

Research Article

Curvature of Indoor Sensor Network: Clustering Coefficient

F. Ariaei, M. Lou, E. Jonckheere, B. Krishnamachari, and M. Zuniga

Department of Electrical Engineering-Systems, University of Southern California, Los Angeles, CA 90089-2563, USA

Correspondence should be addressed to E. Jonckheere, jonckhee@usc.edu

Received 14 June 2008; Accepted 18 November 2008

Recommended by Sayandev Mukherjee

We investigate the geometric properties of the communication graph in realistic low-power wireless networks. In particular, we explore the concept of the curvature of a wireless network via the clustering coefficient. Clustering coefficient analysis is a computationally simplified, semilocal approach, which nevertheless captures such a large-scale feature as congestion in the underlying network. The clustering coefficient concept is applied to three cases of indoor sensor networks, under varying thresholds on the link packet reception rate (PRR). A transition from positive curvature (“meshed” network) to negative curvature (“core concentric” network) is observed by increasing the threshold. Even though this paper deals with network curvature per se, we nevertheless expand on the underlying congestion motivation, propose several new concepts (network inertia and centroid), and finally we argue that greedy routing on a virtual positively curved network achieves load balancing on the physical network.

Copyright © 2008 F. Ariaei et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1. INTRODUCTION

With the advent of wired and wireless networks, graph theory has seen a renewed interest, as it provides a mathematical model of the interconnection of the various communication channels, along with a cost associated with each channel. The latter network model is conceptualized as a (possibly directed) weighted graph. Along with the widespread utilization of graph models of networks, those graph properties embodying their large size and complexity and having a direct bearing on the communications problems have been the more specific targets of the recent investigations.

In the context of wireless networks, the idealized model of random geometric graphs $G(n, R)$ has been studied in great depth [1–5]. In this model, n nodes are scattered uniformly at random in a given area and any pair of nodes within a Euclidean distance R is connected with an edge. Recent empirical studies of low-power wireless sensor networks [6–10] have, however, shown that the real situation is more nuanced: between the distance range within which there is perfect connectivity and a range beyond which the link does not exist lies a large transitional region/gray area which is characterized by high variance in link quality (as measured by the packet reception rate (PRR)). It is of crucial interest to understand the fundamental properties of these realistic wireless networks.

More closely related to the present paper is the fact that the $G(n, R)$ model utilizes the geographical distance between agents, whereas in the context of wireless transmission a more relevant distance is $-\log \text{PRR}(v^i, v^j) \neq d_{\text{geographical}}(v^i, v^j)$. It turns out that the $G(n, R)$ model of uniformly distributed sensor relative to the geographical distance is positively curved [11]. However, relative to the communication distance $-\log \text{PRR}(v^i, v^j)$ the sensors look nonuniformly distributed and a general result asserts that the resulting Delaunay triangulation is negatively curved [12, 13]. The present paper utilizes the communication distance and hence reveals curvatures different than the mere vanishing one [14]. Even though the triangulation is random [14] because of idiosyncrasies of the propagation, the curvature, however, appears robust.

The preceding considerations call for a Riemannian geometry approach to analyzing such wireless networks. From a more practical standpoint, the proposed approach is motivated by the need to understand the various minimum communication cost flows on the graph and the potentially resulting congestion [15–23]. In Riemannian geometry [16], cost minimizing paths are conceptualized as geodesics, and the fundamental properties of the latter are encapsulated in that single parameter—the curvature. Among those flow properties regulated by the curvature, one can mention the exponential growth of balls in negative curvature [17], which

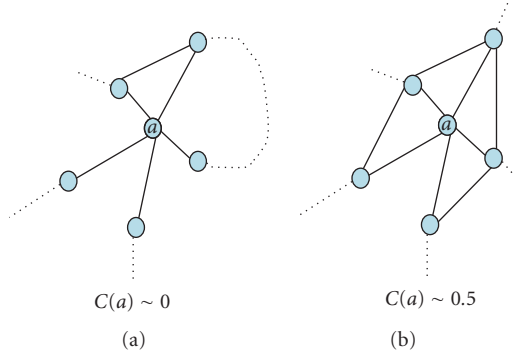


FIGURE 1: Illustration of clustering coefficient of node a . Solid lines between nodes indicate direct links of weight 1, while dotted lines represent multiple link paths of weight >1 . The total number of possible triangles is 10. In Figure (a), the clustering coefficient is $1/10$, while in Figure (b) it is $4/10$.

is a model of worm propagation [18], the reduced sensitivity of the geodesics to link cost variation in negative curvature, which is a model of the fluttering problem, the availability of a great many quasigeodesics in negative curvature [17], which is a model of multipath routing [19, 20], the existence of a unique centroid of a negatively curved manifold, which is a model of congestion, and so forth. Those Riemannian features relevant to communication call for a Riemannian analysis of graphs along with a curvature concept for graphs.

A Riemannian analogue of graphs that has been quite successful in its application to wired networks of massive size is provided by Gromov's coarse geometry [17, 21], modified so as to make it useful at scales relevant to real-life networks [22, 24]. The latter relies on a distance-based approach to curvature that emulates the Riemannian geometry premise that curvature regulates geodesic flows.

The present paper specifically investigates how a semilocal curvature concept, based on the clustering [15], applies to indoor sensor networks. This approach is "semilocal," in the sense that it not only takes into consideration the neighbors of a vertex like the popular degree/heavy-tail analysis, but it also takes into consideration the way the neighbors of the nominal vertex are wired. The latter is crucial, as it provides a quick snapshot at congestion around the nominal vertex. The semiglobal analysis of [22, 24], closer to the mathematically idealized Gromov analysis, is more accurate, but at the expense of accrued computational complexity. One of the premises of Riemannian geometry that extends to distance-based geometry is that a uniformly bounded local curvature implies global properties. The most salient practical manifestation of this fact is that a network with uniformly negative local curvature has a centroid through which most of the (global) traffic transits. Since real-life networks could have high variance in their local properties, here, this heterogeneity is analyzed by means of the distribution of the local curvature across the network.

Another curvature concept, very much in the same spirit, but somewhat more closely related to Gauss curvature, is the one based on Alexandrov angles. The latter is expanded

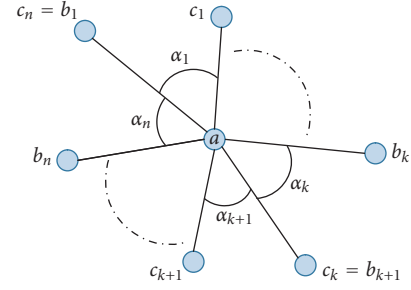


FIGURE 2: Gluing of triangles to make a surface of various curvatures depending on the sum of the α_k 's.

upon in a companion paper [25], where it is shown that the clustering and the Alexandrov angles analyses of the benchmark real-life sensor networks are fully consistent.

As already said, and as we show in Sections 6 and 7, the results we obtain have some practical applications. However, there are deeper implications that deserve further study. In particular, there is a tradeoff in the energy costs associated with minimum length routing paths that are impacted by the connection we find between the network's global curvature and the "blacklisting" threshold chosen for the link packet reception rate.

2. FROM CONGESTION TO CLUSTERING, CURVATURE AND BETWEENNESS

Consider a network $G = (V, E)$ specified by its vertex set V and its edge set E , along with a routing based on the number of hops. We proceed to show how congestion naturally leads to such a mathematical concept as clustering. Consider a network node $a \in V$ along with its neighboring vertices $N(a) = \{v \in V : av \in E\}$. Take two neighboring vertices $b, c \in N(a)$. If the nodes b, c are not directly connected, that is, if $bc \notin E$, messages from b to c will transit via a , hence congesting a . If, on the other hand, $bc \in E(G)$, messages from b to c will follow the edge bc , hence not contributing to congesting a . Consider a demand function $\Lambda_d : V \times V \rightarrow \mathfrak{R}^+$, where $\Lambda_d(x, y)$ is a transmission rate to be achieved from the source x to the destination y . If the demand is uniformly distributed over $N(a) \times N(a)$, the congestion at the nominal node a can be defined as proportional to the number of geodesics paths $ba \cup ac$ traversing a . The latter is equal to the total number of paths $ba \cup ac$ minus the number of those making a triangle Δabc . Hence the congestion is

$$\begin{aligned} \Lambda_T(a) &= \left(\binom{\deg(a)}{2} - |\{\Delta abc : b, c \in N(a)\}| \right) \Lambda_d \\ &= \left(1 - \underbrace{\frac{|\{\Delta abc : b, c \in N(a)\}|}{\binom{\deg(a)}{2}}}_{c(a)} \right) \binom{\deg(a)}{2} \Lambda_d. \end{aligned} \quad (1)$$

