velocity and the hop count and is directly related to the transmission range. The authors observe that under high mobility patterns, the path durations can be approximated using exponential distributions. In [20], *Han* et. al. use Palm calculus to show that under certain conditions, the path durations converge to an exponential distribution as the number of hop count increases.

A number of link stability metrics have been proposed [15], [7], [21], [6], [22], [11], [8], [13], [16], [17] that capture the stability of the links in a MANET. However in almost all cases, these metrics have been only used in modifications of reactive distance vector protocols such as Dynamic Source Routing (DSR) [23] and *Ad Hoc On-demand Distance Vector* (AODV) [24]. There has been very little work on incorporating these link stability metrics for topology control in link state routing mechanisms such as OLSR [3]. In [5], the authors use the residual lifetimes suggested in [12] to modify the *set-cover* topology selection of OLSR. However, these set-cover methods offer no proof guarantees for routing. In Subsection VI-A, we show that this set-cover scheme and in general, any trivial modifications to OLSR cannot solve the stable path topology control problem. In the forthcoming sections, we develop a generalized topology control mechanism for stable path routing that can make use of any of the metrics introduced in this subsection.

III. Mathematical Notations and Definitions

A. Graphs and Neighborhoods

Let $G(V, E)$ denote the communication graph, where V is the vertex set of stations and E is the undirected edge set that captures the communication adjacency between vertices. For $(i, j) \in E$, there is an associated symmetric link stability metric $s(i, j) = s(j, i) \geq 0$. For MANETs, the edge set E and the link stability metrics $s(i, j),\ (i, j) \in E$, are time varying, and we implicitly assume this time dependence for all other functions defined on these quantities.

Next, we introduce the notion of hop-based neighborhoods. For any vertex pair $h, T \in V$, let $P_{(h,T)}$ denote the set of paths from h to T. The *hop count*, $hc : P_{(h,T)} \to Z_+$, of a path is defined as the number of edges in the path. Then the minimal hop count distance between a pair of vertices (h, T) is defined as

$$d_{hc}(h, T) = \min_{p \in P_{(h,T)}} hc(p).$$

We define the *k-hop neighborhood* for $h \in V$ by

$$N_h^k = \{j \in V : d_{hc}(h, j) \leq k\}.$$

Here, k is called the size of the neighborhood. The boundary for the set N_h^k is given by

$$\partial N_h^k = N_h^k \backslash N_h^{k-1},$$

where $N_h^0 = \{h\}$, and $N_h^k = \emptyset, k < 0$. For $V' \subseteq V$, the subgraph induced by V' is denoted by the graph restriction $G|_{V'}(V', E|_{V'})$ (an *induced subgraph* $G|_{V'}$ is a subgraph of G that only contains the vertices in V' and the edges of G that connect these vertices, $E|_{V'}$). For instance, the subgraph induced by $V' = N_h^k$ is denoted by $G|_{N_h^k}(N_i^k, E|_{N_h^k})$.

B. Path Stability

For all protocols discussed in Section II, the path stability metric, $w(p),\ p \in P_{(h,T)}$, is computed from the link stability metrics $s(i,j)$ by either an additive composition

$$w(p) = \sum_{(i,j)\in p} s(i,j),$$

or a bottleneck composition

$$w(p) = \min_{(i,j)\in p} s(i,j).$$

In this paper, we treat only the bottleneck composition for the topology control algorithms. The algorithms and the proof methods for the additive composition are simple extensions. The optimal path stability metric between a host-target vertex pair (h,T) is

$$\begin{aligned} w^*(h,T) &= \max_{p\in P_{(h,T)}} w(p), \\ &= \max_{p\in P_{(h,T)}} \min_{(i,j)\in p} s(i,j). \end{aligned}$$

The corresponding optimal path set,

$$P^*_{(h,T)} = \arg\max_{p\in P_{(h,T)}} w(p)$$

is the set of paths that have the optimal path stability metric from h to T.

C. Lexicographic Optimality of Paths

Lexicographic optimality is a common multicriteria optimization method if there is an order ranking between the different criteria [25]. In this paper, we consider a lexicographic ordering where the stability metric is ranked higher than the hop count metric, i.e., for $p_1, p_2 \in P_{(h,T)}$,

$$\begin{bmatrix} w(p_1) \\ -hc(p_1) \end{bmatrix} \geq_{lex} \begin{bmatrix} w(p_2) \\ -hc(p_2) \end{bmatrix}$$

if $w(p_1) > w(p_2)$, or $w(p_1) = w(p_2)$ and $hc(p_1) \leq hc(p_2)$.

D. Heartbeat Neighbor Discovery and Local Link State Exchange

We assume that every host $h \in V$ has a neighbor discovery module similar to those in [5], [4], where the stations learn about their local neighborhoods using periodic *HELLO* messages. Consequently, every host $h \in V$ has access to the dynamic graph $G|_{N_h^k}$, where k is the *size of the neighborhood* exposed by the neighbor discovery mechanism. For instance, in OLSR $k = 2$ because all hosts $(h \in V)$ exchange the link state information about their neighbors (∂N_h^1). We also assume that the hosts exchange local stability information $(s(h,j),\ j \in \partial N_h^1)$ in the periodic HELLO messages. Thus host h, apart from learning $G|_{N_h^k}$, also learns $s(i,j),\ (i,j) \in E|_{N_h^k}$.

IV. Local Pruning

Topology control by local pruning [2] in multi-hop wireless networks is an interesting graph optimization problem that has been studied in the last decade. Most of these pruning algorithms propose solutions to reduce the *broadcast storm* problem [2]. In such pruning algorithms, the host $h \in V$, which has learned $G|_{N_h^k}$, chooses a subset of neighbors, $R(h) \subseteq \partial N_h^1$, and marks them as *significant relay neighbors*. Then the host h uses $R(h)$ as relays to achieve network-wide broadcast. If the subset $R(h)$ is small compared to ∂N_h^1, then the number of redundant broadcast messages is significantly reduced. In [26], the authors show that these local pruning methods, in essence, try to construct a *Connected Dominating Set* (CDS) for the dynamic graph G with only local information. For every construction discussed in [26], the objective function over the CDS is different. For instance in [27], *Lee* et. al. present a CDS construction that is tradeoff efficient w.r.t. energy and graph resiliency. In OLSR [3], the *Multi-Point Relay* (MPR) set, which is constructed using local set covers, yields a global CDS. However, these CDS constructions do not offer any guarantees on the quality of the routing paths in the pruned CDS. To the best of our knowledge, there have been no local pruning mechanisms that offer path quality guarantees on the pruned global graph.

In this paper, we consider the topology control mechanism that preserves the stable routing paths by local pruning. In the forthcoming subsections, we motivate this pruning problem as a constrained multi-agent optimization problem on the vertices of a graph and show strategies that solve this problem.

A. Local and Global Views

In [28], we extended the notion of local and global views introduced in [26] to encompass dynamic graphs with edge weights. We summarize here this extension. As per the neighbor discovery assumptions of Subsection III-D, every host learns its local dynamic neighborhood. This is formally defined as follows:

Definition: At every host station, $h \in V$, the *local view* consists of the graph restriction $G|_{N_h^k}$ and the edge weights $s(i,j),\ (i,j) \in E|_{N_h^k}$, where k is the size of the neighborhood exposed by the neighbor discovery mechanism at h.

