Journal of Interconnection Networks

BALANCING TRAFFIC LOAD USING ONE-TURN RECTILINEAR ROUTING*

STEPHANE DUROCHER
Department of Computer Science, University of Manitoba, Winnipeg, Manitoba, Canada
durocher@cs.umanitoba.ca

EVANGELOS KRANAKIS
School of Computer Science, Carleton University, Ottawa, Ontario, Canada
kranakis@scs.carleton.ca

DANNY KRIZANC
Department of Mathematics and Computer Science, Wesleyan University, Middletown, Connecticut, USA
dkrizanc@wesleyan.edu

LATA NARAYANAN
Department of Computer Science, Concordia University, Montréal, Québec, Canada
lata@cse.concordia.ca

Received Day Month Year
Revised Day Month Year

We consider the problem of load-balanced routing, where a dense network is modelled by a continuous square region, and origin-destination node pairs correspond to pairs of points in that region. The objective is to define a routing policy that assigns a continuous path to each origin-destination pair while minimizing the traffic, or *load*, passing through any single point. While the average load is minimized by straight-line routing, such a routing policy distributes the load non-uniformly, resulting in higher load near the center of the region. We consider one-turn rectilinear routing policies that divert traffic away from regions of heavier load, resulting in up to a 33% reduction in the maximum load while simultaneously increasing the path lengths by an average of less than 28%. Our policies are simple to implement, being both local and oblivious. We provide a lower bound that shows that no one-turn rectilinear routing policy can reduce the maximum load by more than 39% and we give a polynomial-time procedure for approximating the optimal randomized policy.

Keywords: dense wireless networks; routing; load-balanced; congestion; rectilinear; stretch factor.

*Some of these results appeared in preliminary form at the Conference on Theory and Applications of Models of Computation.[7] Work of the first and fourth authors was supported in part by NSERC. Work of the second author was supported in part by NSERC and MITACS.

1. Introduction

1.1. *Motivation*

The problem of routing in multi-hop wireless networks has received extensive attention in the last decade[2,17,20,24,29]. Many of the proposed routing protocols attempt to find shortest paths between pairs of nodes, or seek to bound the *stretch factor* of the paths, while trying to ensure that the paths are loop-free. These approaches consider individual packets traversing the network and attempt to optimize performance for a single packet. Obtaining a more global and, in many cases, more realistic perspective on the performance of a routing protocol requires considering many simultaneous traffic flows in the network. In this case, *congestion* occurs when several packets need to be forwarded by a common intermediate node at the same time. This congestion is often proportional to the latency experienced by a packet. Therefore, a routing protocol should attempt to avoid creating highly-congested nodes. Not only does such a protocol improve packet latency, it also improves the lifetime of a wireless network, where heavily loaded nodes may run out of battery power and disconnect the network.

In this paper, we investigate routing protocols for wireless networks with the aim of minimizing the congestion experienced at nodes. We consider a multi-hop ad hoc network consisting of identical location-aware nodes, uniformly and densely deployed within a given planar region. Several papers in the literature (e.g., Ref. 30) refer to such networks as *massively wireless*. Furthermore, we assume that the traffic pattern is uniform point-to-point communication, i.e., each node has the same number of packets to send to every other node in the network. This is sometimes called the *all-to-all* communication pattern. A routing policy must define, for every ordered pair of nodes (u, v), a path in the network to get from u to v. The *load* at a given node v is the number of paths that pass through v. The average (maximum) load for a network for a given routing policy is the average (respectively, maximum) load over all nodes in the network. The fundamental question we wish to answer is: *what routing policy minimizes the maximum load in the network?*

Shortest-path routing minimizes the average load in a convex planar region (see Section 2.2). However, this same routing strategy results in non-uniform load distribution and, furthermore, causes high load near the region's geometric center; several studies confirm the crowded center effect for shortest-path routing on various convex planar regions[10,13,14,15,16,21,25,26]. This suggests that if load balancing is a primary objective, then a good routing policy should redirect some of the traffic away from the geometric center and other areas of high load. However, load balancing cannot be the only concern: taking unnecessarily long paths just to bypass the center can drastically increase the stretch factor and the average load of nodes in the network and, consequently, can result in inefficient energy consumption. Furthermore, it is critical that the forwarding strategy required to implement the routing policy be simple and have low memory requirements. Ideally, the routing policy should be *oblivious* (the route between u and v depends only on the identities or locations of u and v) and the forwarding strategy should be *local* (the forwarding node can make its decision based only on knowledge of itself, its neighbors, and the packet header, which contains only the address of the destination).

The problem of load-balanced routing when nodes are uniformly distributed in a given convex planar region has been considered when the region is a unit disk[13,14,15,16,25,26] and, recently, a unit square[16]. However, little research has been done on finding a simple (easily implementable in a scalable distributed fashion) routing policy that succeeds at both minimizing the maximum load and achieving a reasonable stretch factor.

In this paper, we investigate the problem of load-balanced routing when the nodes are uniformly and densely packed in a square or rectangular region. As in Refs. 16 and 26, our approach is to model nodes in the region by a *continuous space* rather than by a discrete set of points. This permits analysis of the average and maximum load induced by a routing policy without regard to the topology of the actual network. At the same time, the results should predict the behavior of a network with very densely and uniformly deployed nodes. Shortest-path routing corresponds to *straight-line routing* in this setting. We derive the average and maximum load for straight-line routing in a unit square and confirm the crowded center effect for squares and rectangles. In keeping with the goal of minimizing load while ensuring a reasonable stretch factor, we investigate the class of *one-turn rectilinear* routing policies that assign to each origin-destination pair of nodes one of the two possible rectilinear paths containing only a single intermediate point. In particular, we consider one-turn rectilinear routing policies that are simple and realistic in the ad hoc network setting; the routing policy is oblivious and the forwarding algorithm is local. We propose and analyze several one-turn rectilinear strategies, the best of which reduces the maximum load by about 33% compared to the straight-line policy. We also characterize the optimal randomized rectilinear policy as the solution to an optimization problem and provide an efficient procedure for approximating it.

1.2. *Overview of Results*

Our main contributions are summarized below:

- We derive an exact expression for the load induced by the straight-line routing policy at an arbitrary point in the unit square. We show that the average and maximum load for the straight-line routing policy are 0.5214 and 1.1478 respectively.
- We show that the average load for every one-turn rectilinear routing policy is 2/3. The maximum and average stretch factor for such policies are shown to be $\sqrt{2}$ and 1.2737 respectively.
- We propose several one-turn rectilinear routing policies and derive their maximum load. The best of these, called the *diagonal rectilinear policy*, achieves a

maximum load of 0.7771, which represents a 33% improvement over straight-line routing. Furthermore, this policy can easily be implemented in a local and oblivious distributed routing environment.

- We prove a lower bound of 0.7076 on the the maximum load for any one-turn rectilinear policy.
- We characterize the optimal randomized one-turn rectilinear policy as the solution to an optimization problem and provide an efficient procedure for approximating it. Numerical results suggest that the maximum load for the best possible one-turn rectilinear policy is close to 0.73.
- We generalize our results to derive the maximum and average loads for straight-line as well as diagonal rectilinear routing policies on $k \times 1$ rectangles.

2. Definitions

2.1. *Routing Policies and Traffic Load*

Given a convex region $A \subseteq \mathbb{R}^2$, a *routing policy* P assigns a route to every origin-destination pair $(u, v) \in A^2$, where the *route* from u to v, denoted $\text{route}_P(u, v)$, is a plane curve segment contained in A, whose endpoints are u and v. For a given routing policy P on a region A, the traffic load at a point p is proportional to the number of routes that pass through p. Formally,

Definition 2.1. Given a routing policy P on a region A, the *load* at point p is

$$\lambda_P(p) = \iint_A f_P(p, u, v)\, du\, dv, \qquad \text{where } f_P(p, u, v) = \begin{cases} 1 \text{ if } p \in \text{route}_P(u, v), \\ 0 \text{ otherwise.} \end{cases}$$

The *average load* of routing policy P on region A is given by

$$\lambda_{\text{avg}}(P) = \frac{1}{\text{Area}(A)} \int_A \lambda_P(p)\, dp, \tag{2.2}$$

where $\text{Area}(A) = \int_A dp$ denotes the area of region A. The average length of a route determined by policy P between two points in A is given by

$$\text{length}_{\text{avg}}(P) = \frac{1}{\text{Area}(A)^2} \iint_A \text{length}(\text{route}_P(p, q))\, dq\, dp. \tag{2.3}$$

Since $\text{length}(\text{route}_P(u, v)) = \int_A f_P(p, u, v)\, dp$, Proposition 2.1 follows from (2.2) and (2.3):

Proposition 2.1. *Given routing policy P on a region A,*

$$\lambda_{\text{avg}}(P) = \text{Area}(A) \cdot \text{length}_{\text{avg}}(P). \tag{2.4}$$

In addition to average load, a routing policy P on a region A is also characterized by its *maximum load*, given by

$$\lambda_{\max}(P) = \max_{p \in A} \lambda_P(p). \tag{2.5}$$

Thus, a primary objective of this work is to identify routing policies P that minimize (2.5).

