Alternating Routes in Sensor Networks

Oliviu Ghica Goce Trajcevski Peter Scheuermann
Department of EECS
Northwestern University
Evanston, Il 60208
Email: {ocg474,goce,peters}@eecs.northwestern.edu
(Extended Abstract – Work in Progress)

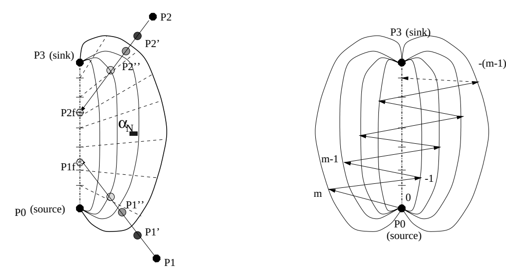
I. Introduction

In this paper, we present our ongoing research related to the problem of extending the lifetime of a given sensor network by alternating the packets' routing among a set of routes that ensure a bounded delay for the recipient. Firstly, we re-iterate over the issues that we have already addressed in our CAR project[5] and, subsequently, we present two topics that constitute the current focus of our work.

A. CAR: Controlled Adjustment of Routes and Sensor Networks Lifetime

This work addressed the problem of extending the lifetime of a sensor network in the settings in which there is a long-running query between a pair of *(sink, source)* nodes. Typically, when a given node (sink) requests continuous readings of the data-values sensed by another node (source), some sort of an *optimal* routing path is established between them [4]. However, due to the transmission, the energy-levels of the batteries in the nodes along the optimal route will drop faster than their neighboring nodes, which, in turn, will affect the network lifetime [2]. We introduced a methodology for constructing a set of *alternate* routes which consist of nodes in the neighborhood of the ones along the optimal route. By using them, the nodes along the optimal route are relieved, hence, the lifetime of the network is prolonged. The main contribution of the work is that a bounded delay on the packets is ensured along any alternate route and the construction of the set of routes can be achieved in a distributed manner.

To represent a particular route, we used Bezier curves. Given $n+1$ points: P_0, P_1, ..., P_n called *control points* and a parameter $u \in [0,1]$, the *Bezier curve* is defined as the set of all the points: $C_b(u) = \sum_{i=0}^{i=n} B_{i,n}(u) \cdot P_i$, where $u \in [0,1]$, and $B_{i,n}(u)$ are the *Bernstein polynomials*, defined as $B_{i,n}(u) = \begin{pmatrix} n \\ i \end{pmatrix} u^i(1-u)^{n-i}$ for $i \in \{0,1,\ldots,n\}$ and $B_{i,n}(u) = 0$ otherwise. Thus, a curve with $n+1$ control points is represented as a polynomial in u of n^{th} degree. Bezier curves have many interesting properties, and the most important ones in our context are: *Pseudo-local control* – moving one control point has predictable effects on the changes of the curve's shape and those effects are strongest around that point; and *Affine invariance* – in order to apply a particular affine transformation to the points on the curve (i.e., for all the values of u), it suffices to apply those transformations to the control points. Another important property is that a Bezier curve can be approximated with a poly-line [1]. Let $j \in \{1,2,\ldots,k\}$ and let $u_j = j\delta$, where $k\delta = 1$. Using the values of u_j's only, one approximate the rest of the curve (the other values of u) by straight line-segments that are at a bounded distance from the corresponding points on the original curve.



a.) Construction of Alternate Routes **b.)** Alternating Among Routes

Fig. 1. Bezier Curves and Alternate Routes

Two main aspects of the problem of the alternate routing are:
1. Routes Construction: Firstly, the sink needs to specify its tolerance for the time-delay of the packets, e.g., with respect to the delivery time along the optimal route [4], which will be used to determine the length of the Bezier curve corresponding to the alternate route furthest from the optimal one. Secondly, the sink needs to determine the control points (i.e., the shape) of the curve and the values of two parameters: 1. the discretization level for approximating the curve; and 2. the discretization level of degenerating the curve, in the sense that its control points will be brought from their initial positions to their final position along the optimal route in a given number of steps. This determines the curves which are the elements of the set of alternative routes; and given these parameters, each node can locally determine the route that it is closest to, as shown in Figure 1.a. for the node α_N.
2. Route Selection: Once the set of alternative routes have been constructed, the source needs a policy for selecting which particular route should be used for transmitting a given data-value. Assuming (c.f. Figure 1.b.) that the counter-clockwise direction is a positive one, the set of alternating routes can be enumerated as $S = [m,\ldots,1,0,-1,\ldots,-m]$, where "0" corresponds to the optimal route. The policy of alternating among the routes amounts to selecting the permutations of

S. For example, the source may randomly select a route; or use the original sequence S. However, this may incur packets collisions due to interference among the nodes along spatially close routes because, with a high sampling frequency, spatially close nodes will be employed for different routes in close time-values. To minimize this, we used the permutation for alternating among the routes which, during one period, corresponds to the following sequence: $m_d = [0, m, -1, (m-1), -2, (m-2), -3, \ldots]$. We call this the *constant mid-distance* alternating of routes. It is illustrated in Figure 1.b.) by the arrowed polyline, and our experiments [5] confirmed that indeed this policy yields largest lifetime-extensions. The experiments were conducted using our simulator[1] *SNSim*, which was built on top of JiST SWANS (http://jist.ece.cornell.edu)

II. CURRENT WORK

Now we describe two problems related to our CAR project, which present our ongoing work.