As previously explained, a pruning algorithm at every host h selects a subset of its neighbors, $R(h) \subseteq \partial N_h^1$. Every host h then broadcasts the edge set $\{(h,j) : \ j \in R(h)\}$. The corresponding broadcast edge set is given by $E_{broadcast} = \cup_{h\in V}\{(h,j) : \ j \in R(h)\}$, which induces the subgraph $G_{broadcast}$. If the broadcast is successful, then every host, $h \in V$, has access to the $G_{broadcast}$. In some link state mechanisms, such as in [29], the edge weights, $s(h,j),\ j \in R(h)$, are also broadcast for routing. In this case, if the broadcast is successful, then every host, $h \in V$, has access to $s(i,j),\ (i,j) \in E_{broadcast}$.

Definition: At every host station, $h \in V$, the *global graph view* G_h^{global} is the graph union $G|_{N_h^k} \cup G_{broadcast}$ that is exposed by the neighbor discovery and link state broadcast mechanism respectively. Depending on the nature of the link state broadcast mechanism, every host h has access to $s(i,j)$ either for $(i,j) \in E_{broadcast} \cup E|_{N_h^k}$ or only for $(i,j) \in E|_{N_h^k}$.

B. Pruning Problem

The fundamental pruning problem for each host $h \in V$ is to construct a small set $R(h) \subseteq \partial N_h^1$ such that the graph G_h^{global} has some required properties. Thus, every host h has an optimization problem with an uncoupled (independent) objective function. The optimization problems (for these hosts) are coupled by the global view constraint. This naturally lends itself to the following constrained multi-agent optimization problem. Every station $h \in V$ solves the pruning problem given its *local view*:

$$\begin{array}{ll} \min\limits_{R(h) \in 2^{\partial N_h^1}} & |R(h)| \\ \text{subject to} & G_h^{global} \in \Pi(G), \end{array}$$

where $\Pi(G)$ is the set of all subgraphs of G that satisfy some property π. The constraint $G_h^{global} \in \Pi(G)$ requires the pruned graph G_h^{global} to preserve the property π. For instance, in the different broadcast schemes illustrated in [2], the property π corresponds to different CDS constraints. For stable path routing, the property π for each host h is given by

$$\pi_s^h : \exists p \in P^*_{(h,T)},\ T \in V,$$

i.e., $\Pi_s^h(G)$ corresponds to all the subgraphs of G that have at least one of the most stable paths between the host h and any other vertex $T \in V$.

C. Loop Freedom in Distributed Pruning

One of the major difficulties for algorithms working on distributed processors is to ensure *loop freedom* [10]. Loops typically occur in distributed algorithms when tie-breaking mechanisms are not employed among the processors. Using an example, we illustrate that a similar problem is likely to occur in distributed pruning without tie-breaking.

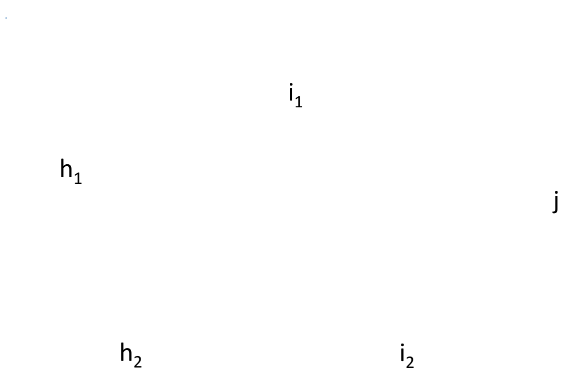


Fig. 1: Loops in pruning

Consider the example graph shown in Figure 1. Let us consider the connectedness property at host $h \in V$,

$$\pi_c^h : \exists p \in P_{(h,T)},\ T \in V.$$

The connectedness property ensures that $G_h^{broadcast}$ is connected from h. Consider the pruning mechanism at stations h_1 and h_2 for a connection to station j. For h_1, there are two paths (h_1, i_1, j) and (h_1, h_2, i_2, j). Similarly for h_2, there are two paths (h_2, i_2, j) and (h_2, h_1, i_1, j). For both stations, both paths satisfy π_c^h. Thus without tie-breaking mechanisms, station h_1 can choose h_2, and h_2 can choose h_1 thereby violating π_c^h for both h_1 and h_2. In this paper, we adopt a tie-breaking mechanism with lexicographic ordering (introduced in Subsection III-C) for distributed stable path pruning to ensure loop freedom.

D. Perfect Pruning for Stable Paths

The constrained multi-agent optimization problem that the hosts $h \in V$ solve for stable path pruning is

$$\begin{array}{lll} \min\limits_{R(h) \in 2^{\partial N_h^1}} & |R(h)| & \textbf{(Pb1)} \\ \text{subject to} & G_h^{global} \in \Pi_s^h(G). & \end{array}$$

Every host h has the local view, $G|_{N_h^k}$ and $s(i,j),\ (i,j) \in E|_{N_h^k}$, as input to Pb1. Given a general graph G, for $G_h^{global} \in \Pi_s^h(G)$, it is clearly necessary for every pruning station $h \in V$ to preserve its most stable path to $l \in \partial N_h^k$ (*necessary condition for stable path pruning*). However, this is not a sufficient condition because it does not ensure loop freedom. In this subsection, we develop a multi-stage procedure (combination of Algorithms 1 and 2, and problem Pb2) that is a sufficient condition for stable path pruning.

In essence, Algorithms 1 and 2 compute the lexicographically optimal paths from the host h to $l \in \partial N_h^k$. By breaking it into two algorithms, we save the cost of storing intermediate paths. Algorithm 1 is a modified version of the standard Floyd-Warshall algorithm [30], which computes the all-pair lexicographically optimal paths among all nodes in N_h^k excluding h. This algorithm only stores the lexicographic optimal path metrics, but not the corresponding paths. Algorithm 2 accepts the output from Algorithm 1, and computes the lexicographically optimal paths from h to $l \in \partial N_h^k$. It also computes the next hop from the host for every lexicographically optimal path (including paths tied in the lexicographic order) to $l \in \partial N_h^k$.

Let the nodes in N_h^k be numbered from 1 to $|N_h^k|$, where the host h is numbered as 1. The *one-hop* neighbors, ∂N_h^1, are numbered from 2 to $|\partial N_h^1| + 1$. The neighborhood boundary nodes, ∂N_h^k are numbered from $n^k = |N_h^k| - |\partial N_h^k| + 1$ to $|N_h^k|$. Let $\mathbf{A}_h = [a_h(i,j)]$ be the $|N_h^k| \times |N_h^k|$ adjacency matrix of $G|_{N_h^k}$ based on the numbering scheme described. The link stability matrix is $\mathbf{S}_h = [s_h(i,j)]$, where

$$s_h(i,j) = \begin{cases} s(i,j) & a_h(i,j) = 1 \\ 0 & \textit{otherwise}. \end{cases}$$

As described above, Algorithm 1 computes all-pair lexicographically optimal paths (as defined in Subsection III-C) for all the vertices in $\{2, 3, \ldots, |N_h^k|\}$. It outputs two matrices $\mathbf{W}(|N_h^k|)$ and $\mathbf{H}(|N_h^k|)$ that contain the stability and hop count of these lexicographically optimal paths. Note that the algorithm does not store the optimal paths. Also, the first row and column of $\mathbf{W}$ and $\mathbf{H}$, which correspond to the host h, are not affected by the algorithm.