2.2. *Straight-Line Routing Policy*

The *straight-line routing policy*, denoted S, assigns to every pair (u, v) the route consisting of the line segment between u and v. In straight-line routing,

$$\text{length}(\text{route}_S(p, q)) = ||p - q|| = \sqrt{(p_x - q_x)^2 + (p_y - q_y)^2}. \tag{2.6}$$

Since the line segment from u to v is the shortest route from u to v, it follows that straight-line routing minimizes (2.3). Consequently, for any convex region A and any routing policy $P \neq S$,

$$\lambda_{\text{avg}}(S) \leq \lambda_{\text{avg}}(P). \tag{2.7}$$

The *average stretch factor* and *maximum stretch factor* of routing policy P on region A are respectively given by

$$\text{str}_{\text{avg}}(P) = \frac{1}{\text{Area}(A)^2} \iint_A \frac{\text{length}(\text{route}_P(p, q))}{\text{length}(\text{route}_S(p, q))}\, dq\, dp \tag{2.8}$$

$$\text{and } \text{str}_{\text{max}}(P) = \max_{\{p,q\} \subseteq A} \frac{\text{length}(\text{route}_P(p, q))}{\text{length}(\text{route}_S(p, q))}. \tag{2.9}$$

2.3. *One-Turn Rectilinear Routing Policies*

In this paper, we consider the case when region A is bounded by a square or a rectangle. As we show in Sections 4.3 and 8.1, the load in straight-line routing on a square or a rectangle is maximized at its center. The maximum load can be decreased by redirecting routes that pass near the center to areas of lower load. This motivates the examination of one-turn rectilinear routing policies which we now define.

A *monotonic rectilinear routing policy* assigns to every pair (u, v) a route consisting of a monotonic rectilinear path from u to v, i.e., a path comprised of a series of axis-parallel line segments such that any axis-parallel line intersects the path at most once. A *one-turn rectilinear routing policy* assigns to every pair (u, v) a monotonic rectilinear path consisting of one horizontal line segment and one vertical line segment joining u to v via an intermediate point w. Point w may coincide with u or v.

For any monotonic rectilinear routing policy P,

$$\text{length}(\text{route}_P(p, q)) = |p_x - q_x| + |p_y - q_y|. \tag{2.10}$$

In general, there are two possible one-turn rectilinear routes from a given origin (u_x, u_y) to a given destination (v_x, v_y). We refer to these as *row-first* and *column-first*, where the row-first route passes through the intermediate point (v_x, u_y) and the column-first route passes through the intermediate point (u_x, u_y).

3. Related Work

In this section, we briefly describe other efforts to address the load-balancing problem. See Toumpis[30] for a survey of results related to physically-based wireless networks, including a section on load-balanced routing in ad hoc wireless networks.

Experimental Approaches. One body of work on congestion detection and control in wireless networks is experimental and takes a cross-layer approach, where transport and MAC layers are involved. Routes are changed reactively (i.e., they are non-oblivious), and an all-to-all communication pattern is not assumed. We mention two of the papers in this area to give a flavor of the techniques used. Hull et al.[12] look at mitigating the congestion using mechanisms such as hop-by-hop flow control, source-rate limiting schemes, and a prioritized MAC layer. Wan et al.[31] propose an approach in which nodes monitor the channel to detect congestion, in the event of which they broadcast backpressure messages upstream towards the source, which in turn can reduce its sending rate. In addition, a closed-loop multi-source regulation mechanism is employed.

Network Layer Approaches. Another class of strategies to reduce maximum load consists of using multiple paths to distribute routing load. Pham and Perreau[25] propose the use of multiple paths to mitigate the crowded center effect. They analyze and compare reactive single-path and multi-path routing with load-balance mechanisms in ad hoc networks, in terms of overhead, traffic distribution and connection throughput. RMRP is an AODV-based multi-path routing protocol in which sensor nodes set up double routing paths toward sink nodes in an AODV fashion and select one of them randomly for forwarding a packet[18]. However, Ganjali and Keshavarzian[7] suggest that the load distribution in multi-path routing would be almost identical to that in single-path routing unless an infeasibly large number of paths is used.

Load versus Stretch Factor. The tradeoff between load and stretch factor in wireless networks has been studied by Meyer et al.[23] and Gao and Zhang[9]. For growth-bounded wireless networks, Gao and Zhang[9] describe routing policies that achieve a stretch factor of c and a load-balancing ratio of $O((n/c)^{1-1/k})$, where k denotes the growth rate and the ratio measures maximum load for a given routing policy relative to the optimal policy. They also derive a routing policy for unit disk graphs with bounded density and show that if the density is constant, shortest-path routing has a load-balancing ratio of $\Theta(\sqrt{n})$. The communication patterns considered are arbitrary, the lower bound does not derive from the all-to-all communication pattern, and the routing policies are not oblivious. Gao and Zhang[8] give a routing policy that achieves a good tradeoff between stretch factor and load balance for the special case when all nodes are located in a narrow strip of width at most $\sqrt{3}/2$ times the transmission radius. Their analysis is not specific to the all-to-all communication pattern.

Graphs Network Models. The all-to-all communication pattern has been studied extensively in the context of interconnection networks, and particularly in WDM optical networks. In this context, Chung et al.[4] define the forwarding index of a communication network with respect to a specific routing policy to be the maximum number of paths going through any vertex in the graph. The forwarding index of the network itself is the minimum over all possible routing policies for the network. This notion was extended to the maximum load on an edge[11], which is more appropriate to wired networks. For wireless networks, however, the node forwarding index is a better representative of the load on a wireless node. While the node forwarding index for specific networks, including ring and torus networks, has been derived exactly[4], it has not been studied for two-dimensional grid networks, which would perhaps be a good approximation for the dense wireless networks of interest to us. Our results in Section 7 provide an approximation for the forwarding index in grid graphs for the class of one-turn rectilinear routing schemes.

Planar Region Network Models. Few results address the problem of reducing load in all-to-all communication on networks contained in a continuous region in the plane – the model of interest in this paper. The average load for straight-line routing in a continuous convex region is proportional to the expected distance between two points selected at random in that region (see Proposition 2.1); Bailey et al.[1], Dunbar[5], and Santaló[28] evaluate this distance for various convex regions including squares, rectangles, and disks. Hyytiä et al.[13], Hyytiä and Virtamo[14,15,16], Pham and Perreau[25], and Popa et al.[26] analytically derive the load of straight-line routing at an arbitrary point in a disk on the plane; as in this paper, the network is modelled as a continuous region rather than as a discrete set of nodes. In a recent result, Hyytiä and Virtamo[16] consider load balancing on the unit disk and the unit square in terms of vector flow fields. Although they derive a flow on the unit square whose maximum load is theoretically lower than that of our diagonal rectilinear policy, there is no obvious way to implement this strategy. In particular, Hyytiä and Virtamo[16] state that "future work includes developing efficient algorithms to find optimal paths in [a] distributed and scalable fashion".

Some recent papers propose implementable routing stretegies with the aim of reducing maximum congestion; exact bounds for the average or maximum load induced by their strategies are not given. For example, Busch et al.[3] analyze load-balanced routing on graphs embedded in the plane via an intermediate node selected at random near the perpendicular bisector of the origin and destination; although their result is on discrete networks, the strategy described by Busch et al. is geometric and can be generalized to the continuous setting (see Section 9). No non-trivial bounds on load have been provided for this strategy. Popa et al.[26] propose a routing policy called curveball routing and present experimental results comparing straight-line routing to curveball routing in disk-, square- and rectangular-shaped regions, providing evidence that curveball routing achieves a reduction in maximum load in such regions. No theoretical bounds are given on either the stretch factor or the maximum load in the network. At present, therefore, our diagonal rectilinear routing policy remains the best simple (easily-implementable in a scalable distributed fashion) path selection strategy with a provable upper bound on maximum load.

4. Straight-Line Routing on a Square

In this section we examine the load of straight-line routing on the unit square and derive formulas for the load at an arbitrary point p, the maximum load, and the average load.[a] These values serve as benchmarks against which the optimality of all other routing policies on the unit square are compared.

4.1. *Average Load*

By Proposition 2.1, the average load in the unit square under straight-line routing is equal to the expected distance between two points selected at random in the square. This value is a box integral with the following solution[1,5,28]:

$$\begin{aligned}
\text{length}_{\text{avg}}(S) &= \int_0^1\!\!\int_0^1\!\!\int_0^1\!\!\int_0^1 \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2}\, dv_y\, dv_x\, du_y\, du_x \\
&= \frac{2 + \sqrt{2} + 5\ln(1 + \sqrt{2})}{15} \\
&\approx 0.5214. \qquad (4.1)
\end{aligned}$$

By (2.7), the average load (and maximum load) of any routing policy on the unit square is bounded from below by (4.1).

4.2. *Load at an Arbitrary Point*

Since straight-line routing is symmetric in the x- and y-dimensions, we derive the load at an arbitrary point p located in the first octant of the unit square. The load at an arbitrary point in the unit square is then easily found using the appropriate coordinate transformation.

Theorem 4.1. Given a point $p = (p_x, p_y)$ such that $1/2 \leq p_y \leq p_x \leq 1$, the load at p using straight-line routing is given by (4.4).

Proof. A route passes through point p if and only if point p lies on the line segment between the origin u and the destination v. Thus, the load at p is calculated

[a] After these results appeared in conference form[6] and shortly before submission of this manuscript, the authors learned that the result of Theorem 4.2 was recently independently derived by Hyytiä and Virtamo[16] using methods involving vector flow fields and will appear in the 2008 Conference on Next Generation Internet Networks. We include our independent proof of Theorem 4.2 for completeness.

by integrating over all lines through p, all origin points u on the line segment between p and the boundary of the square, and all destination points v on the line segment between p and the opposite boundary of the square, where the integrand is appropriately weighted such that points are distributed uniformly in the square.