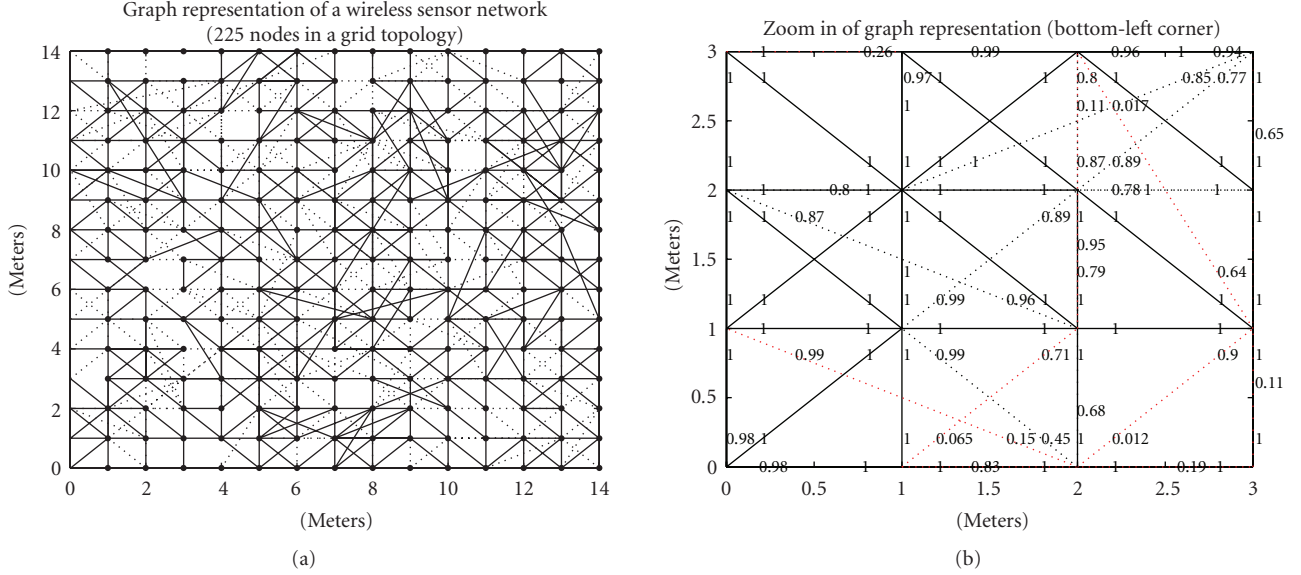


FIGURE 3: (a) Asymmetric graph; 225 nodes. (b) Zoom in of asymmetric graph: bottom-left corner, 16 nodes. The PRR of a given directed link is written close to the transmitter. For example, the link from (0, 0) to (1, 0) has a PRR of 0.98, and the link from (1, 0) to (0, 0) has a PRR of 1.

If we define the clustering coefficient $c(a)$ as above, the congestion at the node a , defined as the numbers of packets transiting per second through a in a greedy routing, is

$$\Lambda_T(a) = (1 - c(a)) \binom{\deg(a)}{2} \Lambda_d. \quad (2)$$

The last factor of the right-hand side reveals the trivial feature that the congestion is proportional to the demand. The middle factor is the traditional “heavy-tailed” paradigm that the congestion at node a should depend on the degree of the node a . The first factor is the novel feature that the congestion depends on a more subtle topological feature—the clustering coefficient.

3. MATHEMATICAL BACKGROUND: FROM CLUSTERING TO LOCAL CURVATURE

Clustering and curvature are concepts that are, here, applied to graphs. The connection between the two concepts is easily understood by considering a complete graph. Interpreting clustering as a measure of connectivity, such graph has high clustering coefficient. But geometrically, a complete graph embedded in a high-dimensional space “looks like” a sphere, which is the archetypical example of a positively curved manifold. Hence high clustering is equivalent to positive curvature.

Here the vertex set V is endowed with an adjacency matrix $A : V \times V \rightarrow \mathfrak{R}^+$ such that $A_{ij} = d(v_i, v_j)$, the nonnecessarily symmetric distance from v_i to v_j . Such distance matrix can be generated experimentally from a packet reception rate (PRR) matrix as $A_{ij} = -\log(\text{PRR}_{ij})$. The sensor network adjacency matrix is symmetrized, that is, if a link does not have the same packet reception rate (PRR)

in both directions, the two PRR’s of the link are replaced by their product. Then a threshold is chosen such that, if the PRR is greater than the threshold, it is assumed that a link is present, otherwise the link does not exist. The latter defines the edge set E .

3.1. Clustering coefficient

The new (symmetrized) adjacency matrix is used to define the edge set, which is itself used to calculate the clustering coefficient. The clustering coefficient at node a is defined as

$$c(a) = \frac{\text{number of existing triangles with a vertex at node } a}{\text{maximum possible number of triangles}}. \quad (3)$$

The denominator can be computed as

$$\text{maximum possible number of triangles} = \binom{\text{degree}(\text{node } a)}{2}, \quad (4)$$

and degree (node a) is a number of links incident upon node a . The number of existing triangles with a vertex at node a is the number of triples $(ab_k, ab_{k+1}, b_k b_{k+1})$, where ab_k, ab_{k+1} are two edges flowing out of a and $b_k b_{k+1}$ denotes a direct link joining b_k to b_{k+1} .

3.2. Alexandrov angles approach to curvature

Here the network graph is weighted by a symmetric adjacency matrix. The difference between negatively and positively curved surfaces can easily be understood by formalizing the intuitive difference between a saddle and

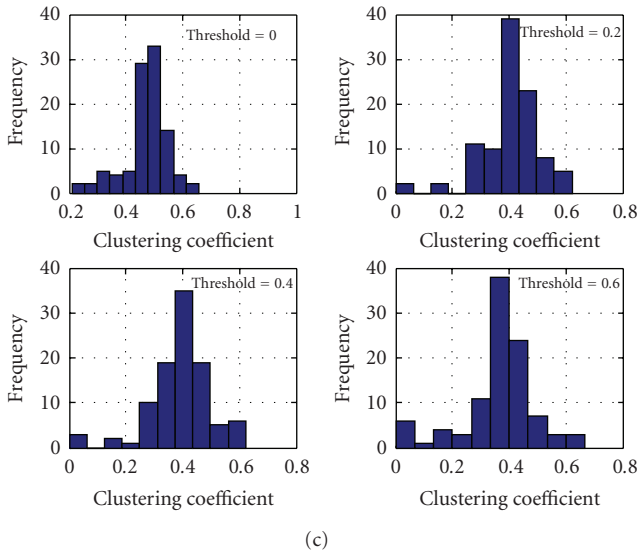
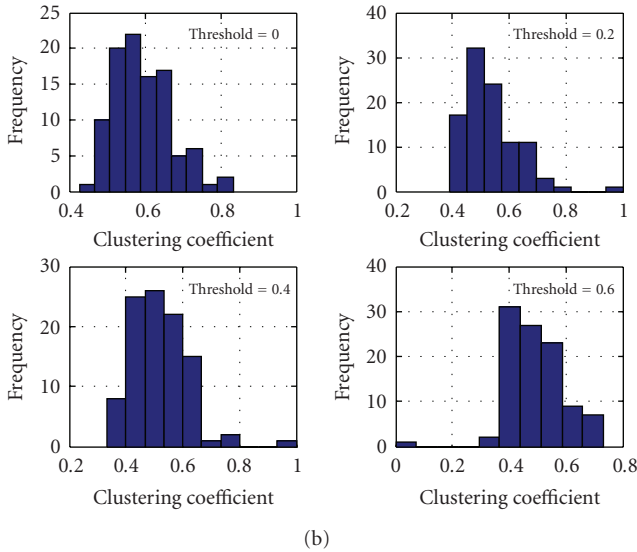
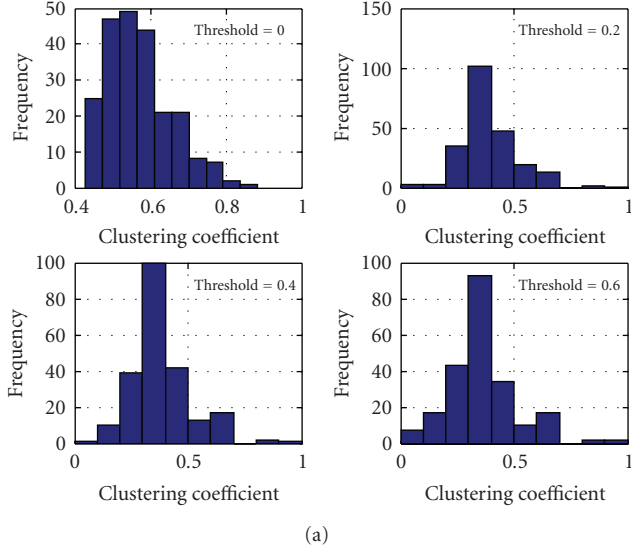


FIGURE 4: Histogram of clustering coefficients: (a) simulation data; (b) real dataset A; (c) real dataset B.