Algorithm 1 Floyd-Warshall function for the lexicographic path problem at station h

Input: $\mathbf{A}_h$, $\mathbf{S}_h$
Init: $\mathbf{W}(1) \leftarrow \mathbf{S}_h$, $\mathbf{H}(1) \leftarrow \mathbf{A}_h$;
for $m = 2$ to $|N_h^k|$ **do**
  for all $i, j \in \{2, 3, \cdots, |N_h^k|\}$ and $i \neq j$ **do**
    if $w_{ij}(m-1) > \min\{w_{im}(m-1), w_{mj}(m-1)\}$ **then**
      $w_{ij}(m) \leftarrow w_{ij}(m-1)$;
      $h_{ij}(m) \leftarrow h_{ij}(m-1)$;
    else if $w_{ij}(m-1) < \min\{w_{im}(m-1), w_{mj}(m-1)\}$ **then**
      $w_{ij}(m) \leftarrow \min\{w_{im}(m-1), w_{mj}(m-1)\}$;
      $h_{ij}(m) \leftarrow h_{im}(m-1) + h_{mj}(m-1)$;
    else
      $w_{ij}(m) \leftarrow w_{ij}(m-1)$;
      $h_{ij}(m) \leftarrow \min\{h_{ij}(m-1),\ h_{im}(m-1)+h_{mj}(m-1)\}$;
    end if
  end for
end for
Output: $\mathbf{W}(|N_h^k|)$, $\mathbf{H}(|N_h^k|)$;

Algorithm 2 Vertex expansion at h

Input: $\mathbf{W}(|N_h^k|)$, $\mathbf{H}(|N_h^k|)$
Init: `List[`$|\partial N_h^k|$`]` L_{cover};
for all $l \in \{n^k, n^k+1, \cdots, |\partial N_h^k|\}$ **do**
  // Init elements to their identities
  $w_{1l}(|N_h^k|+1) \leftarrow 0$;
  $h_{1l}(|N_h^k|+1) \leftarrow \infty$;

  for $j = 2$ to $|\partial N_h^1|$ **do**
    if $w_{1l}(|N_h^k|+1) < \min\{s(1,j), w_{jl}(|N_h^k|)\}$ **then**
      $w_{1l}(|N_h^k|+1) \leftarrow \min\{s(1,j), w_{jl}(|N_h^k|)\}$;
      $h_{1l}(|N_h^k|+1) \leftarrow 1 + h_{jl}(|N_h^k|)$;
    else if $w_{1l}(|N_h^k|+1) = \min\{s(1,j), w_{jl}(|N_h^k|)\}$ **then**
      $h_{1l}(|N_h^k|+1) \leftarrow \min\{1+h_{jl}(|N_h^k|), h_{1l}(|N_h^k|+1)\}$
    end if
  end for
  $L_{cover}(l) \leftarrow \{j \in \partial N_h^1 : w_{1l}(|N_h^k|+1) = \min\{s(1,j), w_{jl}(|N_h^k|)\},\ h_{1l}(|N_h^k|+1) = 1 + h_{jl}(|N_h^k|)\}$;
end for
Output: $w_{1l}(|N_h^k|+1), h_{1l}(|N_h^k|+1)$, $l \in \partial N_h^k$, L_{cover};

Algorithm 2 computes the lexicographically optimal paths from host h to $l \in \partial N_h^k$ using the output of Algorithm 1. The stability and hop count of these lexicographically optimal paths are returned in $w_{1l}(|\partial N_h^k|+1)$ and $h_{1l}(|\partial N_h^k|+1)$. The array element $L_{cover}(l)$ contains the list of all one-hop neighbors that are on the lexicographically optimal paths from h to l ($L_{cover}(l) \subseteq \partial N_h^1$). We now construct a covering problem using $L_{cover}(l)$, $l \in \partial N_h^k$. The problem is to *cover all k-hop neighbors, ∂N_h^k, using minimal number of one-hop neighbors, ∂N_h^1, that lie on the lexicographically optimal paths*. We define

$$covered : \partial N_h^1 \to 2^{\partial N_h^k},$$
$$covered(j) \mapsto \{l \in \partial N_h^k :\ j \in L_{cover}(l)\}.$$

Then the covering problem is stated as

$$\min_{R(h)\in 2^{\partial N_h^1}} \quad |R(h)| \qquad \textbf{(Pb2)}$$
$$\text{subject to} \quad \cup_{j\in R(h)} covered(j) = \partial N_h^k.$$

Let $R_{Pb2}(h)$ be the solution to Pb2. Construct the broadcast edge set $\{(h,j) : j \in R_{Pb2}(h)\}$ for every $h \in V$. Let the corresponding global graph view be $G_h^{globalPb2}$.

Theorem 4.1: $R_{Pb2}(h),\ h \in V$ is a solution to the constrained multi-agent optimization problem Pb1.

Proof: The objective minimization of Pb2 ensures that the objective functions for each $h \in V$ in Pb1 are minimized. It remains to be proved that $G_h^{global} \in \Pi_s^h(G),\ \forall h \in V$. We will prove by construction that for any target station $T \in V$, there exists a path $p \in P^*_{(T,h)}$ in G_h^{global}.

Let $M = \min_{p\in P^*_{(T,h)}} hc(p)$ and $P'^*_{(T,h)} = \{p \in P^*_{(T,h)} : hc(p) = M\}$. We need to consider two cases:
Case I: $M \leq k$. Then by definition, $T \in N_h^k$, and hence $p \in P^*_{(T,h)}$ is visible in the local view. Thus p is in G_h^{global}.
Case II: $M > k$. The pruning mechanism at station T (Algorithms 1, 2 and Pb2), chooses a subset of one-hop neighbors $R_{Pb2}(h)$ that lie on the lexicographically optimal path to every $l \in \partial N_T^k$. By *Bellman's* principle, every sub-path of a lexicographically optimal path is lexicographically optimal. Thus, $\exists T^1 \in R_{Pb2}(T)$ that lies on one lexicographically optimal path $p \in P'^*_{(T,h)}$. By construction $p = (T, p^1)$ where p^1 is a lexicographically optimal path from T^1 to h (by *Bellman's* principle) and $hc(p^1) = M - 1$. We repeat this construction at T^1 to obtain $p = (T, T^1, p^2)$, where p^2 is a lexicographically optimal path from T^2 to h and $hc(p^2) = M - 2$. By induction, we construct a lexicographically optimal path $p = (T, T^1, T^2, \cdots, T^{M-k+1}, p^{M-k})$, where p^{M-k} is the lexicographically optimal path from T^{M-k} to h and $hc(p^{M-k}) = k$. And p^{M-k} satisfies the condition in Case I. The proof is complete since we have constructed a path $p \in P^*_{(T,h)}$ in G_h^{global}. ■

Corollary 4.2: $G_h^{globalPb2}$ preserves the most stable routing paths that have the least hop count.