A. CAR with Energy-Map Awareness

The first extension of our work is to incorporate energy-aware decisions into the adaptive routing scheme. Our goal is to ensure that, given a continuous request between a *(sink, source)* pair, the distribution of the energy levels among the nodes is made and/or maintained as uniform as possible.

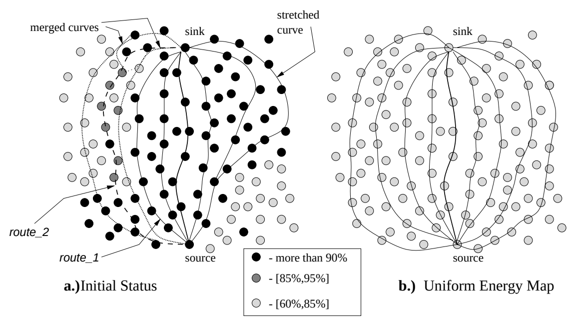


Fig. 2. Energy Awareness

When the sink generates a new request for readings from a particular source, there may be certain regions within the proximity of (or along) the optimal route that have high discrepancy in their energy levels, possibly even depleted. We would like to consider more general settings than simply avoiding holes [3]. As illustrated in Figure 2.a.), some of the curves corresponding to alternate routes may attempt to use nodes that are still alive, although their energy is bellow certain threshold. In such cases, the question becomes how to adjust the routes in a manner that will spare the nodes with low energy reserves. The left portion of Figure 2.a) illustrates the possibility in which two neighboring alternate routes (dotted curves) agree to merge themselves into one "new" route consisting of nodes that initially would not be part of any of them, illustrated with the thick dashed curve. The right portion of Figure 2.a.), on the other hand, illustrates the possibility of stretching the Bezier curve, e.g., by moving the control point, so that deformed alternative routes are constructed.

An important observation is that when there are several distinct intervals of available energy in the nodes (c.f. the legend in Figure 2), aside from the construction of alternate routes, the question arises of *what* is the optimal policy that the source should use for alternating among them. For example, it may be the case that *route_1* in Figure 2.a.) should be used twice as often as *route_2*. Our current goal is to ensure that after running the continuous query for some time, the energy levels of the nodes in the neighborhood of the optimal route become as uniform as possible, as illustrated in Figure 2.b.). Subsequently, the alternating among routes can proceed in a manner which corresponds to the original CAR-like settings.

B. CAR in Multi-Query Settings

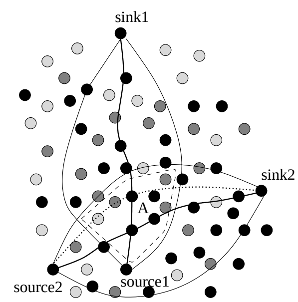


Fig. 3. Multiple Queries

The other extension of our work is concerned with the problem of prolonging the network lifetime from another perspective. While still attempting to alternate among more than one possible route, we also consider the settings in which more than one query-request can be posed.

An instance of the problem is illustrated in Figure 3. Assume that the pair *(sink1,source1)* has started the monitoring process in a CAR-like manner at t_1. Later on, at time $t_2 > t_1$, another *(sink2,source2)* pair initiates a continuous query. Now we have an area in which certain nodes have multiple task-identities. For example, the node A is along the optimal route for *(sink2,source2)*, however, it is also along an alternate route for *(sink1,source1)*. Clearly, one possible remedy is to allow extra-sets of wider curves for the alternate routes, however, it comes at the expense of longer time-delays plus the run-time cost for constructing the extra-sets. Currently, we are working on route selection policies that prolong the lifetime of the network in such multi-query settings.

REFERENCES

[1] H. Alt, E. Welzl, and B. Wolfers. Piecewise linear approximation of bezier curves. In *Symp. Computational Geometry*, 1997.
[2] Q. Dong. Maximizing system lifetime in wireless sensor networks. In *IPSN*, 2005.
[3] Q. Fang, J. Liu, and L. Guibas. Locating and bypassing routing holes in sensor networks. In *INFOCOM*, 2004.
[4] G. Karp and H. T. Kung. Gpsr: Greedy perimeter stateless routing for wireless sensor networks. In *MOBICOM*, 2000.
[5] G. Trajcevski, O. Ghica, and P. Scheuermann. Car: Controlled adjustement of routes and sensor networks lifetime. In *MDM*, 2006.

[1] Available at *www.ece.northwestern.edu/peters/˜peters/sensors.*