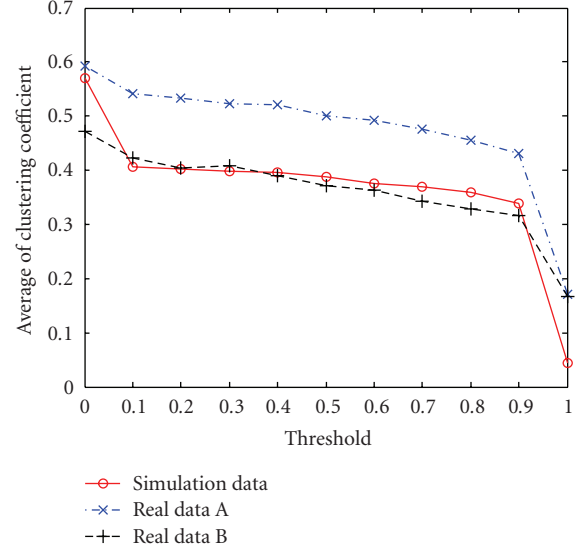


FIGURE 5: Variation of mean of clustering coefficient with threshold.

a sphere. Assume we have a collection of rectilinear triangles ab_kc_k , where $k = 1, 2, \dots, n$. In each such triangle, let $\alpha_k = \angle b_kac_k$ be the angle at the vertex a . α_k is easily computed using the rectilinear law of cosine, in which case it is called *Alexandrov angle* for background Euclidean metric. Let us glue the edge ab_{k+1} along the edge ac_k , $k \leq n-1$, with the understanding that $c_n = b_1$. If $\sum_k \alpha_k < 2\pi$, the resulting surface is a pyramid, and with a little bit of imagination, it looks like a sphere at its apex. The Gauss curvature at the apex a is defined as $\kappa(a) = (2\pi - \sum_k \alpha_k) / \sum_k A(ab_kc_k) > 0$, where $A(\cdot)$ denotes the area functional. If, on the other hand, $\sum_k \alpha_k > 2\pi$, the resulting surface will have a “fold” and hence will look like a saddle. The local curvature at the vertex a is $\kappa(a) = (2\pi - \sum_k \alpha_k) / \sum_k A(ab_kc_k) < 0$.

Consider the more general setting of an N -dimensional Riemannian manifold M . By the definition of a manifold, there exists a local homeomorphism $h : M \rightarrow \mathfrak{R}^N$, $h(a) = 0$. A section through a is defined as $\sigma = h^{-1}(\mathfrak{R}^2)$, where $\mathfrak{R}^2 \subseteq \mathfrak{R}^N$. By the Nash theorem, there is an isometric embedding $f : M \rightarrow \mathfrak{R}^D$ of M in a Euclidean space of dimension $D = N(3N + 11)/2$. In this latter space, $f(\sigma)$ is a surface; its curvature can be computed using the methods of the preceding paragraph, resulting in the sectional curvature $\kappa(a, \sigma)$ of the manifold.

Next, to develop a Riemannian manifold approach to graphs, we need to define the sectional curvature around a vertex a . Clearly, a cyclic ordering of a subset of vertices flowing out of a could be thought of as a section. However, a typical feature of a network graph is that the degree of a vertex is a heterogeneous property, with high variance in the scale-free case. There is thus a need to define the concept of a section consistently across the network, which calls for a minimum number of edges. Here we invoke the Gromov 4-point condition [26], essentially saying that the curvature can be assessed from 4 points, that is, the sectional curvature is defined from 3 edges.

As an illustration consider a tree [27, 28]. Assume the degree of the nodes is three at least. Consider a triple

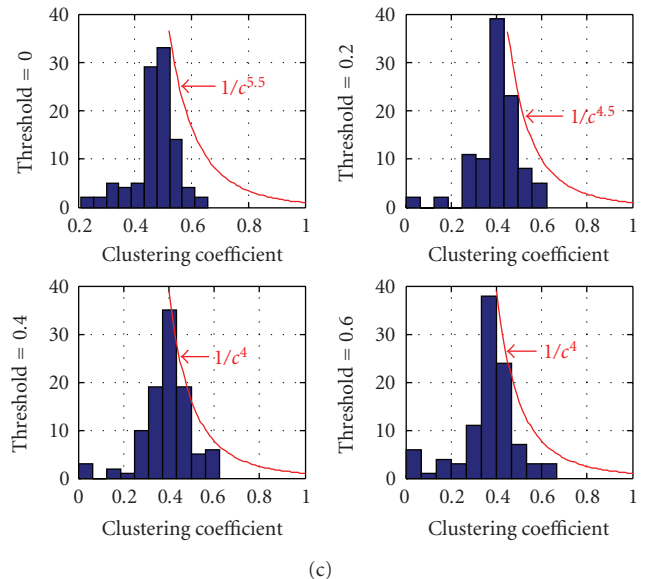
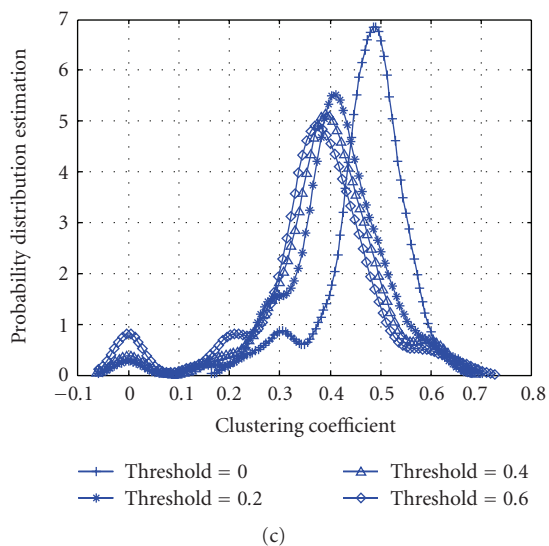
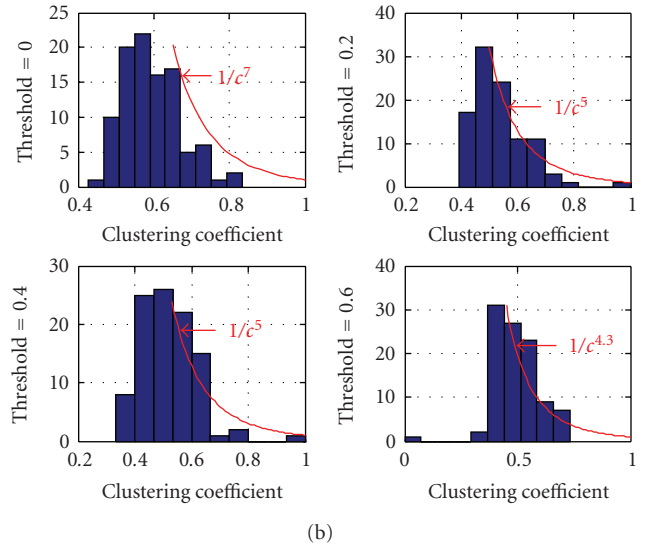
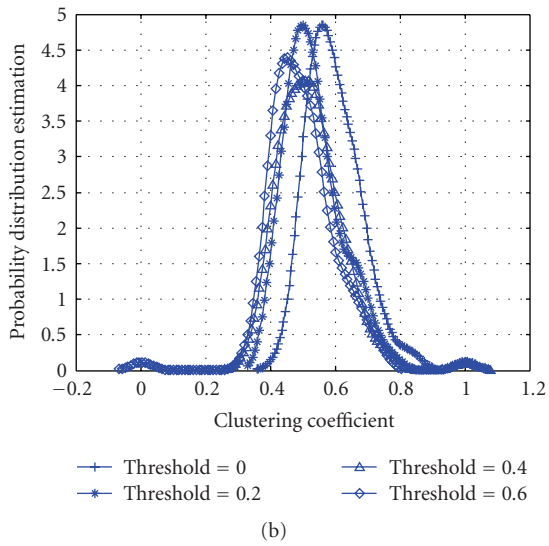
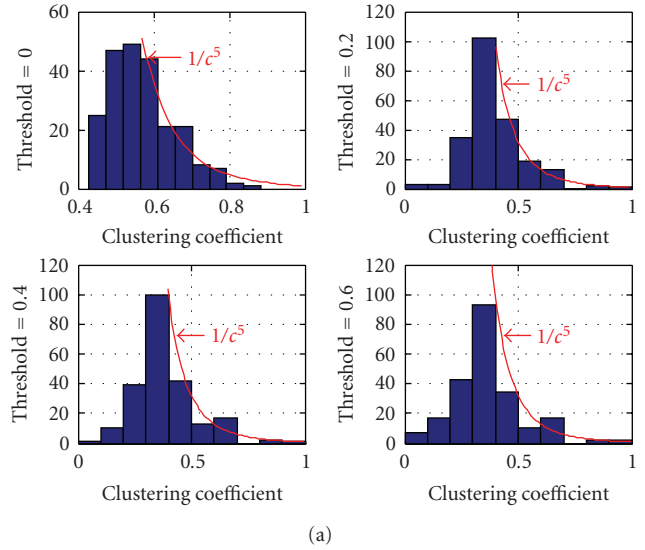
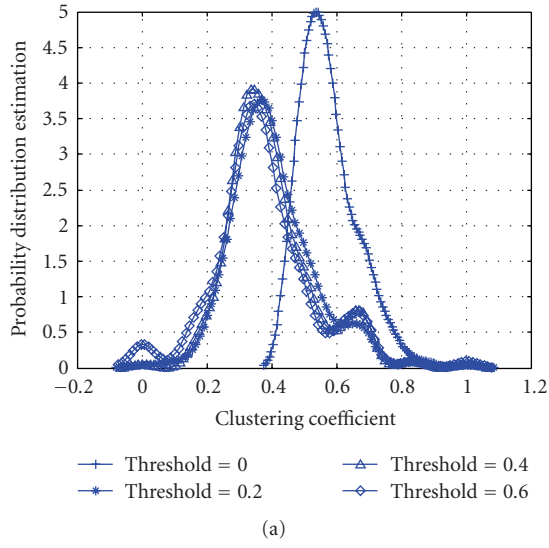