Proof: The proof follows trivially from the lexicographic path construction of Thm. 4.1. ■

This corollary establishes that the pruning mechanism yields paths that are sensitive to both stability and hop count. This intuitively implies that the pruned paths in $G_{broadcast}$ not only have good path longevity, but also have acceptable path delays.

Note that the primary difference between our algorithm and OLSR's topology control is the means by which paths are constructed locally from h to $l \in \partial N_h^2$. In OLSR, only two-hop paths from h to $l \in \partial N_h^2$ are considered for the MPR selection problem. However, in our algorithm the lexicographically optimal paths are not restricted by this hop count constraint. This difference is illustrated in Subsection VI-A.

E. Greedy Set-Cover Solution

We need to solve Pb2 to complete the pruning procedure. Problem Pb2 is a form of the set-cover problem [30]. It belongs to the class of NP-complete problems. There are a number of approximation algorithms for this problem [30]. In this paper, we choose a popular greedy algorithm suggested in [30] to solve Pb2.

Algorithm 3 Greedy Cover Algorithm at h

Input: $L_{cover}(l),\ l \in \partial N_h^k$
Init: $R_{greedy}(h) \leftarrow \emptyset$, $U \leftarrow \partial N_h^k$;

// Find and append essential cover elements
for all $\{l \in \partial N_h^k : \ |L_{cover}(l)| = 1\ \}$ **do**
  $R_{greedy}(h) \leftarrow R_{greedy}(h) \cup \{L_{cover}(l)\}$;
  $U \leftarrow U \backslash \{l\}$;
end for

// Greedy selection
while $U \neq \emptyset$ **do**
  $j^* \leftarrow \arg\max_{j \in \partial N_h^1} |\{l \in U : \ j \in L_{cover}(l)\}|$
  $R_{greedy}(h) \leftarrow R_{greedy}(h) \cup \{j^*\}$
  $U \leftarrow U \backslash \{l \in U : \ j^* \in L_{cover}(l)\}$
end while
Output: $R_{greedy}(h)$

Let $d^* = \max_{j \in \partial N_h^1} |covered(j)|$. Then the following lemma gives the approximation bounds for the greedy solution $R_{greedy}(h)$:

Lemma 4.3: $|R_{greedy}(h)| \leq H(d^*)|R_{Pb2}(h)|$, where $H(N) = \sum_{n=1}^{N} \frac{1}{n}$.

This lemma is proved in Chapter 11 of [30].

V. Distorted Pruning

The perfect pruning procedure described in Subsection IV-D is a sufficient mechanism for preserving the most stable path from h to any target vertex T in G_h^{global}. But in many cases, like those discussed in [8], [6], the stable paths tend to be longer (in hop count). The simulation studies in [6] show that for both *TCP* and *UDP* traffic, the path delays for long lived paths are much larger than those for the shortest hop count paths. We develop a distorted pruning method to systematically trade path stability off against shorter hop count paths. There are several ways to apply distortion to the path stability metric. We choose to formulate it as a truncation problem. The standard truncation function parameterized by the truncation parameter Δ is defined for $x \in \mathbb{R}^+$ using the floor function:

$$trunc(x, \Delta) = \begin{cases} \lfloor \frac{x}{\Delta} \rfloor \times \Delta & \text{if } \Delta \in \mathbb{N} \\ 0 & \text{if } \Delta = \infty \end{cases}$$

Algorithm 4 Distorted Pruning Procedure

Input: $\mathbf{A}_h$, $\mathbf{S}_h$, Δ
$\mathbf{S}_h^{\Delta} \leftarrow trunc(\mathbf{S}_h, \Delta)$
$\mathbf{W}(|N_h^k|)$, $\mathbf{H}(|N_h^k|) \leftarrow$ *floydWarshallForLexicographicPathProblem* $(\mathbf{A}_h, \mathbf{S}_h^{\Delta})$;
$L_{cover} \leftarrow$ *vertexExpansion*$(\mathbf{W}(|N_h^k|),\ \mathbf{H}(|N_h^k|))$;
$R_{greedy}^{\Delta}(h) \leftarrow$ *greedyCover*(L_{cover});
OUTPUT: $R_{greedy}^{\Delta}(h)$

At every $h \in V$, we truncate the link stability metric $s(i,j),\ \ (i,j) \in E|_{N_h^k}$. Then the distorted pruning procedure is constructed using the truncation function and the algorithms described in Subsection IV-D. This is shown in Algorithm 4. At every $h \in V$, the algorithm outputs a greedy cover $R_{greedy}^{\Delta}(h)$. The constructed broadcast edge set is $E_{broadcast(\Delta)} = \{(h,j) :\ \ j \in R_{greedy}^{\Delta}(h)\}$, and the corresponding global graph view is denoted by $G_h^{global(\Delta)}$. We need to show that this pruning preserves shorter paths compared to those preserved in G_h^{global} (obtained by perfect pruning). Let $P_{(h,T)}|_{G_h^{global}}$ and $P_{(h,T)}|_{G_h^{global(\Delta)}}$ denote the set of paths from h to T in G_h^{global} and $G_h^{global(\Delta)}$ respectively. $w^{\Delta}(p) = \min\limits_{(i,j)\in p} trunc(s(i,j), \Delta)$ denotes the truncated path stability metric.

Theorem 5.1: For $p \in P_{(h,T)}|_{G_h^{global}}$, there exists a $p^{\Delta} \in P_{(h,T)}|_{G_h^{global(\Delta)}}$ such that $hc(p^{\Delta}) \leq hc(p)$.

Proof: We will assume that there exists no such p^{Δ} preserved in $P_{(h,T)}|_{G_h^{global(\Delta)}}$ such that $hc(p^{\Delta}) \leq hc(p),\ p \in P_{(h,T)}|_{G_h^{global)}}$, and derive a contradiction. Theorem 4.1 and Corollary 4.2 establish that our pruning mechanism preserves the lexicographically optimal paths from h to every station T. Consider a lexicographically optimal path $p^* \in P_{(h,T)}|_{G_h^{global}}$. Then p^* is not preserved in $P_{(h,T)}|_{G_h^{global(\Delta)}}$ if there exists a p^{Δ} such that

Case I:

$$\begin{aligned} w^{\Delta}(p^{\Delta}) &> w^{\Delta}(p^*) \\ \Rightarrow \min_{(i,j)\in p^{\Delta}} trunc(s(i,j),\Delta) &> \min_{(i,j)\in p^*} trunc(s(i,j),\Delta) \\ \Rightarrow \min_{(i,j)\in p^{\Delta}} s(i,j) &> \min_{(i,j)\in p^*} s(i,j) \\ \Rightarrow w(p^{\Delta}) &> w(p^*) \end{aligned}$$

But this contradicts the lexicographic optimality of p^*.