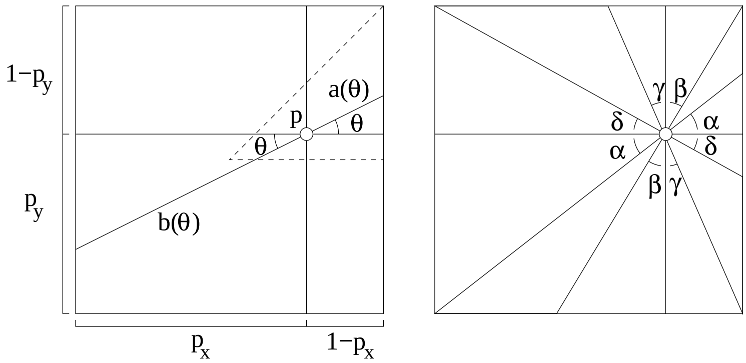

Fig. 1. Deriving the load at a point p by straight-line routing

Let l_θ denote the line through p parallel to the vector $(\cos\theta, \sin\theta)$. Let $a(\theta)$ and $b(\theta)$ denote the lengths of the corresponding line segments of l_θ contained in $[0,1]^2$ that lie above and below p, respectively. The values of $a(\theta)$ and $b(\theta)$ can be expressed in terms of θ, p_x, and p_y by considering the different cases as θ increases and l_θ intersects different edges of the square's boundary. As illustrated in Figure 1, let $\alpha = \tan^{-1}(p_y/p_x)$, $\beta = \tan^{-1}((1-p_x)/(1-p_y))$, $\gamma = \tan^{-1}((1-p_x)/p_y)$, and $\delta = \tan^{-1}((1-p_y)/p_x)$. This divides the interval $[0,\pi]$ into six intervals: $[0,\alpha]$, $[\alpha, \pi/2-\beta]$, $[\pi/2-\beta, \pi/2]$, $[\pi/2, \pi/2+\gamma]$, $[\pi/2+\gamma, \pi-\delta]$, and $[\pi-\delta, \pi]$. The corresponding values for $a(\theta)$ and $b(\theta)$ are given in Table 1. Let $\lambda_S^\theta(p)$ denote the contribution to load at p for routes that lie on l_θ. Since θ rotates about p, origins and destinations must be weighted by their distance from p. As is done in Refs. 13 and 15, this weighting gives

$$\lambda_S^\theta(p) = a(\theta)b(\theta)[a(\theta)+b(\theta)]. \tag{4.2}$$

Table 1. Values of $a(\theta)$, $b(\theta)$, and $a(\theta)b(\theta)[a(\theta)+b(\theta)]$

range of θ	$a(\theta)$	$b(\theta)$	$a(\theta)b(\theta)[a(\theta)+b(\theta)]$
$[0,\alpha]$	$(1-p_x)\sec\theta$	$p_x\sec\theta$	$(1-p_x)p_x \cdot \sec^3\theta$
$[\alpha,\pi/2-\beta]$	$(1-p_x)\sec\theta$	$p_y\csc\theta$	$(1-p_x)^2 p_y \cdot \sec^2\theta\csc\theta$ $+(1-p_x)p_y^2 \cdot \sec\theta\csc^2\theta$
$[\pi/2-\beta,\pi/2]$	$(1-p_y)\csc\theta$	$p_y\csc\theta$	$(1-p_y)p_y \cdot \csc^3\theta$
$[\pi/2,\pi/2+\gamma]$	$(1-p_y)\csc\theta$	$p_y\csc\theta$	$(1-p_y)p_y \cdot \csc^3\theta$
$[\pi/2+\gamma,\pi-\delta]$	$(1-p_y)\csc\theta$	$-(1-p_x)\sec\theta$	$(1-p_y)(1-p_x)^2 \cdot \csc\theta\sec^2\theta$ $-(1-p_y)^2(1-p_x) \cdot \csc^2\theta\sec\theta$
$[\pi-\delta,\pi]$	$-p_x\sec\theta$	$-(1-p_x)\sec\theta$	$-p_x(1-p_x) \cdot \sec^3\theta$

For all θ, (4.2) can be decomposed into functions $f_{i_1}(p_x, p_y) \cdot g_{j_1}(\theta)$ or $f_{i_1}(p_x, p_y) \cdot g_{j_1}(\theta) + f_{i_2}(p_x, p_y) \cdot g_{j_2}(\theta)$ such that f_{i_k} is independent of θ and g_{j_k} is independent of p_x and p_y for $k \in \{1, 2\}$. Function f_{i_k} corresponds to one of seven functions that are quadratic in p_x or p_y, whereas g_{j_k} corresponds to one of four functions that are cubic in $\sec\theta$ or $\csc\theta$. See Table 1. Below are indefinite integrals with respect to θ for each of the four possible functions g_{j_k}:

$$
\begin{aligned}
\text{Let } h_1(\theta) &= \int \sec^3\theta\, d\theta = \begin{cases} \frac{1}{2}\left[\tan\theta\sec\theta + \ln(\sec\theta + \tan\theta)\right], & \text{if } \theta \in [0, \pi/2], \\ \frac{1}{2}\left[\tan\theta\sec\theta + \ln(-\sec\theta - \tan\theta)\right], & \text{if } \theta \in [\pi/2, \pi], \end{cases} \\
h_2(\theta) &= \int \csc^3\theta\, d\theta = \frac{1}{2}\left[-\cot\theta\csc\theta + \ln(\csc\theta - \cot\theta)\right], \\
h_3(\theta) &= \int \csc^2\theta\sec\theta\, d\theta = \begin{cases} -\csc\theta + \ln(\sec\theta + \tan\theta), & \text{if } \theta \in [0, \pi/2], \\ -\csc\theta - \ln(-\sec\theta + \tan\theta), & \text{if } \theta \in [\pi/2, \pi], \end{cases} \\
h_4(\theta) &= \int \csc\theta\sec^2\theta\, d\theta = \sec\theta + \ln(\csc\theta - \cot\theta).
\end{aligned} \tag{4.3}
$$

The load at p is given by

$$
\begin{aligned}
\lambda_S(p) &= \int_0^\pi \lambda_S^\theta(p)\, d\theta \\
&= \int_0^\pi a(\theta)b(\theta)[a(\theta) + b(\theta)]\, d\theta, \\
&= \int_{\theta=0}^{\alpha} (1-p_x)p_x \sec^3\theta\, d\theta + \int_{\theta=\alpha}^{\pi/2-\beta} (1-p_x)^2 p_y \csc\theta\sec^2\theta + (1-p_x)p_y^2 \csc^2\theta\sec\theta\, d\theta \\
&+ \int_{\theta=\pi/2-\beta}^{\pi/2} (1-p_y)p_y \csc^3\theta\, d\theta + \int_{\theta=\pi/2}^{\pi/2+\gamma} (1-p_y)p_y \csc^3\theta\, d\theta \\
&+ \int_{\theta=\pi/2+\gamma}^{\pi-\delta} (1-p_y)(1-p_x)^2 \csc\theta\sec^2\theta - (1-p_y)^2(1-p_x)\csc^2\theta\sec\theta\, d\theta \\
&+ \int_{\theta=\pi-\delta}^{\pi} -p_x(1-p_x)\sec^3\theta\, d\theta \\
&= (1-p_x)p_x \left[\Big|_{\theta=0}^{\alpha} h_1(\theta)\right] + (1-p_x)^2 p_y \left[\Big|_{\theta=\alpha}^{\pi/2-\beta} h_4(\theta)\right] \\
&+ (1-p_x)p_y^2 \left[\Big|_{\theta=\alpha}^{\pi/2-\beta} h_3(\theta)\right] + (1-p_y)p_y \left[\Big|_{\theta=\pi/2-\beta}^{\pi/2} h_2(\theta)\right] \\
&+ (1-p_y)p_y \left[\Big|_{\theta=\pi/2}^{\pi/2+\gamma} h_2(\theta)\right] + (1-p_y)(1-p_x)^2 \left[\Big|_{\theta=\pi/2+\gamma}^{\pi-\delta} h_4(\theta)\right] \\
&- (1-p_y)^2(1-p_x) \left[\Big|_{\theta=\pi/2+\gamma}^{\pi-\delta} h_3(\theta)\right] - p_x(1-p_x) \left[\Big|_{\theta=\pi-\delta}^{\pi} h_1(\theta)\right].
\end{aligned} \tag{4.4}
$$

$\square$

Expression (4.4) has a closed-form polylogarithmic representation (free of any trigonometric terms). The complete expression is not reproduced here due to the large number of terms but can be easily reconstructed from (4.3) and (4.4). Figure 8 displays a plot of (4.4) for $p \in [0, 1]^2$.

4.3. *Maximum Load*

We now derive the maximum load for straight-line routing on the unit square and show that this value is realized at the center of the square.

Theorem 4.2. The maximum load for straight-line routing on the unit square is

$$\lambda_{\max}(S) = \frac{1}{\sqrt{2}} + \frac{3}{8}\ln(\sqrt{2}+1) - \frac{1}{8}\ln(\sqrt{2}-1) \approx 1.1478, \tag{4.5}$$

realized uniquely at the center of the square.

Proof. Choose any point $p = (p_x, p_y)$ in the unit square $[0,1]^2$.

Case 1. Suppose p lies in the first octant of the square. That is, $1/2 \leq p_y \leq p_x \leq 1$.