FIGURE 6: Estimated probability distribution of clustering coefficient: (a) simulation data; (b) real dataset A; (c) dataset B.

FIGURE 7: Power law for distribution of clustering coefficient: (a) simulation data; (b) dataset A; (c) dataset B.

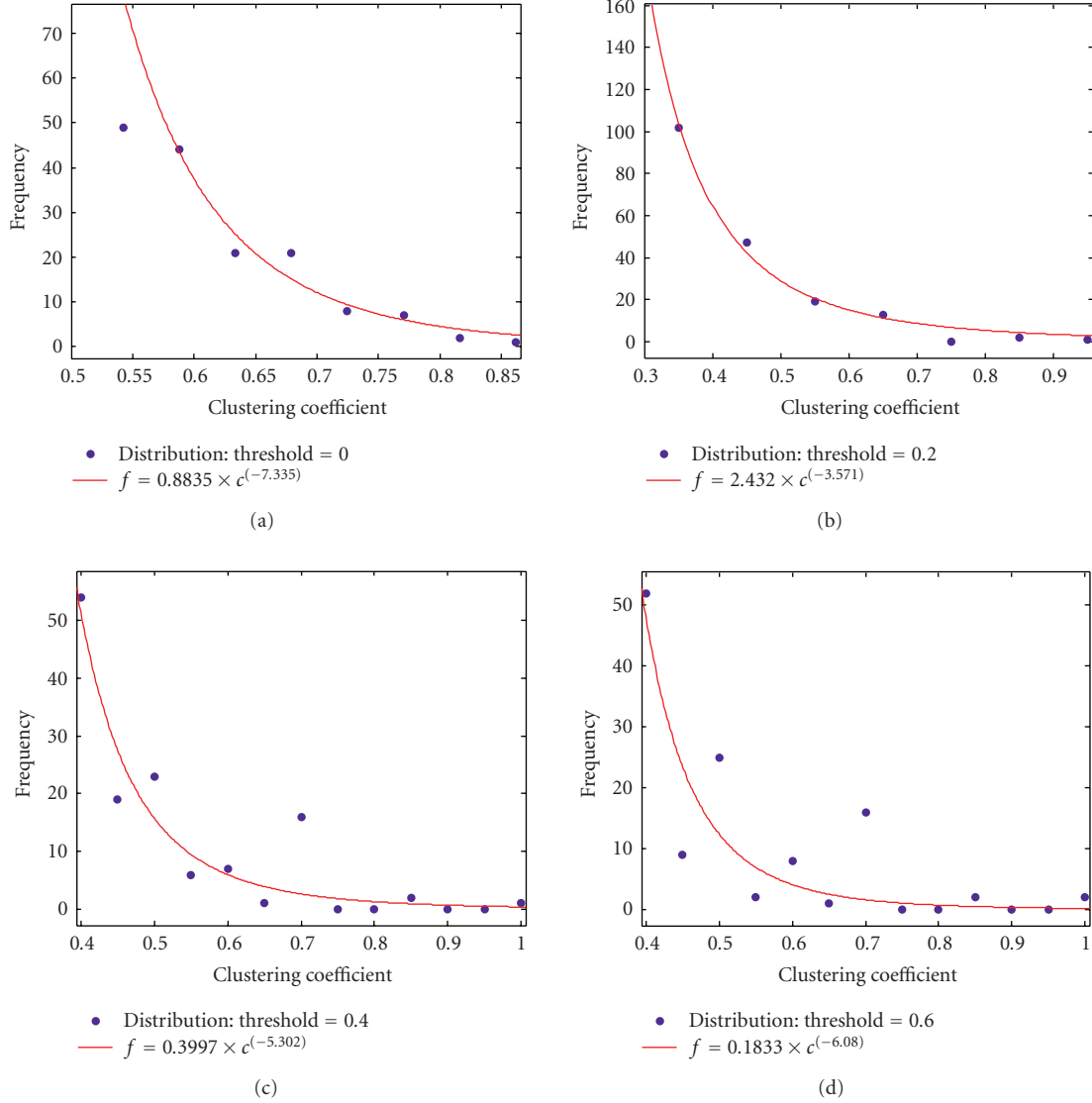


FIGURE 8: Power law for distribution of clustering coefficient (MATLAB): simulation data.

ab_1, ab_2, ab_3 . Clearly, $d(b_k, b_{k+1}) = d(a, b_k) + d(a, b_{k+1})$ from which the rectilinear law of cosines yields $\alpha_1 = \cos^{-1}((d^2(a, b_1) + d^2(a, b_2) - d^2(b_1, b_2))/2d(a, b_1)d(a, b_2)) = \pi$. Hence $2\pi - \sum_{k=1}^3 \alpha_k = -\pi$, but since the area of every single triangle $ab_k b_{k+1}$ vanishes, the curvature is $-\infty$.

3.3. Clustering coefficient approach to curvature

Now, we have to assemble the triangles offered to us by the clustering analysis in such a way as to make sections in which the curvature can be assessed. From the simplified clustering analysis, two vertices are either connected by one single edge, with a weight normalized to 1, or can only be connected by a path of at least two edges, in which case their distance is ≥ 2 . From the clustering analysis around a vertex a , the two edges ab_1, ab_2 either make a triangle or not. In case they make a triangle, b_1, b_2 are directly linked by an edge of weight normalized to one, in which case the triangle is equilateral

with Alexandrov angle $\alpha_k = \pi/3$. The other possibility is that there is no triangle associated with ab_1, ab_2 , which means that b_1, b_2 are connected by a string of at least two edges making a path of length at least 2. Since $d(b_1, b_2)$ is defined as the minimum of all lengths of paths joining b_1, b_2 , the minimum length path is $[ab_1] \cup [ab_2]$; hence $d(b_1, b_2) = 2$. From the metric point of view, $ab_1 b_2$ appears a “flat” triangle and the rectilinear law of cosines yields an Alexandrov angle $\alpha_1 = \pi$.