Case II: $w^{\Delta}(p^{\Delta}) = w^{\Delta}(p^*)$ and $hc(p^{\Delta}) < hc(p^*)$. But this contradicts our initial assumption. ■

Note that this proof works only for the bottleneck metric and cannot be extended to the additive metric. It is trivial to

Group	Parameter	Value
MAC and PHY	Protocol	802.11b
	Transmission Rate	11 Mbps
	Transmit Power	5 mW
	Receiver Sensitivity	-95 dBm
	Error Correction Capabilities	None
Routing and TC	Protocol	OLSR or SPTC
	HELLO message interval	2 s
	Neighbor hold time	6 s
	TC message interval	5 s
	TC hold time	15 s
Traffic	Type	UDP CBR
	Packet length	1024 bits

TABLE I: Parameters for simulation

show that we can recover OLSR's topology control mechanism from our distorted pruning using $\Delta = \infty$. This is because at $\Delta = \infty$ all paths become equivalent in the stability metric and the least hop count paths are preserved. By observing that OLSR's topology control is a special case of our distorted pruning algorithm, we also obtain an alternative (and a more general) proof method to that presented in [31] for the *shortest path preserving property* of topology control mechanisms.

VI. Simulation Results

All simulations were carried out in OPNET Modeler 14.5 [9]. For the simulations, the mobile node model *manet station* is chosen. Unless specified otherwise, the parameters given in Table I were used in the simulations.

We modified the default code for the OLSR model to implement our Stable Path Topology Control (SPTC) algorithm in OPNET, which includes the modification of the HELLO messages (to carry the link stability metrics) and the topology selection mechanisms (for stable paths). To study the performance of SPTC, we simulated several scenarios that are presented in the forthcoming subsections. In each of these scenarios, we used a suitable metric that captures link stability. We emphasize that these metrics are only based on heuristics and are intended only to serve as a proof of concept. Our focus in this paper is not to engineer these metrics, but to develop a general topology control mechanism for stable path routing. In all these simulations, the size of the neighborhood exposed by the neighbor discovery mechanism is $k = 2$.

In the simulations, we compared both the data traffic carrying and Topology Control (TC) overhead performance of SPTC and OLSR. For the data traffic carrying performance, we studied the *carried load* for various *offered loads*. We set up a *UDP* traffic generator that sends a *Constant Bit Rate* (CBR) traffic between pairs of stations. We then swept across this CBR rate to study the traffic performance with OLSR and SPTC for various truncation parameters. In the forthcoming subsections, we present the sample mean of the carried load for each scenario.

In link state mechanisms such as OLSR, the TC broadcast mechanism is proactive, and consequently not all TC messages broadcast correspond to topology changes. To study the overhead due to topology changes, we measured the rate of reactive TC messages. Reactive TC messages are those that are generated due to changes in the selected topology $(R(h),\ h \in V)$ as the G changes. This is a good estimate of the actual rate of topology change in the pruned network. Let $r_{TC}(SPTC)$ and $r_{TC}(OLSR)$ denote the reactive TC traffic rate in *kbps* for SPTC and OLSR respectively. Then

$$\%G_{TC} = \frac{r_{TC}(OLSR) - r_{TC}(SPTC)}{r_{TC}(OLSR)}$$

indicates the percentage saving in the TC overhead for SPTC. In all the scenarios, along with $\%G_{TC}$, we will also mention $r_{TC}(OLSR)$ to better quantify the saving.

A. Limitation of OLSR's MPR construction

In OLSR, the size of the neighborhood is $k = 2$. Every host $h \in V$ chooses a minimal MPR set, $MPR(h) \subseteq \partial N_h^1$, that *covers* ∂N_h^2 [3]. The 4 node network shown in Fig. 2 is an illustrative example that explains the limitations of these covering constructions. The scenario was set up such that the wireless link from node a to node b is unstable (the link goes ON and OFF frequently) due to radio propagation losses, but the other links are stable.

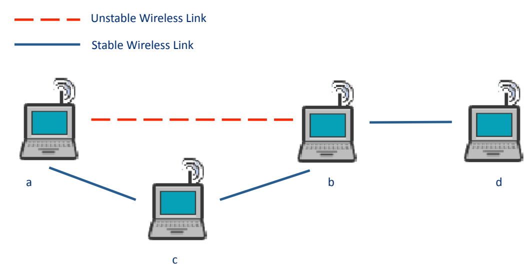
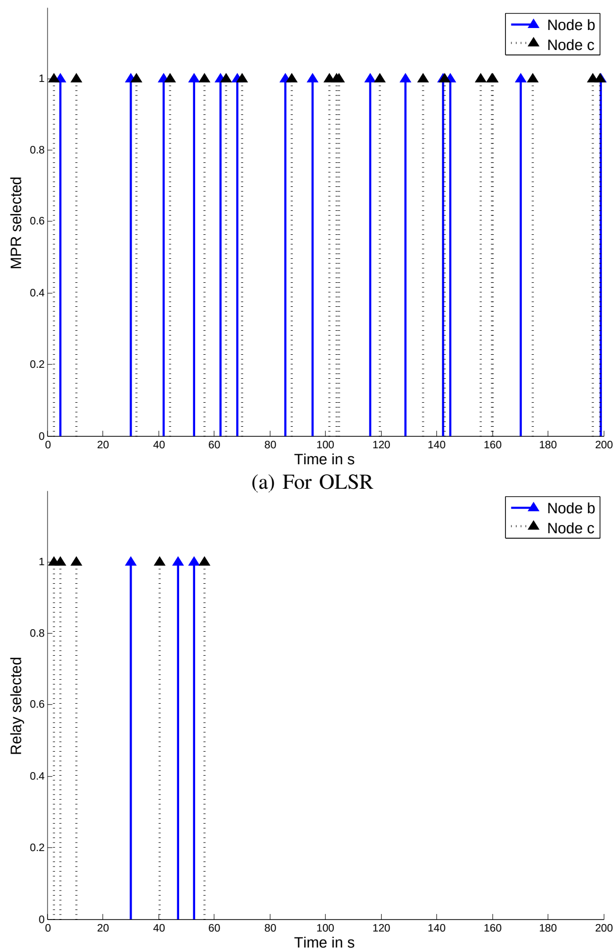


Fig. 2: 4 node static network

Let us consider OLSR's MPR selection process at node a. Since the link (a, b) is unstable, it goes ON and OFF frequently. Whenever this link is ON, $d \in \partial N_a^2$ and is covered only by node b. As a result, node a chooses node b as its only MPR. And when the link is OFF, $b \in \partial N_a^2$ and is covered only by node c. As a result, node a chooses node c as its only MPR. This is depicted in Subfigure 3a, which shows a realization of the point process for relay selection at node a. It illustrates that node a keeps changing its MPR between node b and c. Since a Topology Control (TC) message is broadcast for every change in the MPR set, the high link dynamics translates to a significant broadcast storm. As explained in Subsection IV-D, the limitation in this topology selection mechanism arises from the manner in which the host a tries to reach its two hop neighbor. When (a, b) is ON, the host is forced to choose b as an MPR. However, there is a stable alternative path (a, c, b), which cannot be used in OLSR's topology selection. The modification to OLSR in [5] suffers from the same limitation. In general, this problem would occur in any CDS scheme that uses the local set-cover construction (even if it uses link stability metrics).