Case 1a. Suppose $p_x \neq p_y$. We show that if p_x decreases (i.e., p moves left toward the diagonal $p_x = p_y$) then the load increases. It suffices to show that $a(\theta)b(\theta)[a(\theta)+b(\theta)]$ increases for all θ, where $a(\theta)$ and $b(\theta)$ are defined as in (4.2). Choose any $\theta \in [0,\pi]$. Consider again the six intervals $\theta \in [0,\alpha]$, $[\alpha, \pi/2-\beta]$, $[\pi/2-\beta, \pi/2]$, $[\pi/2, \pi/2+\gamma]$, $[\pi/2+\gamma, \pi-\delta]$, and $[\pi-\delta, \pi]$, where α, β, γ, and δ are defined as in the proof of Theorem 4.1.

It is straightforward to see that for four of the intervals, $a(\theta)b(\theta)$ and $a(\theta)+b(\theta)$ are either unchanged or increased. See Table 2. The only decrease occurs when $\theta \in [0,\alpha]$ or $\theta \in [\pi-\delta, \pi]$. However, $a(\theta)+b(\theta)$ remains unchanged over both of these intervals, i.e., $a(\theta) = k - b(\theta)$, where $k = a(\theta)+b(\theta)$ is constant. Since the function $x(k-x)$ is decreasing on the interval $x \in [k/2, k]$, it follows that $a(\theta)b(\theta)$ increases as p moves left.

Table 2. Analyzing the magnitude of $a(\theta)b(\theta)$ and $a(\theta)+b(\theta)$ as p moves along the x-axis towards the square's diagonal

range of θ	$a(\theta)$	$b(\theta)$	$a(\theta)b(\theta)$	$a(\theta)+b(\theta)$
$[0,\alpha]$	increases	decreases	?	unchanged
$[\alpha, \pi/2-\beta]$	increases	unchanged	increases	increases
$[\pi/2-\beta, \pi/2]$	unchanged	unchanged	unchanged	unchanged
$[\pi/2, \pi/2+\gamma]$	unchanged	unchanged	unchanged	unchanged
$[\pi/2+\gamma, \pi-\delta]$	unchanged	increases	increases	increases
$[\pi-\delta, \pi]$	decreases	increases	?	unchanged

Case 1b. Suppose $p_x = p_y$. Observe that $\alpha = \beta = \pi/2$. Consequently, the interval $[\alpha, \pi/2-\beta]$ is empty. Using argument analogous to that used in Case 1, it follows that $a(\theta)b(\theta)[a(\theta)+b(\theta)]$ increases as p moves along the diagonal $p_x = p_y$ toward the center of the square.

Together, Cases 1a and 1b imply that if p lies in the first octant then the maximum load is achieved at $p = (1/2, 1/2)$.

Case 2. Suppose p does not lie in the first octant of the square. Using an argument analogous to Case 1 for the corresponding octant, it follows that the maximum load increases as p approaches $(1/2, 1/2)$.

The result follows upon substituting $p = (1/2, 1/2)$ in (4.4). □

5. One-Turn Rectilinear Routing on a Square

In this section we consider various one-turn rectilinear routing policies on the unit square and compare these against straight-line routing. Our objective in designing these policies was to reduce the maximum load by redirecting routes for selected regions of origin-destination pairs away from high-traffic areas and towards low-traffic areas while maintaining a low stretch factor. Sections 5.1 and 5.2 begin by deriving the average load and stretch factor for any monotonic rectilinear routing policy. In Section 5.3 we introduce diagonal rectilinear routing, and derive the corresponding load at an arbitrary point and the maximum load. In Section 5.4 we describe other one-turn rectilinear routing policies considered whose maximum load is worse than that of diagonal rectilinear routing. These results are summarized in Section 5.5.

5.1. *Average Load*

Theorem 5.1. The average load for any monotonic rectilinear routing policy on the unit square is 2/3.

Proof. By Proposition 2.1 and (2.10), the average load is equal to the average ℓ_1 distance between two points in the unit square. This value is

$$\lambda_{\text{avg}}(P) = \int_0^1\int_0^1\int_0^1\int_0^1 |u_x - v_x| + |u_y - v_y| \, dv_y \, dv_x \, du_y \, du_x = \frac{2}{3}. \tag{5.1}$$

□

5.2. *Average Stretch Factor*

It is straightforward to see that the maximum stretch factor for any monotonic rectilinear routing policy is $\sqrt{2}$. We now consider the average stretch factor.

Theorem 5.2. The average stretch factor for any monotonic rectilinear routing policy P on the unit square is

$$\text{str}_{\text{avg}}(P) = \frac{1}{6}\left(10\ln(2+\sqrt{2}) + 2\sqrt{2} - 4 - 5\ln(2)\right) \approx 1.2737. \tag{5.2}$$

Proof. The result follows by reparameterizing (2.8) first to two parameters and then into polar coordinates. By (2.6), (2.8), and (2.10),

$$\text{str}_{\text{avg}}(P) = \int_0^1\int_0^1\int_0^1\int_0^1 \frac{|u_x - v_x| + |u_y - v_y|}{\sqrt{(u_x - v_x)^2 + (u_y - v_y)^2}} \, dv_y \, dv_x \, du_y \, du_x \tag{5.3}$$

By symmetry, it suffices to consider routes for which the destination appears above and to the right of the origin, resulting in a scalar factor of 4. The number of parameters can be reduced to two because stretch factor is invariant under translation. That is, every combination of length and orientation of line segment is weighed by the measure of the set of all similar line segments within the unit square. For each (x, y), let $L_{x,y}$ denote the set of line segments contained within the unit square that are parallel to and of equal length to the line segment from $(0, 0)$ to (x, y); the set $L_{x,y}$ has measure $(1 - x)(1 - y)$, resulting in the following reparameterization of (5.3):

$$= \int_0^1\int_0^1 4(1-x)(1-y)\frac{x+y}{\sqrt{x^2+y^2}}\,dy\,dx \tag{5.4}$$

Next we express (5.4) in polar coordinates by substituting $x = r\cos\theta$ and $y = r\sin\theta$. Angle θ is in the range $[0, \pi/2]$ since the destination is above and to the right of the origin. The range $[\pi/4, \pi/2]$ is symmetric with the range $[0, \pi/4]$ on the unit square. Consequently, a factor of 2 is introduced since θ ranges over $[0, \pi/4]$ in our calculation. A multiplicative factor of r is necessary for scaling by the distance from the point of rotation. This gives the following reparameterization of (5.4):

$$\begin{aligned} &= 2\int_0^{\pi/4}\int_0^{\sec\theta} 4(1-r\cos\theta)(1-r\sin\theta)r\left(\frac{r\cos\theta+r\sin\theta}{r}\right)dr\,d\theta \\ &= \frac{1}{6}\left(10\ln(2+\sqrt{2})+2\sqrt{2}-4-5\ln(2)\right). \qquad \square \end{aligned}$$

5.3. *Diagonal Rectilinear Routing*

We define a routing policy in terms of the partition of the unit square induced by its two diagonals. Let R_1 through R_4 denote the four regions of the partition such that R_1 is at the bottom of the square and the regions are numbered in clockwise order. If the origin lies in R_1 or R_3, the row-first route is selected. Otherwise, the column-first route is selected. See Figure 2. We refer to this routing policy, denoted P_D, as *diagonal rectilinear routing.*

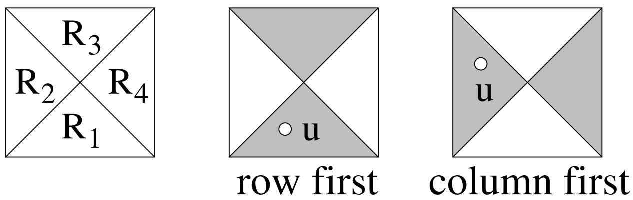

Fig. 2. The row-first route is selected if the origin u lies in regions R_1 or R_3 (regardless of the location of the destination). Otherwise, the column-first route is selected.

As we did in Section 4.2, we derive the load at an arbitrary point p located in an octant of the unit square since P_D is symmetric in the x- and y-dimensions. The load at an arbitrary point in the unit square is then easily found using the appropriate coordinate transformation.

Theorem 5.3. Given a point $p = (p_x, p_y)$ such that $0 \leq p_y \leq p_x \leq 1/2$, the load at p using diagonal rectilinear routing is

$$\lambda_{P_D}(p) = 2p_x^3 - 5p_x^2 + \frac{7}{2}p_x - 2p_xp_y + \frac{3}{2}p_y - 3p_y^2 + 2p_y^3. \tag{5.5}$$

Proof. Let $u = (u_x, u_y)$ denote the origin and let $v = (v_x, v_y)$ denote the destination. The relative positions of u, v, and p can be divided into seven cases such that the load at p corresponds to the sum of the measure of the regions of possible origin-destination combinations in each case. In Cases 1a through 1d, u lies in region R_1 or R_3 and, consequently, the row-first route is selected. In Cases 2a through 2c, u lies in region R_2 or R_4 and, consequently, the column-first route is selected. See Figure 3.

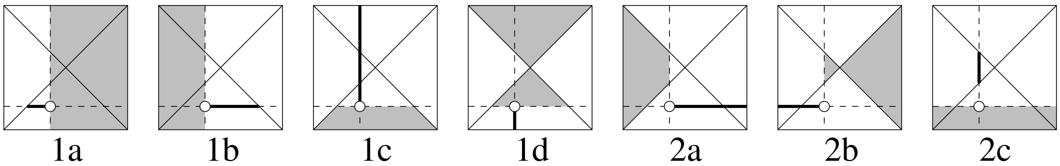

Fig. 3. Illustration in support of Theorem 5.3. The white dot denotes point p. In Cases 1a, 1b, and 2c the origin u lies on the highlighted line segment and the destination v lies in the shaded region; the reverse is true in the remaining cases.