If the node a is completely clustered, if $N(a)$ is completely meshed, the Alexandrov angles are all equal to $\pi/3$ and $2\pi - \sum_{k=1}^3 \alpha_k = \pi > 0$, and the curvature is positive. If the node has vanishing clustering coefficient, if $N(a)$ is star connected, the Alexandrov angles are all equal to π and $2\pi - \sum_{k=1}^3 \alpha_k = -\pi < 0$, and the curvature is negative.

It should be noted that an ad hoc wireless mesh network need not have positive curvature, unless it is fully meshed. As a counterexample, observe that a planar network of node

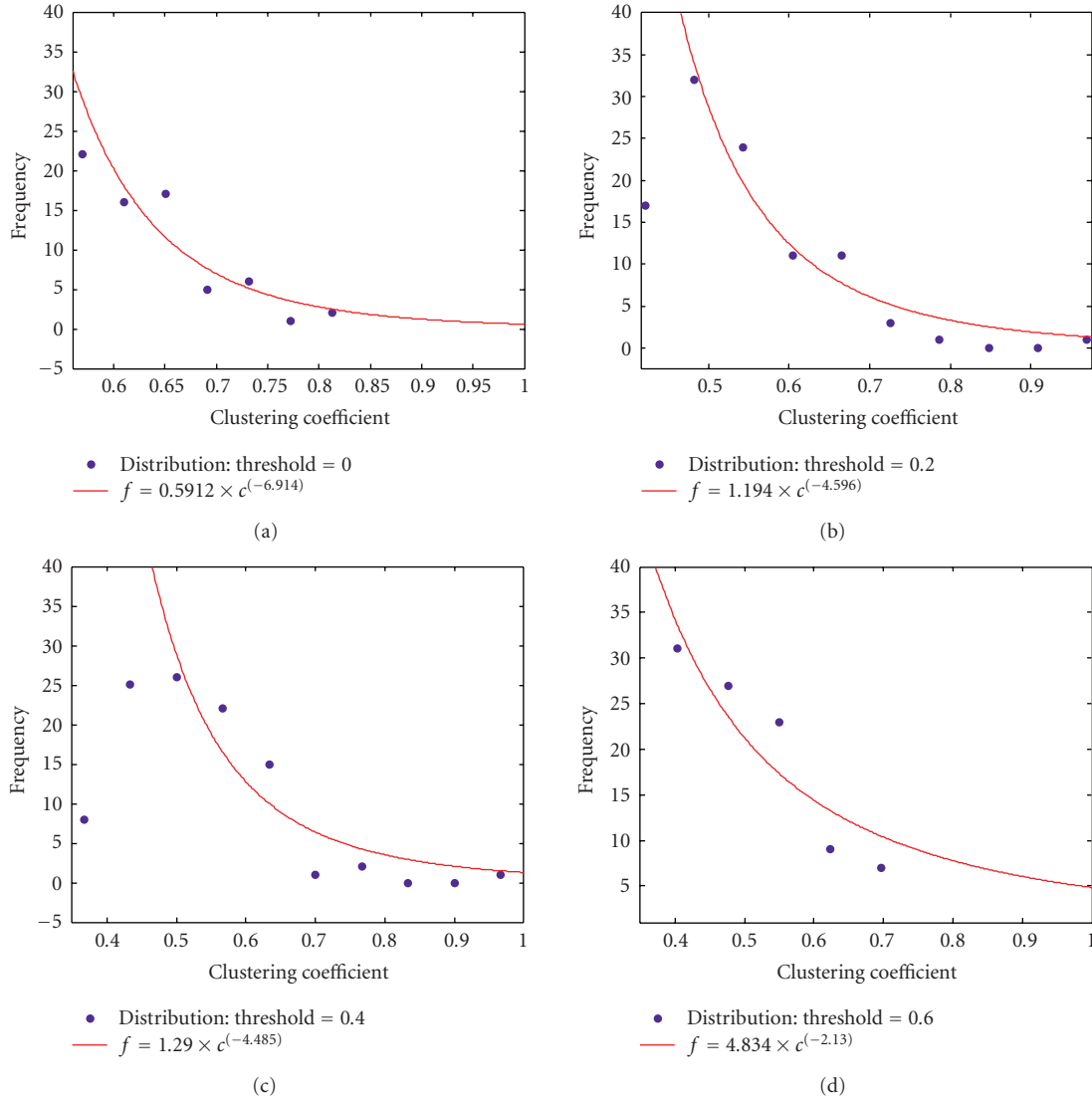


FIGURE 9: Power law for distribution of clustering coefficient (MATLAB): dataset A.

degree uniformly greater than 6 has uniformly negative curvature, even though it would be qualified as “meshed.”

4. SIMULATION/EXPERIMENTAL SETUP

4.1. Simulation data

The virtual network consists of 225 nodes in a grid topology, where the grid size is 1 meter. Simulation was based on the following environmental parameters, which were measured on the aisle of the third floor in the Electrical Engineering Building in the University Park Campus of the University of Southern California (USC):

- (i) path loss exponent = 3.0,
- (ii) shadowing standard deviation = 3.8,
- (iii) path loss reference = 55.0 dB (for a distance of 1 meter),

(iv) radio parameters: these parameters characterize an MICA2 mote using noncoherent FSK modulation with Manchester encoding and a frame length of 52 bytes,

- (v) output power = -20 dBm,
- (vi) standard deviation of output power = 1.2 dB,
- (vii) noise floor = -90 dBm,
- (viii) standard deviation of noise floor = 0.7 dB.

The connectivity matrix for the topology is the prrMatrix.mat MATLAB file available at <http://ceng.usc.edu/~anrg/downloads.html>

(3. Realistic Wireless Link Quality Model and Generator). The nodes are numbered in a right-top approach, where the node at (0,0) is node 1, the node at (14,0) is node 15, the node at (0,1) is node 16, and so forth.

Figure 3 shows a random instance of the connectivity graph for the given topology. Figure 3(a) has the following