For this static network, a good link stability metric for SPTC is the *current age* of the link, i.e., if the current age of a



(a) For OLSR

(b) For SPTC

Fig. 3: Realizations of the point process of relay selection

link is high, it is rated as stable; otherwise, it is unstable. SPTC circumvents the aforementioned problem, i.e., whether $d \in \partial N_a^2$ or $b \in \partial N_a^2$, the unstable link (a, b) is never chosen for broadcast. This is because, whenever the link (a, b) exists, there is an better alternative path through node c to reach b, i.e.,

$$w(a, c, b) > w(a, b).$$

Thus node c serves as a stable MPR for node a. This is illustrated in a realization of the point process of relay selection in Subfigure 3b. Once the link ages are high enough, the link stability metric is able to discern between the unstable and stable links, and the relay selection process converges.

B. Static Grid Topology

Consider a 25-node static grid topology shown in Fig. 4. The network consists of many stable and unstable links. The nature of the incident links at one of the stations is shown in Fig. 4. This topology suffers from the same problem explained in Subsection VI-A. The one-hop neighbors that are far off (in physical distance) typically cover more two-hop neighbors. However, by the nature of radio propagation, these links are unstable. Again for this static scenario, we chose the *current*

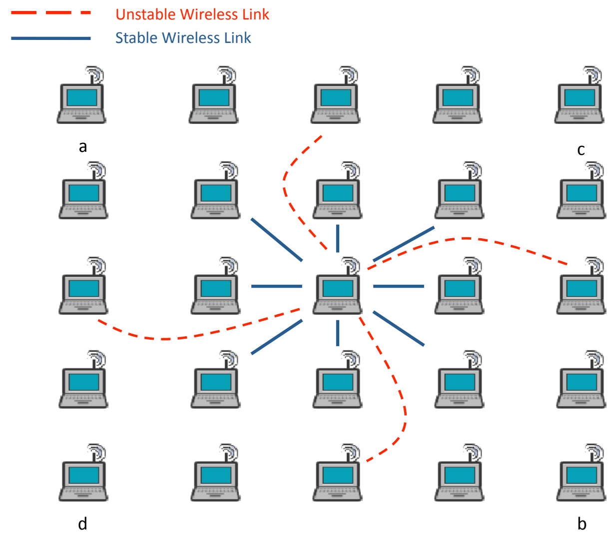


Fig. 4: 25-node grid network

Δ	1	5	25	50	100
$\%G_{TC}$	62.6565	59.0382	44.4224	35.8546	34.5826

TABLE II: TC traffic overhead for SPTC in 25 node grid, $r_{TC}(OLSR) = 8.437kbps$

age as the link stability metric for SPTC. The UDP traffic generator sent traffic from node a to b and from node c and d. The traffic performance for both SPTC and OLSR are shown in Fig. 5. The simulation results show that the SPTC is able to carry about 10% more traffic than OLSR. We also observe that the carried load is not significantly influenced by varying Δ.

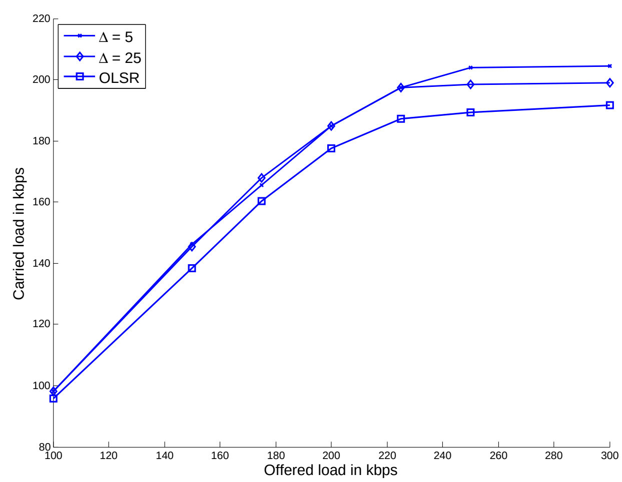


Fig. 5: Carried load vs. Offered load for 25-node grid

The TC overhead study, shown in Table II, indicates significant gains – as high as 60 % – in the rate of the TC traffic broadcast. This implies that the pruned subnetwork for SPTC is stable/long lived compared to the OLSR's MPR subnetwork.

Parameter	Value
No of stations	25
Simulation Area	100m × 100m
Speed	(0,1.5] m/s
Pause time	0 s
Transmit Power	10 μW

TABLE III: Random waypoint parameters

C. Random Waypoint Mobility Scenario

Random waypoint mobility pattern is a commonly used simulation setup to study protocol performances in a mobile environment [18]. The simulation parameters that we used for the random waypoint mobility pattern are shown in Table III. Note that the radio transmission power was set to $10\mu W$, which gave a radio range of approximately $50m$ (with the default PHY layer path loss models in OPNET) when the receiver sensitivity is $-95dBm$. This simulation was set up to match with the node density and transmission range of the random waypoint simulations in [11]. All statistics were collected once the simulations reached stochastic stationarity. However, we observe a very different link age distribution (Fig. 6) compared to those reported in [11]. This is due to the realistic PHY layer loss models of OPNET (which exhibit higher losses at the boundary of the transmission range) as opposed to the simple disc models. Fig. 6 indicates that a major fraction of lifetimes is centered around $6s$, which corresponds to the *neighbor hold time* (this is a hysteresis parameter in OLSR for the neighbor discovery mechanism). This implies that the majority of the links are short lived, and consequently this scenario becomes a good candidate to demonstrate SPTC.

We apply the formula presented in Section 5 of [11] to compute the expected residual lifetime conditioned on the current age. This yields a residual lifetime distribution shown in Fig. 7, which we used as the link stability metric.

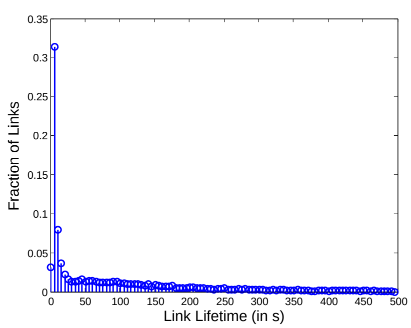


Fig. 6: Link lifetime distribution for random waypoint scenario

We chose a random source-target node pair for the UDP packet generator. The sample mean of the carried load as a function of the offered load is shown in Fig. 8. We observe that SPTC is capable of carrying more load than OLSR because compared to SPTC, in OLSR significantly more traffic is routed through unstable links.