Table 3. Measuring load in Cases 1a through 2c

Case	Region containing $\boldsymbol{u}$	Assumptions	Measure of set of possible origin-destination pairs
1a	$R_1 \cup R_3$	$u_y = p_y$, $u_x \leq p_x$, and $v_x \geq p_x$	$(p_x - p_y)(1 - p_x)$
1b	$R_1 \cup R_3$	$u_y = p_y$, $u_x \geq p_x$, and $v_x < p_x$	$(1 - p_x - p_y)p_x$
1c	$R_1 \cup R_3$	$u_y \leq p_y$, $v_x = p_x$, and $v_y \geq p_y$	$p_y(1 - p_y)^2$
1d	$R_1 \cup R_3$	$u_y \geq p_y$, $v_x = p_x$, and $v_y < p_y$	$p_y(1/2 - p_y + p_y^2)$
2a	$R_2 \cup R_4$	$u_x \leq p_x$, $v_x \geq p_x$, and $v_y = p_y$	$p_x(1 - p_x)^2$
2b	$R_2 \cup R_4$	$u_x \geq p_x$, $v_x < p_x$, and $v_y = p_y$	$p_x(1/2 - p_x + p_x^2)$
2c	$R_2 \cup R_4$	$u_x = p_x$ and $v_y \leq p_y$	$p_y(1 - 2p_x)$

Case 1a. Suppose $u \in R_1 \cup R_3$, $u_y = p_y$, $u_x \leq p_x$, and $v_x \geq p_x$. See Figure 3(1a). Point u must lie on the highlighted line segment of length $p_x - p_y$. Point v may lie anywhere in the shaded region of area $(1 - p_x)$. Therefore, the set of possible origin-destination pairs has measure $(p_x - p_y)(1 - p_x)$.

Cases 1b through 2c follow by analogous arguments. See Table 3. Summing these seven cases gives (5.5). □

Figure 8 displays a plot of (5.5) for $p \in [0,1]^2$. It is straightforward to find the roots of the derivatives of (5.5) with respect to p_x and p_y, respectively. Upon substitution back into (5.5), this gives:

Corollary 5.1. *The maximum load for diagonal rectilinear routing on the unit square is*

$$\lambda_{\max}(P_D) = \frac{1}{27}\left[\sqrt{11} - \frac{31}{2}\right] \approx 0.7771,$$

realized at

$$p_x = \frac{5}{6} - \frac{1}{3}\sqrt{3 - \frac{1}{2}\sqrt{11}} \approx 0.4472 \text{ and } p_y = \frac{2}{3} - \frac{\sqrt{11}}{6} \approx 0.1139.$$

5.4. *Additional Policies Considered*

We describe additional one-turn rectilinear routing policies that were considered good candidates for reducing the maximum load. In each case, the maximum load was shown to be strictly greater than that of diagonal rectilinear routing. Recall that all one-turn rectilinear routing policies have equal average load (Theorem 5.1). Values and bounds on maximum load for these policies are summaries in Table 4.

5.4.1. *Equal Distribution*

An *equal-distribution policy* is a one-turn rectilinear routing policy that assigns the pairs (u, v) and (v, u) different one-turn rectilinear routes for all u and v whenever u and v do not lie on the same row or column. Any such policy P corresponds to a bijection between the set of possible one-turn rectilinear routes and the set of origin-destination pairs. Such policies include assigning the one-turn rectilinear route that follows the row-first route, the route that follows a clockwise turn, or assigning each one-turn rectilinear route at random (in this case, the policy's load corresponds to its expected load). A simple equal-distribution strategy to consider is to assign to each origin-destination pair (u, v) the row-first route. For any point $p \in [0,1]^2$, $\lambda_{P_R}(p) = \lambda_P(p)$, where P_R denotes the row-first routing policy and P denotes any equal-distribution policy.

Theorem 5.4. Given a point $p = (p_x, p_y)$, the load at p using a one-turn rectilinear routing policy P_R with equal distribution is

$$\lambda_{P_R}(p) = 2[p_x(1 - p_x) + p_y(1 - p_y)]. \tag{5.8}$$

Proof. Let $u = (u_x, u_y)$ denote the origin and let $v = (v_x, v_y)$ denote the destination. As illustrated in Figure 4, four cases are possible:

Case 1. Suppose $u_y = p_y$ and $v_x \leq p_x$. Node u lies on the line segment of length $1 - p_x$. Node v lies in the rectangular region of area p_x. Therefore, the space of possible origin-destination pairs has measure $p_x(1 - p_x)$.

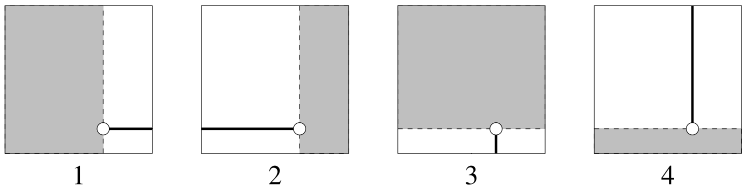

Fig. 4. Illustration in support of Theorem 5.4. The white dot denotes point p.

Case 2. Suppose $u_y = p_y$ and $v_x > p_x$. Again, we get $p_x(1 - p_x)$.
Case 3. Suppose $v_x = p_x$ and $u_y \geq p_y$. Analogously, we get $p_y(1 - p_y)$.
Case 4. Suppose $v_x = p_x$ and $u_y < p_y$. Again, we get $p_y(1 - p_y)$.
Summing these four cases gives (5.8). □

Figure 8 displays a plot of (5.8) for $p \in [0,1]^2$. It is straightforward to find the roots of the derivatives of (5.8) with respect to p_x and p_y, respectively. Upon substitution back into (5.8), this gives:

Corollary 5.2. *The maximum load for any one-turn rectilinear routing policy P_R with equal distribution is $\lambda_{\max}(P_R) = 1$, realized uniquely at $p = (1/2, 1/2)$.*

5.4.2. *Outer Turn*

Consider the routing policy that selects the one-turn rectilinear route whose intermediate point is furthest from the center of the square. If the two intermediate points are equidistant from the origin, then a route is assigned as in the equal distribution policy. See Figure 5.

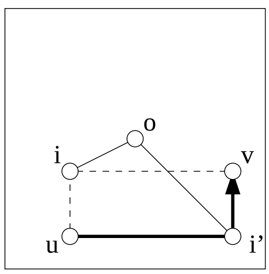

Fig. 5. The outer-turn rectilinear routing policy selects the route passing through intermediate point i', since point i' is further from the center of the square (o) than is point i.

Theorem 5.5. The outer-turn rectilinear routing policy P has maximum load bounded by

$$\lambda_{\max}(P) \geq \frac{1}{3}[2 + \ln(2)] \approx 0.8977. \tag{5.9}$$

Proof. We show that $\lambda_P(p) = (5.9)$ for $p = (1/2, 0)$. Let $u = (u_x, u_y)$ denote the origin and let $v = (v_x, v_y)$ denote the destination. Let $i = (u_x, v_y)$ and $i' = (v_x, u_y)$ denote the two possible intermediate points. Let o denote the center of the square. Since $p_y = 0$, any route that passes through p implies that $u_y = 0$ or $v_y = 0$.

Case 1. Suppose $u_y = 0$ and $v_x \geq 1/2$. The outer-turn policy selects the route passing through p if and only if $|o-i| \leq |o-i'|$. Observe that the position of i' is fixed for any given value of v_x. Furthermore, the measure of the set of possible positions for i corresponds to the area of the intersection of the circle of radius $|o-i'|$ centered at o and the left half of the unit square. Let R denote this region. See Figure 6A. Region R can be divided into two symmetric circular sectors and four symmetric right-angle triangles. Let $x = v_x - 1/2$ and let $\theta = \arctan(2x)$ denote the angle of the triangle opposite the side of length x. The area of each right-angle triangle is $x/4$. The area of each circular sector is $(\pi/4 - \theta)(x^2 + 1/4)$. Therefore, region R has area $2(\pi/4 - \theta)(x^2 + 1/4) + x$. See Figure 6B.

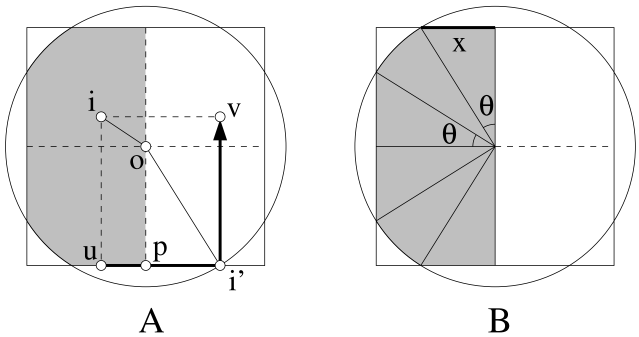

Fig. 6. Illustration in support of Theorem 5.5. Region R is shaded.

Therefore, the set of possible values for u_x, v_x, and v_y has measure

$$\int_0^{1/2} 2\left(\frac{\pi}{4} - \arctan(2x)\right)\left(x^2 + \frac{1}{4}\right) + x \, dx = \frac{1}{12}[2 + \ln(2)]. \tag{5.10}$$

Case 2. Suppose $u_y = 0$ and $v_x < 1/2$. Analogously, we get (5.10).
Case 3. Suppose $v_y = 0$ and $u_x \geq 1/2$. Again, we get (5.10).
Case 4. Suppose $v_y = 0$ and $u_x < 1/2$. Again, we get (5.10).
Summing these four cases gives (5.9). □

5.4.3. *Line Division*

Given an origin-destination pair (u, v), where $u \neq v$, let l denote the line passing through origin u and destination v. If l does not pass through the center of the square, o, select the one-turn rectilinear route whose intermediate point is opposite l from o. If l passes through o, then a route is assigned as in the equal distribution policy.