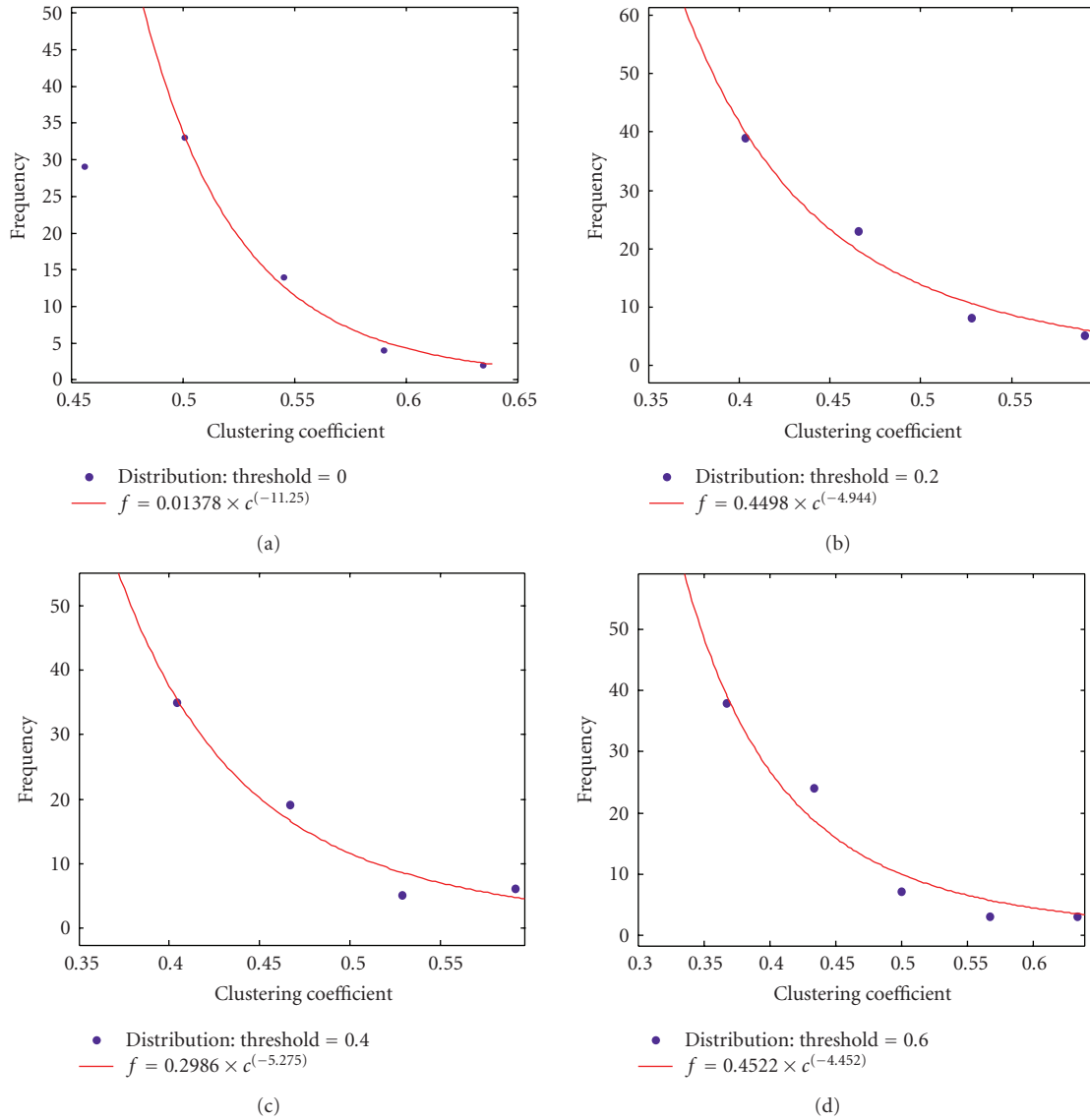


FIGURE 10: Power law for distribution of clustering coefficient (MATLAB): dataset B.

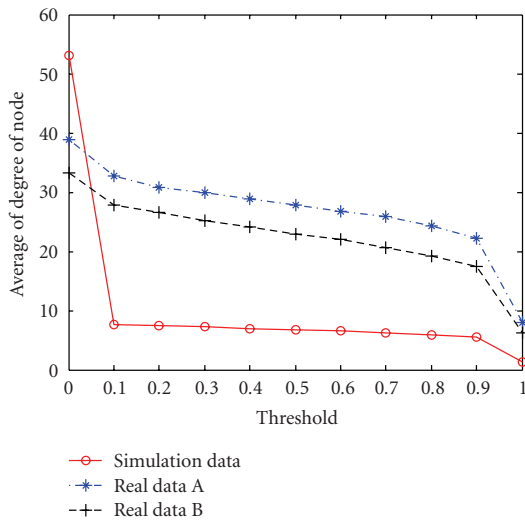


FIGURE 11: Variation of degree of the node with threshold.

convention for the links (edges). Recall that this is a directed graph. The direction of the edges is not shown and instead the following convention is used for illustration purposes.

- (i) If a pair of nodes (A, B) has a packet reception rate (PRR) above 0.9 in both directions (i.e., $A \rightarrow B$ and $B \rightarrow A$), then the edge is drawn as a full line. In this case, the link can be considered as symmetric.
- (ii) If a pair of nodes (A, B) has a PRR above 0.3 in both directions, but one or both directions are below 0.9, then the edge is drawn as a dotted line. This link can be considered as asymmetric.
- (iii) If a pair of nodes (A, B) has a PRR below 0.3 in at least one direction, then the edge is not drawn. However, in Figure 3(b) (zoom in), it is plotted as a dotted red line. These links can be considered as highly asymmetric or very weak.

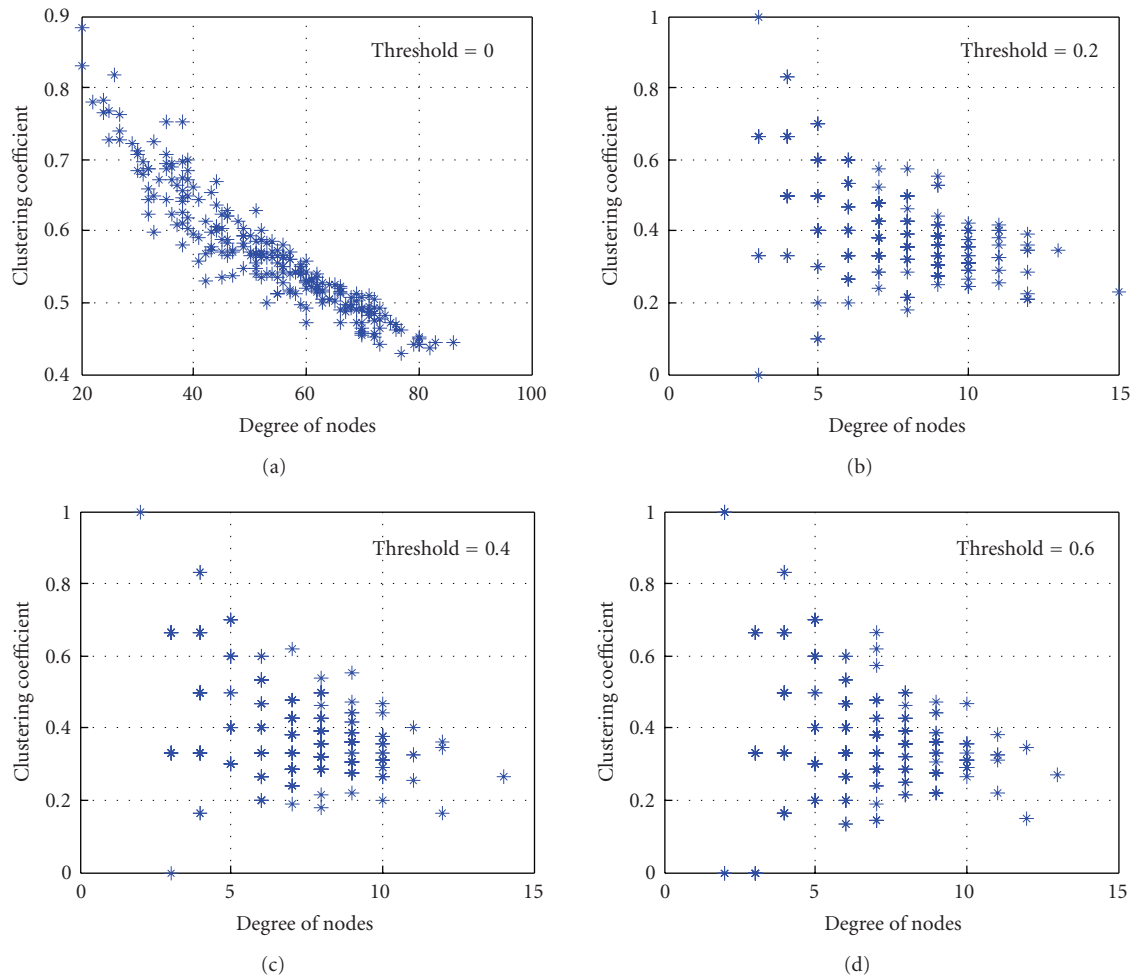


FIGURE 12: Variation of clustering coefficient with degree of nodes: simulation data.

4.2. Real data

Two other sets of data, those real, are also analyzed. These are two representative deployments of 100 nodes placed on the ground in an indoor basketball court at USC. The deployments consisted of a mix of 59 moteiv tmote sky wireless devices and 41 crossbow micaz wireless devices. Both devices have the same IEEE 802.15.4 radio transceiver (chipcon CC2420), but as evident in the results, the tmote sky nodes have a significantly higher transmission range. This is attributable to differences in antenna design (external wire versus printed-on-board). The key difference between the two deployments is the higher internode spacing in one (10 ft apart versus 6 ft apart).

This real network deployment data is also made available online at <http://ceng.usc.edu/~anrg/downloads.html> (6. Measurement of pairwise PRR values from two real 100-node rectangular grid deployments).

5. RESULTS

After computing the clustering coefficients for all nodes of the graph, their distribution is plotted and the best fitting probability distribution, estimated using a kernel smoothing

method, is derived. Also, using Curve Fitting Toolbox in MATLAB, the power-law behavior of the network clustering distribution is tested for some values of threshold. It should be reminded that the analysis of Section 2 singled out the clustering as a degree-independent factor contributing to congestion. The experimental analysis of Section 5.2.3 will confirm the near independence of the clustering on the degree. Hence the power law behavior of the clustering coefficient should *not* be confused with the traditional heavy-tailed phenomenon.