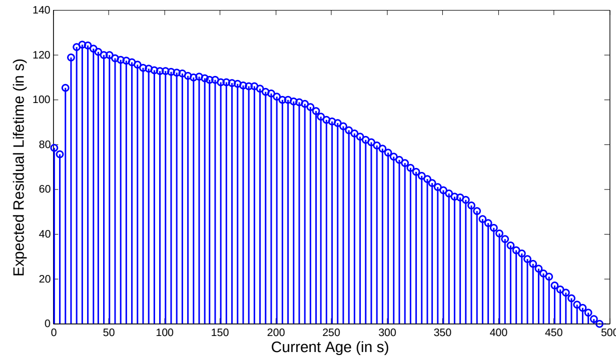


Fig. 7: Link Life Metric: Expected residual lifetime as a function of current age

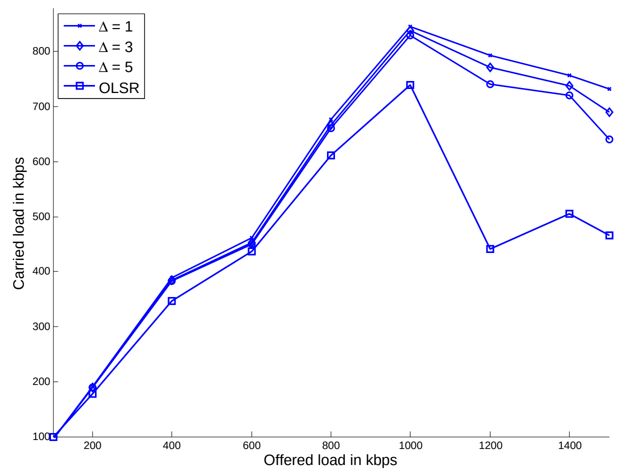


Fig. 8: Carried load vs. Offered load of Random Waypoint

However, this performance improvement is achieved at the cost of increased TC traffic. Table IV shows that for the random waypoint scenario, the $r_{TC}(SPTC)$ is significantly higher compared to $r_{TC}(OLSR)$. In particular, when $\Delta = 5$ the TC traffic rate is *6.254 kbps*, which corresponds to a percentage overhead of more than 100% over OLSR. However, note that this TC traffic is not prohibitive for the network and is small compared to the traffic carried by the network (as illustrated in Fig. 8).

D. Battlefield Scenario

Finally, we consider a battlefield scenario, introduced in [32], with an initial topology as shown in Subfigure 9a. It comprises of 3 platoons of stations: Platoon A consists of nodes 0 to 9, Platoon B consists of nodes 10 to 19, and

Δ	1	2	3	4	5
$\%G_{TC}$	-88.0739	-72.4476	-72.6310	-48.8758	-115.9446

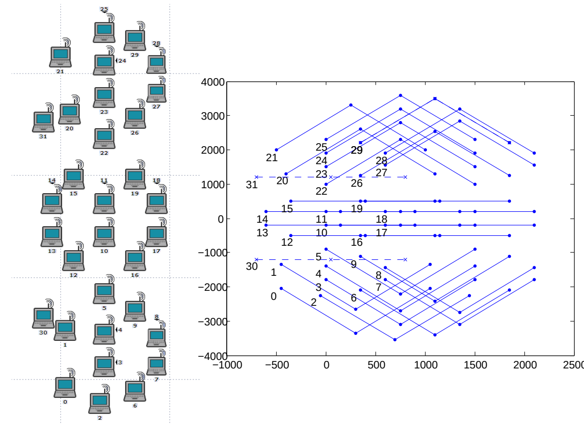
TABLE IV: TC traffic overhead for SPTC in Random Waypoint, $r_{TC}(OLSR) = 2.896kbps$

Type	Source-Destination	Offered Load (*kbps*)
Intra-Platoon	(1,3),(2,9),(4,6),(7,5),(20,29),(14,17),(16,11),(17,18),(19,12),(21,22),(23,27),(23,28)	12
Inter-Platoon	(1,18)	2.4
	(20,11),(20,0)	6
	(10,1),(21,10)	12

TABLE V: Traffic connections for Battlefield scenario

Δ	1	5	25	50	100
$\%G_{TC}$	32.7268	14.5622	-10.6280	-5.7759	-20.8613

TABLE VI: TC traffic overhead for SPTC in Battlefield scenario, $r_{TC}(OLSR) = 3.3857kbps$



(a) Initial topology (b) Node trajectories

Fig. 9: Battlefield Scenario

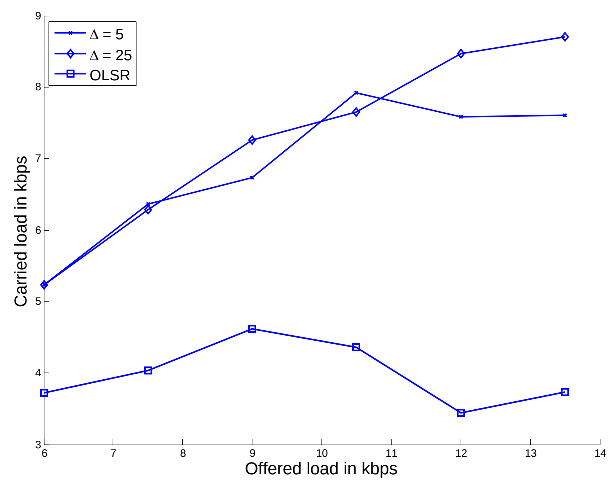


Fig. 10: Carried load vs. Offered load for the longest connection of Battlefield Scenario

Platoon C consists of nodes 20 to 29. The three platoons move in the trajectories shown in Subfigure 9b: Platoon B moves forward along the x direction, and Platoons A and C move forward and away from platoon B at speed of 1.5 m/s in the y direction. Then the platoons move together back to the initial formation. To ensure better connectivity among the platoons, two supporting nodes 30 (to support connections between Platoon A and B) and 31 (to support connections between Platoon B and C) move alongside the platoons (in the x direction). The simulations were carried out with the parameters shown in Table I. This yields a radio range of approximately 900m. Hence within each platoon, all the nodes are at most two hops from each other. When the platoons are close together, the inter-platoon communication is stable without using the supporting nodes 30 and 31. However, when the platoons move away from each other, the direct inter-platoon connections become unstable and the supporting stations become necessary for delivering high traffic. Again for SPTC, we chose the *current age* as link stability metric (this is only a heuristic).

UDP traffic was sent between 17 source-destinations pairs. Table V shows the base traffic for the scenario. For the traffic analysis, we focus on the connection $(20, 0)$ (from Platoon C to A) because this is a long connection and would be potentially sensitive to path stability. We scale the base traffic (offered load) of all connections (in Table V) by the same factor and obtain the carried load vs. offered load performance for connection $(20, 0)$ shown in Fig. 10. Again, we observe that SPTC carries significantly more load than OLSR for this connection. This is because when the platoons are maximally apart, we observe that for connection $(20, 0)$, SPTC routes significantly more traffic (about 1.5 times more) through the supporting nodes 30 and 31 when compared to OLSR's routing mechanism. We observe that the carried load for the other connections is also higher. Thus the overall network throughput is improved. For example, when the offered load to the network (all connections) is 410.5 *kbps*, SPTC is able to carry 386.8 *kbps*, while OLSR is able to carry 365.4 *kbps*.