Theorem 5.6. The line-division rectilinear routing policy P has maximum load bounded by $\lambda_{\max}(P) \geq 7/8$.

Proof. We show that $\lambda_P(p) = 7/8$ for $p = (1/2, 0)$. Let $u = (u_x, u_y)$ denote the origin and let $v = (v_x, v_y)$ denote the destination. Since $p_y = 0$, any route that passes through p implies that $u_y = 0$ or $v_y = 0$.

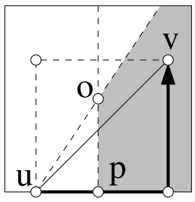

Fig. 7. Illustration in support of Theorem 5.6. Region R is shaded.

Case 1. Suppose $u_y = 0$ and $v_x \geq 1/2$. It is straightforward to see that v must lie in the shaded region R as illustrated in Figure 7. Region R has area $1/2-(1/2-u_x)/4$. The corresponding set of possible values for u_x, v_x, and v_y has measure

$$\int_0^{1/2} \frac{1}{2} - \frac{1}{4}\left(\frac{1}{2} - u_x\right)\, du_x = \frac{7}{32}. \tag{5.11}$$

Case 2. Suppose $u_y = 0$ and $v_x < 1/2$. Analogously, we get (5.11).
Case 3. Suppose $v_y = 0$ and $u_x \geq 1/2$. Again, we get (5.11).
Case 4. Suppose $v_y = 0$ and $u_x < 1/2$. Again, we get (5.11).
Summing these four cases gives 7/8. □

5.5. *Comparison and Summary of Results*

In this section we introduced diagonal rectilinear routing along with additional one-turn rectilinear routing policies. The corresponding values for maximum load are compared against that of straight-line routing in Table 4. Also included in Table 4 is a lower bound on maximum load for any one-turn rectilinear routing policy (see Section 6). We derived the load at an arbitrary point in the unit square for the diagonal and equal-distribution policies; plots of these loads are illustrated in Figure 8 along with the corresponding plot for straight-line routing. The plots provide intuition as to how the load is distributed to reduce the maximum load under the constraint that average load remains constant at 2/3. With respect to the objective of minimizing the maximum load, the diagonal rectilinear routing policy, P_D, achieves the lowest maximum load (0.7771), significantly lower than the maximum load of straight-line routing (1.1478) and not much greater than the lower bound (0.7076).

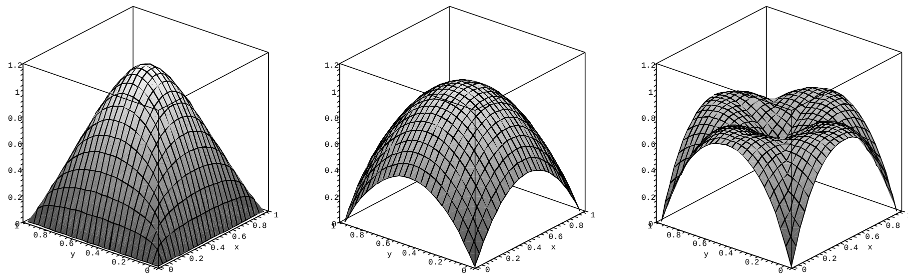

Fig. 8. These plots display $\lambda_P(p)$ for $p \in [0,1]^2$ for three routing policies: (left to right) straight-line S, equal distribution P_R, and diagonal P_D.

6. Lower Bounds on Load for One-Turn Rectilinear Routing Policies

Naturally, no monotonic rectilinear routing policy can have a maximum load less than the average load of 2/3. In this section we establish a stronger lower bound on the maximum load of any one-turn rectilinear routing policy.

Theorem 6.1. No one-turn rectilinear routing policy can guarantee a maximum load less than 0.7076.

In brief, the proof capitalizes on the observation that load is low near the corner regions of the square for any one-turn rectilinear routing. An upper bound on the average load in the corner regions provides a corresponding lower bound on the average load and, therefore, maximum load, in the remaining region of the unit square.

Let R_1 denote the lower left corner region of the unit square $[0,1]^2$ bounded by the diagonal from $(0,k)$ to $(k,0)$ for a fixed $k \in (0,1/2)$. Let $\lambda_{\text{avg}}^{R_1}(P)$ denote the average load for points in R_1 under policy P on the unit square. The proof of Theorem 6.1 relies on the following lemma.

Lemma 6.1. *For any one-turn rectilinear routing policy P on the unit square,*

$$\lambda_{\text{avg}}^{R_1}(P) \leq \frac{2k^3}{15} - \frac{4k^2}{3} + 2k.$$

Proof. Choose any one-turn rectilinear routing policy P. Let $p = (p_x, p_y)$ denote a point in R_1. Let $u = (u_x, u_y)$ and $v = (v_x, v_y)$ denote the origin and destination, respectively, of a one-turn rectilinear route that passes through p. Up to two routes are possible for every such u and v, one of which passes through p. We consider eight cases with respect to the relative positions of p, u, and v. See Figure 9. An upper bound on average load in R_1 is given by considering the load at p for the route that contributes the greatest load to R_1 in each case, summing these contributions for all cases, and integrating over all p in R_1.

Case 1. Suppose $u_x = p_x$, $u_y \geq p_y$, $v_x \leq p_x$, and $v_y \leq p_y$. By symmetry, this is equivalent to the case in which u and v are interchanged (the same holds for Cases 2 through 8). Both routes from u to v have equal contributions to load in R_1 but only one of these passes through p. The contribution to load at p is at most $(1-p_y)p_xp_y + p_y(1-p_x)(1-p_y)$.

Case 2. Suppose $u_x = p_x$, $u_y \geq p_y$, $v_x \geq p_x$, and $v_y \leq p_y$. The route from u to v that maximizes load in R_1 passes through p. Therefore, the contribution to load at p is at most $2(1-p_y)p_y(1-p_x)$.

Case 3. Suppose $u_x = p_x$, $u_y \leq p_y$, $v_x \geq p_x$, and $v_y \geq p_y$. An argument analogous to Case 1 shows that the contribution to load at p is at most $(1-p_y)p_xp_y + p_y(1-p_x)(1-p_y)$.

Case 4. Suppose $u_x = p_x$, $u_y \leq p_y$, $v_x \leq p_x$, and $v_y \geq p_y$. The route from u to v that maximizes load in R_1 avoids p. Therefore, the contribution to load at p is 0.

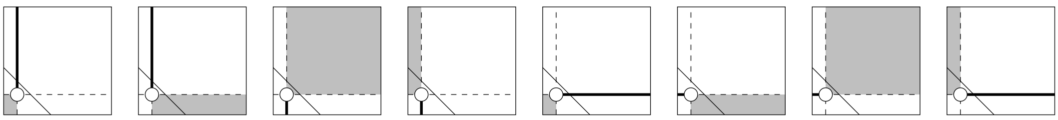

Fig. 9. Illustration in support of Theorem 6.1. The white dot denotes point p. **Cases 2. & 8.** The route must pass through p since it maximizes the load in R_1. **Cases 1., 3., 5. & 7.** Both routes contribute equally to the load in R_1. **Cases 4. & 6.** This route will not pass through p since it does not maximize the load in R_1.

Analogous arguments give $(1-p_x)p_xp_y$ for Case 5, 0 for Case 6, $p_x(1-p_y)(1-p_x)$ for Case 7, and $2(1-p_x)p_x(1-p_y)$ for Case 8. The load at p is bounded from above by the sum of the upper bounds on the contributions of Cases 1 though 8:

$$\forall p \in R_1,\ \lambda_P(p) \leq 3p_x(1-p_x) + 3p_y(1-p_y) + 2p_xp_y(p_x+p_y) - 4p_xp_y. \tag{6.2}$$

By (2.2) and (6.2),

$$\lambda_{\text{avg}}^{R_1}(P) = \frac{1}{\text{Area}(R_1)} \int_0^k \int_0^{k-x} \lambda_P(p)\, dp_y\, dp_x \leq \frac{2k^3}{15} - \frac{4k^2}{3} + 2k. \qquad \square$$

The upper bound on average load in corner regions implies a corresponding lower bound on average load in the complementary region, which in turn provides a lower bound on the maximum load over the entire unit square. We now prove Theorem 6.1.