5.1. Probability distribution of clustering

The clustering coefficient for each node is calculated. This has been done by symmetrizing the adjacency matrix and considering different values of the threshold. The distribution of the clustering coefficients for the whole graph is shown in Figure 4 for simulated data, real data A, and real data B.

The clustering coefficient varies with the threshold. The average values of the clustering coefficients for various thresholds are listed in Table 1 and the graphical representation is found in Figure 5.

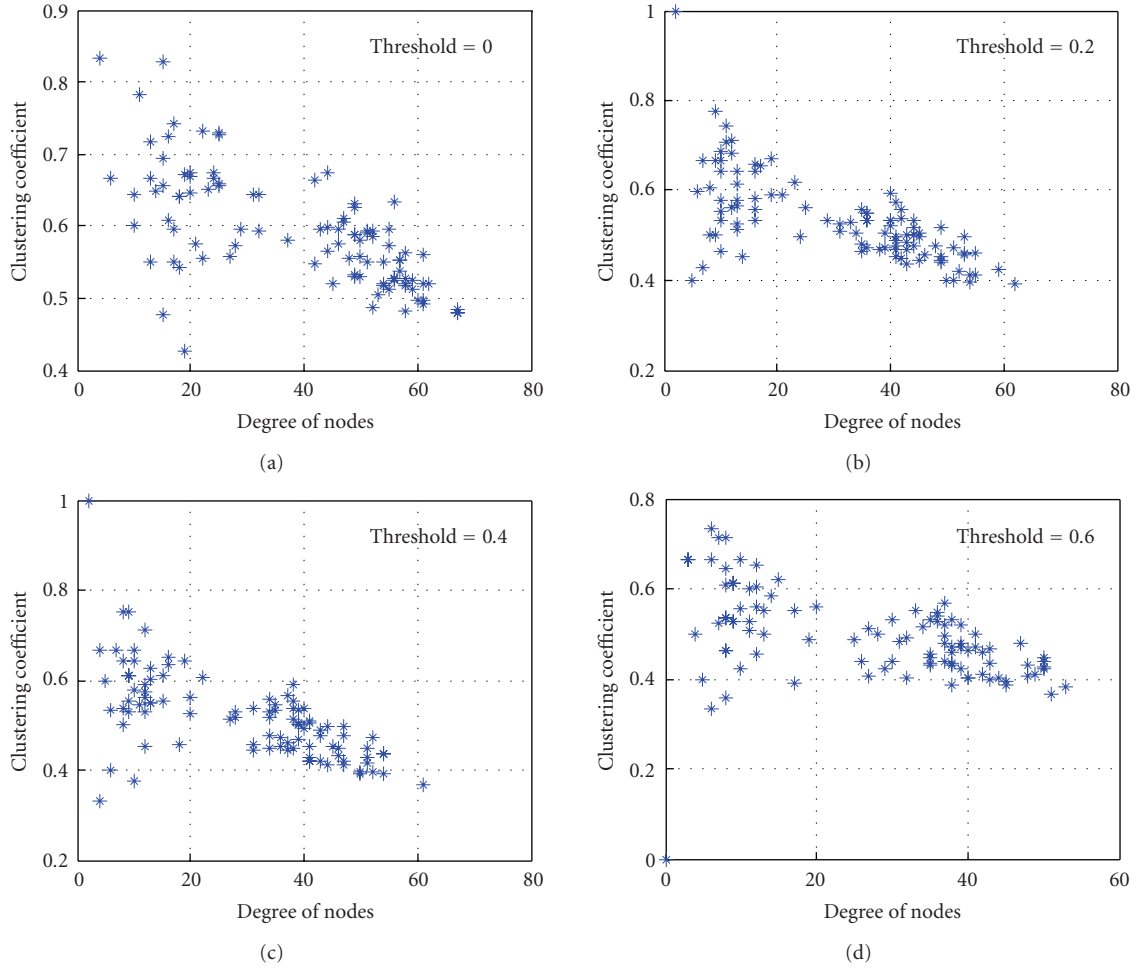


FIGURE 13: Variation of clustering coefficient with degree of nodes: dataset A.

The mean of the clustering coefficient decreases as the threshold increases. This appears to be a specific property of the wireless protocol, as there is no way to predict how in general the clustering coefficient of a weighted graph would vary with the threshold. Indeed, by increasing the value of the threshold, the degree of the nodes decreases (as it is shown later) and hence both the numerator and the denominator of $c(a)$ decrease. For example, if we set threshold to zero, that is, considering all links even the weakest ones in the network, the average of the clustering coefficient (for symmetrized adjacency matrix) would be equal to 0.5702, 0.592, and 0.47118 for simulated data, real data A, and real data B, respectively.

For example, if we set the threshold to zero, that is, considering all links even the weakest ones in the network, the average of the clustering coefficient (for symmetrized adjacency matrix) would be equal to 0.5702, 0.592, and 0.47118 for simulated data, real data A, and real data B, respectively.

The probability distribution estimation for the clustering coefficient is done using a kernel smoothing method in MATLAB. The graphs of Figure 6 show the variation of the

TABLE 1: Average of clustering coefficient versus threshold.

Threshold	Average of clustering coefficient		
	Simulation data	Real data A	Real data B
0.0	0.5702	0.59200	0.47118
0.1	0.40637	0.54088	0.42361
0.2	0.40132	0.53346	0.40419
0.3	0.39763	0.52327	0.40871
0.4	0.39513	0.52027	0.38939
0.5	0.38846	0.50095	0.37233
0.6	0.37657	0.49171	0.36372
0.7	0.36929	0.47562	0.34293
0.8	0.35847	0.45492	0.32908
0.9	0.33943	0.43005	0.31704
1.0	0.04444	0.16992	0.16729

probability distribution with the threshold for all three sets of data.

For simulated data and real dataset A, the probability distribution is more right skewed whereas it turns out

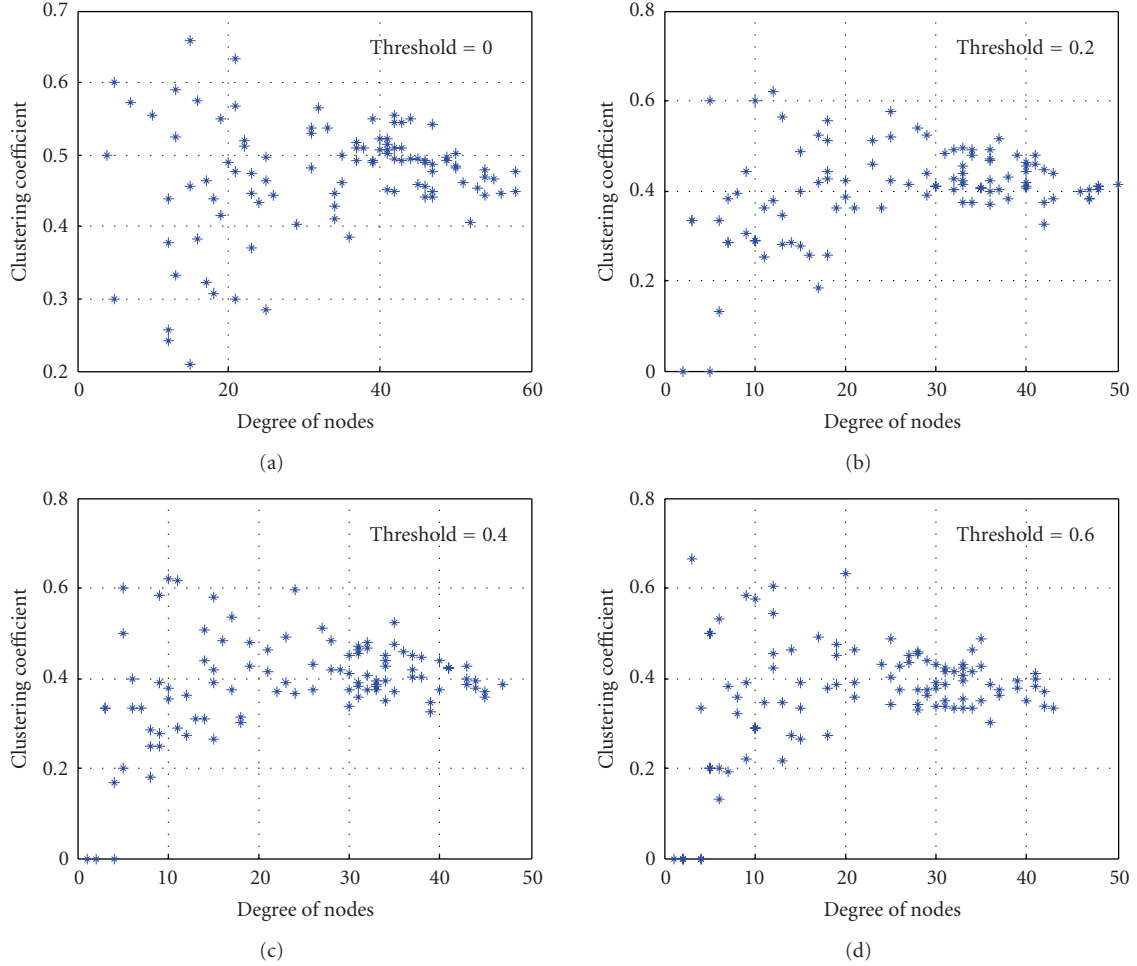


FIGURE 14: Variation of clustering coefficient with degree of nodes: dataset B.

