Update. Given a set of points P in $\mathbb{R}^3$ contained within a slab of thickness $1/\sqrt{2}$, the projection of $UBG(P)$ onto the plane parallel to the slab gives a $(1/\sqrt{2})$-quasi unit disc graph, for which a distributed, memoryless routing algorithm is given by Barrière et al. [1]. Furthermore, Kuhn et al. [2] propose the same algorithm as that described below for the projection of $UBG(P)$. As such, our routing algorithm was preceeded by these two solutions to the same problem which have similar running times to ours.

Problem statement. Let P denote a set of points in $\mathbb{R}^3$ such that P is contained within some slab of thickness λ. Let $USG(P)$ denote the unit sphere graph of P. That is, points p_1 and p_2 in P are adjacent in $USG(P)$ iff $||p_1 - p_2|| \leq 1$. For a given source-target pair (s, t) in P, we describe a distributed and memoryless algorithm that guarantees delivery of a message from s to t. The algorithm is based on the two-dimensional face routing algorithm of Bose, Morin, Stojmenovic, and Urrutia (Wireless Networks, 2001) in unit disc graphs.

Observations and progress. Without loss of generality, assume the slab is parallel with the xy-plane such that the z-coordinate of each point in P is in the range $[0, \lambda]$. Let $e_1 = (u_1, v_1)$ and $e_2 = (u_2, v_2)$ be two edges in $USG(P)$ such that e_1 and e_2 cross in the projection of $USG(P)$ onto the xy-plane. The minimum distance between endpoints of e_1 and e_2 is maximized when both edges have unit length, the relative position of the edges is perpendicular in the xy-plane, each endpoint is a distance $1/2$ from the edge crossing in the projection, and the edges are parallel to opposite planes on the boundary of the slab. Without loss of generality, assume $u_1 = (0, 0, 0)$, $v_1 = (1, 0, 0)$, $u_2 = (1/2, 1/2, \lambda)$, and $v_2 = (1/2, -1/2, \lambda)$. The minimum distance between endpoints of e_1 and e_2 is $\sqrt{\lambda^2 + 1/2}$. Therefore, for any e_1 and e_2, some endpoint of e_1 must be adjacent to some endpoint of e_2 in $USG(P)$ if $\lambda \leq 1/\sqrt{2}$. If $\lambda > 1/\sqrt{2}$, then it is possible for e_1 and e_2 to cross in the projection of $USG(P)$ even if no edge exists between any endpoints of e_1 and any endpoints of e_2 in $USG(P)$. Therefore, assume $\lambda \leq 1/\sqrt{2}$. This ensures that edge crossings in the projection of $USG(P)$ can be detected locally. In particular, given an edge e_1 we can enumerate all edges that cross e_1 in the projection by considering the set of edges adjacent to neighbours of either endpoint of e_1.

We now describe the routing algorithm. Consider the projection of $USG(P)$ onto the xy-plane. The initial direction in which to send the message is given by the face routing algorithm. Each step in the route is determined as follows. Say the message is to be passed from node u_1 to node v_1. Let u' denote an intermediate point along the line segment u_1v_1 that was determined in the last step of computation (u' may be equal to u_1). Assume the face routing algorithm is presently following a "left hand" rule (the face is being traversed in a clockwise diretion). By examining the set of edges adjacent to neighbours of u_1 and v_1, determine the edge that crosses line segment $u'v_1$ for which the edge crossing is nearest to u'. Denote this edge by (u_2, v_2) such that v_2 lies inside the current face. The message is passed to v_1 and then to u_2. The algorithm repeats to pass the message from node u_2 to v_2 with the new value of u' set to the intersection of (u_1, v_1) and (u_2, v_2). If no edge crosses line segment $u'v_1$, then the first anticlockwise edge from v_1 is followed and the algorithm repeats with $u' = v_1$. When considering the segment $u'v_1$, the algorithm observes the same rules as the 2D face routing algorithm regarding crossing the line segment st. Unlike the 2D algorithm which operates on the subset of edges given by the Gabriel graph of P (which can be calculated locally), our algorithm considers all edges of $USG(P)$ and resolves edge crossings by creating a virtual vertex (u') at every edge crossing. In 3D, the Gabriel graph is not sufficient to eliminate all edge crossings in the projection.

Open questions. Can we bound the stretch factor? That is what is the worst-case ratio between the length of route taken by the message between s and t as compared to the length of the shortest path between s and t, where both path lengths are measured by total number of edges.

Can any memoryless distributed routing algorithm guarantee delivery when $\lambda > 1/\sqrt{2}$?

References

[1] Lali Barrière, Pierre Fraigniaud, and Lata Narayanan. Robust position-based routing in wireless ad hoc networks with unstable transmission ranges. In *Proceedings of the International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*, pages 19–27, 2001.

[2] Fabian Kuhn, Roger Wattenhofer, and Aaron Zollinger. Ad-hoc networks beyond unit disk graphs. In *Joint Workshop on Foundations of Mobile Computing*, pages 69–78, 2003.