Proof. Choose any one-turn rectilinear routing policy P. Region R_1 is symmetric to the three other corner regions of the unit square, denoted by R_2, R_3, and R_4. Let $R = R_1 \cup R_2 \cup R_3 \cup R_4$, let $Q = [0,1]^2 - R$, and let $\lambda_{\text{avg}}^{Q}(P)$ denote the average load in Q. By (2.2), Theorem 5.1, and Lemma 6.1,

$$\begin{aligned}
& \lambda_{\mathrm{avg}}(P) = \mathrm{Area}(R)\lambda^{R_1}_{\mathrm{avg}}(P) + \mathrm{Area}(Q)\lambda^{Q}_{\mathrm{avg}}(P) \\
\Leftrightarrow \quad & \frac{2}{3} = 2k^2\lambda^{R_1}_{\mathrm{avg}}(P) + (1-2k^2)\lambda^{Q}_{\mathrm{avg}}(P) \\
\Leftrightarrow \quad & \lambda^{Q}_{\mathrm{avg}}(P) = \frac{\frac{2}{3} - 2k^2\lambda^{R_1}_{\mathrm{avg}}(P)}{1-2k^2} \\
& \geq \frac{10 - 4k^5 + 40k^4 - 60k^3}{15(1-2k^2)}. \qquad (6.4)
\end{aligned}$$

Exp. (6.4) is maximized when $k \approx 0.3271$; the exact value of k corresponds to the unique root of

$$6k^5 - 40k^4 + 25k^3 + 40k^2 - 45k + 10 = 0$$

whose value lies in the interval $(0, 1/2)$. Substituting this value for k in (6.4) gives $\lambda^{Q}_{\mathrm{avg}}(P) \geq 0.7076$. Let $\lambda^{Q}_{\max}(P)$ denote the maximum load in Q. By (2.2) and (2.5), and because $Q \subseteq [0,1]^2$,

$$\lambda_{\max}(P) \geq \lambda^{Q}_{\max}(P) \geq \lambda^{Q}_{\mathrm{avg}}(P) \geq 0.7076. \qquad \square$$

7. Optimal Randomized One-Turn Rectilinear Routing Policies

In this section we give a characterization of the optimal randomized one-turn rectilinear policy as the solution of an optimization problem and provide an efficient procedure for approximating it. A deterministic one-turn rectilinear policy is equivalent to a function $P : [0,1]^4 \to \{0,1\}$ where $P(u,v,s,t) = 1$ if and only if the route from (u,v) to (s,t) uses the column-first path. (Note that in all rectilinear schemes if $u = s$ or $v = t$ the straight-line path is always taken. In this case we define $P(u,v,u,t) = P(u,v,s,v) = 1$.) We can generalize this to randomized rectilinear schemes by considering $Q : [0,1]^4 \to [0,1]$ where if $Q(u,v,s,t) = q$ then a packet travelling from point (u,v) to (s,t) takes the the column-first path with probability q and the row-first path with probability $1-q$. For a given Q the expected load at a point (x,y) is given by

$$\begin{aligned}
\lambda(x,y) = & \int_x^1 \int_0^x \int_0^1 Q(s,t,u,y) + 1 - Q(u,y,s,t)\, dt\, ds\, du \\
& + \int_0^x \int_x^1 \int_0^1 Q(s,t,u,y) + 1 - Q(u,y,s,t)\, dt\, ds\, du \\
& + \int_0^y \int_0^1 \int_y^1 Q(x,v,s,t) + 1 - Q(s,t,x,v)\, dt\, ds\, dv \\
& + \int_y^1 \int_0^1 \int_0^y Q(x,v,s,t) + 1 - Q(s,t,x,v)\, dt\, ds\, dv.
\end{aligned}$$

The optimal policy is given by the solution to the following optimization problem:

$$\min_{Q} \max_{(x,y)} \lambda(x,y).$$

While we cannot directly solve this problem, we can approximate it by considering finer and finer partitions of the square into n^2 subsquares of size $1/n$ by $1/n$ and giving a policy for all packets routing between each pair of subsquares. Now our problem is equivalent to finding a randomized one-turn rectilinear routing policy for an $n \times n$ grid that minimizes the number of packets using any particular node of the grid under an all-to-all communication pattern.

Let p_{ijkl}, $1 \leq i, j, k, l \leq n$, be the probability that a packet starting in subsquare (i, j) going to subsquare (k, l) uses the column-first path. The expected load at any point in subsquare (r, s) for the case when $1 \leq r, s \leq \lceil n/2 \rceil$ is bounded from above by:

$$\begin{aligned}
\lambda(r,s) \leq \frac{1}{n^3}\Bigg(& \sum_{i=r+1}^{n}\sum_{k=1}^{r}\sum_{l=1,l\neq s}^{n}(p_{klis}+1-p_{iskl}) + \sum_{i=1}^{r-1}\sum_{k=r}^{n}\sum_{l=1,l\neq s}^{n}(p_{klis}+1-p_{iskl}) \\
& + \sum_{j=1}^{s-1}\sum_{k=1,k\neq r}^{n}\sum_{l=s+1}^{n}(p_{rjkl}+1-p_{klrj}) + \sum_{j=s+1}^{n}\sum_{k=1,k\neq r}^{n}\sum_{l=1}^{s-1}(p_{rjkl}+1-p_{klrj}) \\
& + 2(n-s)(s-1) + 2(n-r)(r-1) + (n-1)(n-r+n-s) \\
& + 2(r-1) + 2(s-1) + \frac{2(n-r)^2}{n-1} + \frac{2(n-s)^2}{n-1} + 1\Bigg).
\end{aligned}$$

Bounds for the cases $\lceil n/2 \rceil < r \leq n$, $1 \leq s \leq \lceil n/2 \rceil$; $\lceil n/2 \rceil < s \leq n$, $1 \leq r \leq \lceil n/2 \rceil$; and $\lceil n/2 \rceil < r, s \leq n$ are similar. Our problem now reduces to

$$\min_{p_{ijkl}} \max_{r,s} \lambda(r,s),$$

which is equivalent to the following linear program with n^4+1 variables and $2n^4+n^2$ constraints (solvable in polynomial time):

$$\begin{aligned}
&\text{Minimize } z \text{ subject to}\\
&0 \leq p_{ijkl} \leq 1,\\
&1 \leq i, j, k, l \leq n,\\
&z - \lambda(r,s) \geq 0,\\
&1 \leq r, s \leq n.
\end{aligned}$$

Table 5 shows an upper bound on the maximum load achieved by the policy obtained by using an $n \times n$ grid to approximate the unit square for $2 \leq n \leq 20$. The results indicate[b] that the optimal policy achieves a maximum load of approximately 0.73. The solutions were found using the Gnu Linear Programming Kit[19]. We were unable to obtain results for larger n due to memory limitations.

[b]Although the sequence converges, due to increased load in the central subsquare when n is odd, the difference between successive values for maximum load alternates in magnitude; observe that the differences between successive odd values of n and successive even values of n are both decreasing sequences that converge to zero.

The approach taken here can easily be extended to find the optimal randomized policy for a $k \times 1$ rectangle using a grid of size $n \times kn$. A similar mixed integer linear program could be developed to approximate the best deterministic one-turn rectilinear scheme. This would be equivalent to computing the forwarding index of the $n \times n$ grid when the paths are restricted to one-turn rectilinear paths. We were unable to determine if the resulting problem is solvable in polynomial time. (The forwarding index problem is NP-hard in general[27].) One could extend this idea to obtain a linear program approximation to the best monotonic rectilinear scheme but the resulting linear program would have an exponential number of variables (for each of the exponential number of potential paths).

8. Generalizing to Rectangular Regions

In this section we briefly mention generalizations of our results to a $k \times 1$ rectangular region. These results follow by arguments analogous to those used to prove the corresponding results on square regions; as such, the proofs are straightforward to derive and have been omitted.

8.1. *Straight-Line Routing*

The proof of Theorem 4.2 generalizes to rectangles. The corresponding maximum load under straight-line routing for any k is

$$\lambda_{\max}(S) = \frac{1}{4}\left[2k\sqrt{k^2+1} + k^3\ln(1+\sqrt{k^2+1}) - k^3\ln(k) + \ln(k+\sqrt{k^2+1})\right].$$

As shown by Santaló[28], the average distance between two points in the rectangle is

$$\begin{aligned}\text{length}_{\text{avg}}(S) = \frac{1}{15}\Bigg[&k^3 + \frac{1}{k^2} + \sqrt{1+k^2}\left(3 - k^2 - \frac{1}{k^2}\right)\\ &+\frac{5}{2}\left(\frac{1}{k}\ln\left(k+\sqrt{1+k^2}\right) + k^2\ln\left(\frac{1+\sqrt{1+k^2}}{k}\right)\right)\Bigg],\end{aligned}$$

for all $k \geq 1$. By Proposition 2.1 it follows that $\lambda_{\text{avg}}(S) = k \cdot \text{length}_{\text{avg}}(S)$.

8.2. *Diagonal Rectilinear Routing*

Theorem 5.3 also generalizes, giving that for any k the load under diagonal rectilinear routing at a point $p = (p_x, p_y)$ is

$$\lambda_{P_D}(p) = \begin{cases} \frac{7}{2}kp_x - k^2p_y - 5p_x^2 + \frac{5}{2}kp_y \\ -3kp_y^2 + 2kp_y^3 + \frac{2}{k}p_x^3 - 2p_xp_y, & \text{if } p_x \in [0,k/2], p_y \in [0,p_x/k], \\ \frac{7}{2}kp_y - 5kp_y^2 - p_x + \frac{5}{2}kp_x \\ -3kp_x^2 + \frac{2}{k}p_x^3 - 2p_xp_y + 2kp_y^3, & \text{if } p_x \in [0,kp_y], p_y \in [0,1/2], \end{cases} \tag{8.2}$$

where $0 \leq p_x \leq k/2$ and $0 \leq p_y \leq 1/2$. For $k \geq 3/2$, we conjecture that (8.2) achieves maxima at $p \in \{(k/2,0),(k/2,1)\}$ with corresponding load $\lambda_{\max}(P_D) = 3k^2/4$. By

Proposition 2.1, the average load for any monotonic rectilinear routing policy P is

$$\lambda_{\text{avg}}(P) = k \cdot \text{length}_{\text{avg}}(P_D) = \frac{1}{k} \int_0^k \int_0^1 \int_0^k \int_0^1 |u_x - v_x| + |u_y - v_y| \, dv_y \, dv_x \, du_y \, du_x = \frac{k(k+1)}{3},$$

providing a generalization of Theorem 5.1. Observe that

$$\lim_{k \to \infty} \frac{\lambda_{\text{avg}}(P)}{\lambda_{\text{avg}}(S)} = 1,$$

where P denotes any monotonic rectilinear routing policy. An argument based on bisection width gives a lower bound of $k^2/2$ on maximum load for any routing policy.