The parameter Δ significantly influences the TC traffic. From Table VI, we see that for low Δs, the TC traffic rate is lower than in OLSR. However, for larger Δs, some unstable links (direct inter-platoon links) are also chosen as relay links, and these contribute to increased TC traffic overhead.

VII. Conclusion

In this paper, we introduced a new topology control problem for preserving stable routing paths. We formulated the problem as a constrained multi-agent optimization problem with only local neighborhood information. We developed localized perfect pruning methods that solve this problem. We also showed an extension of the perfect pruning to distorted pruning, which gives a systematic method to tradeoff path stability for shorter hop count paths. Finally, we quantified the two-fold advantage of stable path topology control mechanism with different simulation scenarios. We showed that the topology formed is stable and is able to carry significantly higher traffic compared to OLSR.

REFERENCES

[1] R. Rajaraman, "Topology control and routing in ad hoc networks: a survey," *ACM SIGACT News*, vol. 33, no. 2, pp. 60–73, 2002.

[2] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad-hoc networks," in *Proceedings of the ACM International Symposium on Mobile Ad-Hoc Networking and Computing (MOBIHOC)*, 2002, pp. 194–205.

[3] T. Clausen and P. Jacquet, "Optimized link state routing protocol (olsr)," RFC, Oct 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3626.txt

[4] R. Ogier, F. Templin, and M. Lewis, "Topology dissemination based on reverse-paht forwarding," RFC, February 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3684.txt

[5] S. Obilisetty, A. Jasti, and R. Pendse, "Link stability based enhancements to olsr (ls-olsr)," in *IEEE 62nd Vehicular Technology Conference*, 2005, pp. 306– 310.

[6] Z. Cheng and W. B. Heinzelman, "Exploring long lifetime routing (llr) in ad hoc networks," in *International workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems*, 2004, pp. 203–210.

[7] N. Sarma and S. Nandi, "Route stability based qos routing in mobile ad hoc networks," *Wireless Personal Communications*, 2009.

[8] K. Sridhar and M. Chan, "Stabilty and hop-count based approach for route computation in manet," in *International Conference on Computer Communications and Networks*, 2005, pp. 25–31.

[9] OPNETTechnologies, "Opnet modeler," URL. [Online]. Available: http://www.opnet.com/solutions/network_rd/modeler.html

[10] C. E. Perkins, *Ad hoc networking*. Addison Wesley, 2001.

[11] W. Su, S.-J. Lee, and M. Gerla, "Mobility prediction in wireless networks," in *MILCOM*, vol. 1, 2000, pp. 491–495.

[12] M. Gerharz, C. d. Waal, M. Frank, and P. Martini, "Link stability in mobile wireless ad hoc networks," in *Proceedings of the 27th Annual IEEE Conference on Local Computer Networks*, 2002, pp. 30–42.

[13] O. A. Al-Amoudi, M. S. El-Azhari, M. Woodward, and I. Awan, "Performance analysis of angle routing in manets," in *Lecture Notes in Computer Science*, 2008, pp. 21–29.

[14] D. S. J. D. Couto, D. Aguayo, B. A. Chambers, and R. Morris, "Performance of multihop wireless networks: Shortest path is not enough," *ACM SIGCOMM Computer Communication Review*, vol. 33, pp. 83–88, 2003.

[15] C.-K. Toh, "Associativity-based routing for ad hoc networks," *Wireless Personal Communications*, vol. 4, pp. 103–139, 1997.

[16] R. Dube, C. D. Rais, K.-Y. Wang, and S. K. Tripathi, "Signal stability-based adaptive routing (ssa) for ad hoc mobile networks," *IEEE Personal Communication*, vol. 4, no. 1, pp. 36–45, 1997.

[17] S. Agarwal, A. Ahuja, J. P. Singh, and R. Shorey, "Route-lifetime assessment based routing (rabr) protocol for mobile ad-hoc networks," in *IEEE International Conference on Communications*, vol. 3, 2000, pp. 1697–1701.

[18] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless Communications and Mobile Computing: Special Issues on Mobile Ad Hoc Networking: Research, Trends and Applications*, vol. 2, no. 5, pp. 483–502, 2002.

[19] N. Sadagopan, F. Bai, B. Krishnamachari, and A. Helmy, "Paths: analysis of path duration statistics and their impact on reactive manet routing protocols," in *ACM Internation Symposium on Mobile Ad Hoc Networking and Computing*, 2003, pp. 245–256.

[20] Y. Han, R. J. La, A. M. Makowski, and S. Lee, "Distribution of path durations in mobile ad hoc networks: Palm's theorem to the rescue," *The International Journal of Computer and Telecommunications Networking: Special Issue on Network Modeling and Simulation*, vol. 50, pp. 1887–1900, 2006.

[21] G. Lim, K. Shin, S. Lee, H. Yoon, and J. S. Ma, "Link stability and route lifetime in ad-hoc wireless networks," in *Proceedings of the International Conference on Parallel Processing Workshops*, 2002, p. 116.

[22] P. Yang and B. Huang, "Qos routing based on link stability with dynamic delay prediction in manet," in *IEEE Pacific-Asia Workshop on Computational Intelligence and Industrial Application*, vol. 1, 2008, pp. 515–518.

[23] D. Johnson and Y. Hu, "The dynamic source routing protocol (dsr) for mobile ad hoc networks for ipv4," RFC, Feb 2007. [Online]. Available: http://www.ietf.org/rfc/rfc4728.txt

[24] P. C., "Ad hoc on-demand distance vector (aodv) routing," RFC 3561, July 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3561.txt

[25] M. Ehrgott, *Multicriteria Optimization*. Springer, 2005.

[26] J. Wu and F. Dai, "A generic distirbuted broadcast scheme in ad hoc wireless networks," *IEEE Transactions of Computers*, vol. 53, no. 10, pp. 1343–1354, 2004.

[27] S. Lee, B. Bhattacharjee, A. Srinivasan, and S. Khuller, "Efficient and resilient backbones for multihop wireless networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 11, pp. 1349–1362, 2008.

[28] K. K. Somasundaram and J. S. Baras, "Semiring pruning for information dissemination in mobile ad hoc networks," in *Workshop on Applications of Graph Theory in Wireless Ad hoc Networks and Sensor Networks*, 2009.

[29] J. Moy, "Ospf version 2," RFC 2328. [Online]. Available: http://www.ietf.org/rfc/rfc2328.txt

[30] J. Kleinberg and É. Tardos, *Algorithm Design*. Addison Wesley, 2005.

[31] P. Jacquet, A. Laouiti, P. Minet, and L. Viennot, "Performance of multipoint relaying in ad hoc mobile routing protocols," in *NETWORKING 2002: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; and Mobile and Wireless Communications": Second International IFIP-TC6 Networking Conference, Pisa, Italy, May 19-24, 2002. Proceedings*. Springer, February 2002, pp. 387–398.

[32] J. S. Baras, S. Perumal, V. Tabatabaee, K. Somasundaram, and P. Purkayastha, "Loss network models and multiple metric performance sensitivity analysis for mobile wireless multi-hop networks," in *4th Wireless Internet Conference*, 2008.