9. Discussion

Applicability to the Discrete Network Setting. We have investigated the class of one-turn rectilinear routing policies on square and rectangular regions with the aim of reducing the maximum load while maintaining a low stretch factor. Our techniques model a dense wireless network with uniformly distributed nodes as a continuous space, enabling us to analyze the maximum or average load for various routing policies. We now consider how our results in the continuous setting relate to a discrete network setting. In particular, how would straight-line or one-turn rectilinear routing policies actually be implemented? Straight-line routing can be approximated by using either greedy[22] or compass routing[20]. The calculation that a forwarding node performs to determine the next node is entirely local; a node only needs to know its own coordinates, the coordinates of its neighbors, and the coordinates of the destination.

For one-turn rectilinear routing policies, suppose a message originating at a node located at (u_x, u_y) is trying to reach its destination node located at (v_x, v_y) using a row-first rectilinear route. The node can approximate a straight-line route to the point (v_x, u_y) (where there may or may not be a network node located) and then approximate a straight-line route to the final destination. The packet header needs only to contain the final destination's address and one bit to indicate whether a row-first or a column-first rectilinear route is being followed. In the diagonal rectilinear policy, for instance, a node can determine from its own coordinates whether a row-first or column-first route should be chosen, set the appropriate bit in the packet header, and forward appropriately.

In curveball routing[26], every node must store or compute its spherical coordinates in addition to knowing its Cartesian coordinates within the unit disk. This information is not local; to compute its spherical coordinates, a node requires knowledge of its relative position between the center and the boundary of the disk. The network flows described in Ref. 16 have no obvious simple implementation on a discrete network.

Directions for Future Research. The results of Section 7 make it clear that randomization can help in lowering the maximum load induced by a rectilinear policy. Finding a simple randomized rectilinear policy that would improve on the diagonal rectilinear policy would be an interesting challenge. While the results for squares translate easily to rectangles, it would be interesting to investigate other convex regions, as well as regions that contain holes.

While the focus of this paper is on one-turn rectilinear routing policies, many other strategies could result in possible solutions and remain to be analyzed. For example, instead of selecting one of two rectilinear intermediate points at random as in the equal distribution routing policy, an intermediate point could be selected at random from a subregion of the unit square. An intermediate point selected uniformly at random from the entire unit square and then connected to the origin and destination by straight-line routes results in an expected load exactly double that of straight-line routing. Perhaps a better strategy is one similar to that described by Busch et al.[3] for graphs embedded in the plane, which can be generalized to convex regions. In this case, an intermediate point is selected at random from an interval of the perpendicular bisector of the origin and destination nodes. At this point, no non-trivial bounds on maximum load are known for this strategy. Popa et al.[26] mention empirical results for adapting the curveball routing policy to the unit square. Theoretical bounds on load for this strategy remain to be analyzed. Finally, Hyytiä and Virtamo[16] have recently analyzed flows and vector fields to define generalized routing strategies on the unit disk and unit square. Although these results provide values for loads that can theoretically be achieved, these loads remain to be realized by a simple distributed routing policy. These are some of the several directions that remain to be explored to identify simple routing strategies on convex planar regions with the objective of minimizing maximum load and stretch factor.

References

1. D. H. Bailey, J. M. Borwein, and R. E. Crandall. Box integrals. *Journal of Computational and Applied Mathematics*, 206(1):196–208, 2007.
2. P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks*, 7:609–616, 2001.
3. C. Busch, M. Magdon-Ismail, and J. Xi. Oblivious routing on geometric networks. In *Proceedings of the ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, volume 17, 2005.
4. F. R. K. Chung, E. G. Coffman Jr., M. I. Reiman, and B. Simon. The forwarding index of communication networks. *IEEE Transactions on Information Theory*, 33(2):224–232, 1987.
5. S. R. Dunbar. The average distance between points in geometric figures. *College Mathematics Journal*, 28(3):187–197, 1997.
6. S. Durocher, E. Kranakis, D. Krizanc, and L. Narayanan. Balancing traffic load using one-turn rectilinear routing. In *Proceedings of the Conference on Theory and Applications of Models of Computation (TAMC)*, volume 4978 of *Lecture Notes in Computer*

Science, pages 473–484. Springer-Verlag, 2008.

7. Y. Ganjali and A. Keshavarzian. Load balancing in ad hoc networks: single-path routing vs. multi-path routing. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pages 1120–1125, 2004.
8. J. Gao and L. Zhang. Load balanced short path routing in wireless networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, volume 23, pages 1099–1108, 2004.
9. J. Gao and L. Zhang. Tradeoffs between stretch factor and load balancing ratio in routing on growth restricted graphs. In *Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC)*, pages 189–196, 2004.
10. P. Gupta and P. R. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, 2000.
11. M. C. Heydemann, J. C. Meyer, and D. Sotteau. On forwarding indices of networks. *Discrete Applied Mathematics*, 23:103–123, 1989.
12. B. Hull, K. Jamieson, and H. Balakrishnan. Mitigating congestion in wireless sensor networks. In *Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SENSYS)*, pages 134–147, 2004.
13. E. Hyytiä, P. Lassila, and J. Virtamo. Spatial node distribution of the random waypoint mobility model with applications. *IEEE Transactions on Mobile Computing*, 6(5):680–694, 2006.
14. E. Hyytiä and J. Virtamo. On optimality of single-path routes in massively dense wireless multi-hop networks. In *Proceedings of the ACM International Symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM)*, pages 28–35, 2007.
15. E. Hyytiä and J. Virtamo. On traffic load distribution and load balancing in dense wireless multihop networks. *EURASIP Journal on Wireless Communications and Networking*, 2007:1–15, 2007.
16. E. Hyytiä and J. Virtamo. Near-optimal load balancing in dense wireless multi-hop networks. In *Proceedings of the Conference on Next Generation Internet Networks (NGI)*, pages 181–188, 2008.
17. D. Johnson, Y. Hu, and D. Maltz. The dynamic source routing protocol (DSR) for mobile ad hoc networks for IPv4. RFC 4728 (Experimental), 2007.
18. K.-H. Kim, W.-D. Jung, J.-S. Park, H.-G. Seo, S.-H. Jo, C.-M. Shin, S.-M. Park, and H.-N. Kim. A resilient multipath routing protocol for wireless sensor networks. In *Proceedings of the International Conference on Networking*, volume 3421 of *Lecture Notes in Computer Science*, pages 1122–1129. Springer-Verlag, 2005.
19. GLPK: Gnu Linear Programming Kit. http://www.gnu.org/software/glpk.
20. E. Kranakis, H. Singh, and J. Urrutia. Compass routing on geometric networks. In *Proceedings of the Canadian Conference on Computational Geometry (CCCG)*, pages 51–54, 1999.
21. J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris. Capacity of ad hoc wireless networks. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, pages 61–69, 2001.
22. X. Lin and I. Stojmenović. GEDIR: Loop-free location based routing in wireless networks. In *Proceedings of the Conference on Parallel and Distributed Computing and*

Systems (IASTED), pages 1025–1028, 1999.

23. F. Meyer auf der Heide, C. Schindelhauer, K. Volbert, and M. Grunwald. Energy, congestion, and dilation in radio networks. In *Proceedings of the ACM Symposium on Parallel Algorithms and Architectures (SPAA)*, pages 230–237, 2002.
24. V. Park and S. Corson. A highly adaptive distributed routing algorithm for mobile wireless networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pages 1405–1413, 1997.
25. P. P. Pham and S. Perreau. Performance analysis of reactive shortest path and multipath routing mechanism with load balance. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, pages 251–259, 2003.
26. L. Popa, A. Rostami, R. M. Karp, C. Papadimitriou, and I. Stoica. Balancing traffic load in wireless networks with curveball routing. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing (MOBIHOC)*, pages 170–179, 2007.
27. R. Saad. Complexity of the forwarding index problem. *SIAM Journal on Discrete Mathematics*, 6(3):418–427, 1993.
28. L. A. Santaló. *Integral Geometry and Geometric Probability*. Cambridge University Press, 2004.
29. I. Stojmenovic. Position based routing in ad hoc networks. *IEEE Communications*, 40(7):128–134, 2002.
30. S. Toumpis. Mother nature knows best: A survey of recent results on wireless networks based on analogies with physics. *Computer Networks*, 52:360–383, 2008.
31. C. Y. Wan, S. B. Eisenman, and A. T. Campbell. CODA: Congestion detection and avoidance in sensor networks. In *Proceedings of the ACM Conference on Embedded Networked Sensor Systems (SENSYS)*, pages 266–279, 2003.

Table 4. Comparing routing policies on the unit square

routing policy	**avg. load**	**max. load**
straight-line S	0.5214	1.1478
diagonal P_D	2/3	0.7771
equal distribution P_R	2/3	1
outer turn	2/3	≥ 0.8977
division line	2/3	$\geq 7/8$
lower bound		0.7076

Table 5. Approximations to the optimal randomized policy using an $n \times n$ grid, for $n \leq 20$

n	**max. load**								
		5	0.8009	9	0.7610	13	0.7435	17	0.7353
2	1.0000	6	0.7813	10	0.7530	14	0.7404	18	0.7329
3	0.8889	7	0.7759	11	0.7499	15	0.7393	19	0.7325
4	0.8264	8	0.7650	12	0.7446	16	0.7364	20	0.7310