to be left skewed for real dataset B. For a value of the threshold equal to zero, these curves have maximum means, hence pointing toward positive curvature. This result is not surprising, since decreasing the threshold creates more and more links (of poor PRR's), and tends to make the graph fully meshed, hence positively curved. By increasing the value of the threshold, it is seen that the mean value decreases and the variance increases. Therefore, for very high threshold, the graph tends to be negatively curved and the clustering distribution tends toward becoming heavy tailed.

5.2. Degree-independent power-law behavior of clustering

5.2.1. Clustering coefficient distribution

Considering the clustering coefficient as a random variable, the power-law behavior of its density is investigated. This is the issue of whether the probability density could be fitted by

$$f_c = \beta c^\alpha, \quad (5)$$

where β and α are constants, c is the clustering coefficient, and f_c represents the density at clustering coefficient c . The above behavior is investigated in two different ways.

First, by trial and error, the best fit could be found as $f_c = 1/c^5$ for simulation data, which, as seen in Figure 7(a), is almost verified for all threshold values. For real dataset A, α varies from 4.3 to 7 and for real dataset B, from 4 to 5.5 (see Figures 7(b) and 7(c)).

As it can be seen, this curve fitting works best at low threshold and, on the other extreme, it does not match the distribution of the clustering coefficients as the threshold increases.

The second method utilizes the Statistic Toolbox of MATLAB to estimate with a confidence level the exponent in the tail. The results are plotted in Figures 8, 9, and 10 for simulated data, real dataset A, and real dataset B, respectively.

These statistically more reliable results are consistent with those of the first method. In all cases, the absolute value of α is greater than 3. One can conclude that the tail of distribution of the clustering coefficient obeys a Pareto law, but is not exactly heavy tailed, especially for low values of the threshold (see Table 2).

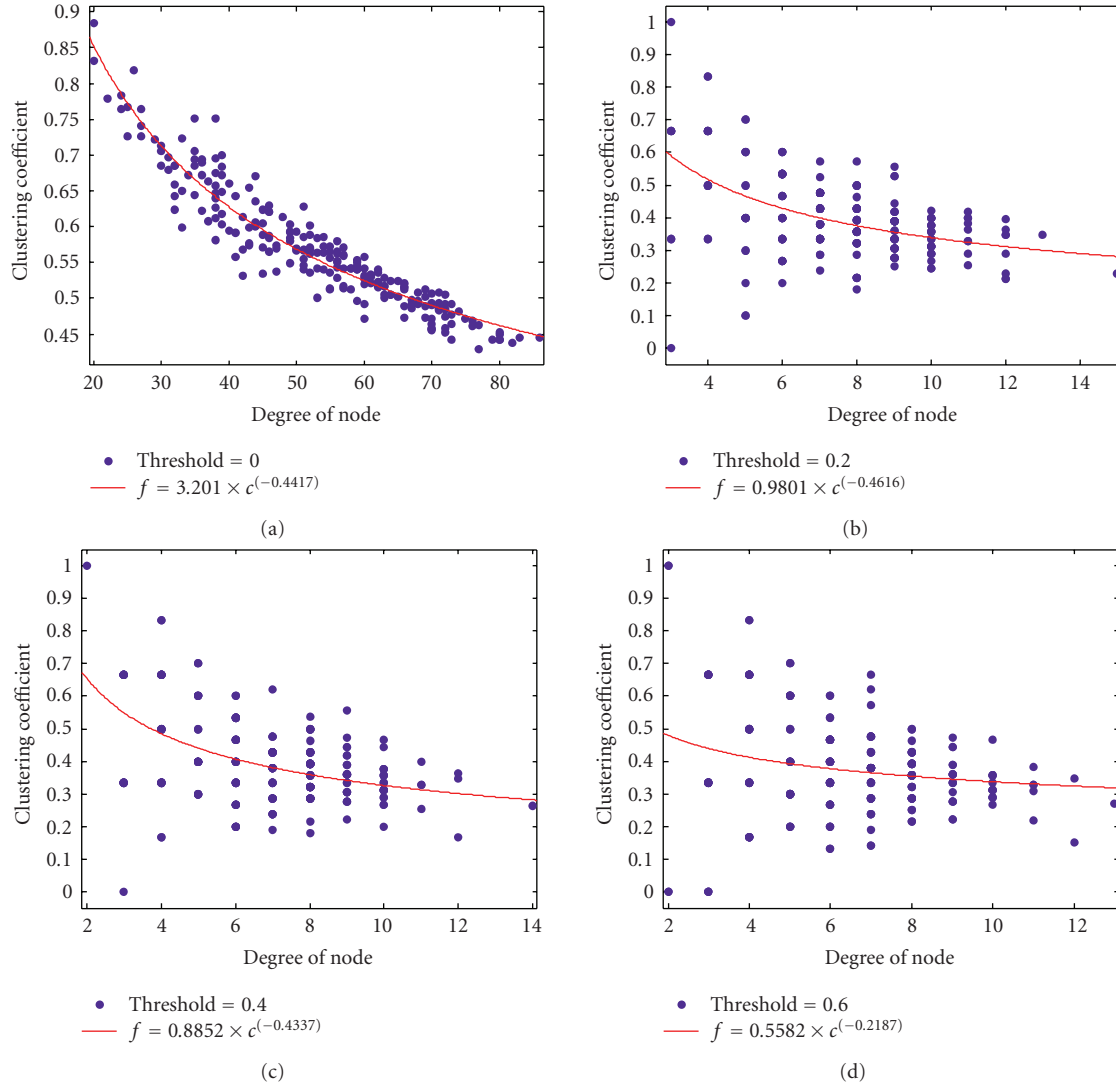


FIGURE 15: Power law for distribution of clustering coefficient versus degree of nodes: simulation data.

5.2.2. Probability distribution of the degree of nodes

The degree of each node is calculated for different values of the threshold. In all cases, except for a threshold equal to zero, the degree of the nodes is much less than the total number of nodes, n .

The average of the degree of the nodes for each value of the threshold is shown in Table 3 and a graphical representation can be found in Figure 11 for all three datasets. The average of the degree of the nodes varies almost linearly with the threshold between 0.1 and 0.9. But the bigger conclusion drawn from this figure is that, as long as the threshold is increased, the degree of the nodes decreases. This can be justified on the ground that, as the threshold is increasing, we remove some poor quality links, while keeping the good ones, which decreases the degree.

5.2.3. Clustering coefficient versus degree

In this part, the subset of clustering coefficients of nodes of a fixed degree is considered as a function of the degree.

The graphs in Figures 12, 13, and 14 show the distribution of the clustering coefficient versus the degree for different values of the threshold. In Figures 15, 16, and 17, we inspired ourselves from [15] and plotted the best power law fit of the clustering versus the degree using the Curve Fitting Toolbox of MATLAB (see Table 4).

We take the simulated data of Figure 15 as benchmark case study. It is quite obvious that at zero threshold (positive curvature), we have a well-defined power law (α negative enough) whereas at high threshold (negative curvature), the power law is less marked ($\alpha \uparrow 0$) and in fact the dependency of the clustering on the degree becomes almost constant. From Table 4, it transpires that as we proceeded from low to high threshold, the relative size of the confidence interval for α increases, hence statistically the analysis is slightly less reliable. The same trend can be seen for dataset A. The overall trend for dataset B is more toward constancy, which can be justified on the ground that the curvature is more negative for this dataset.

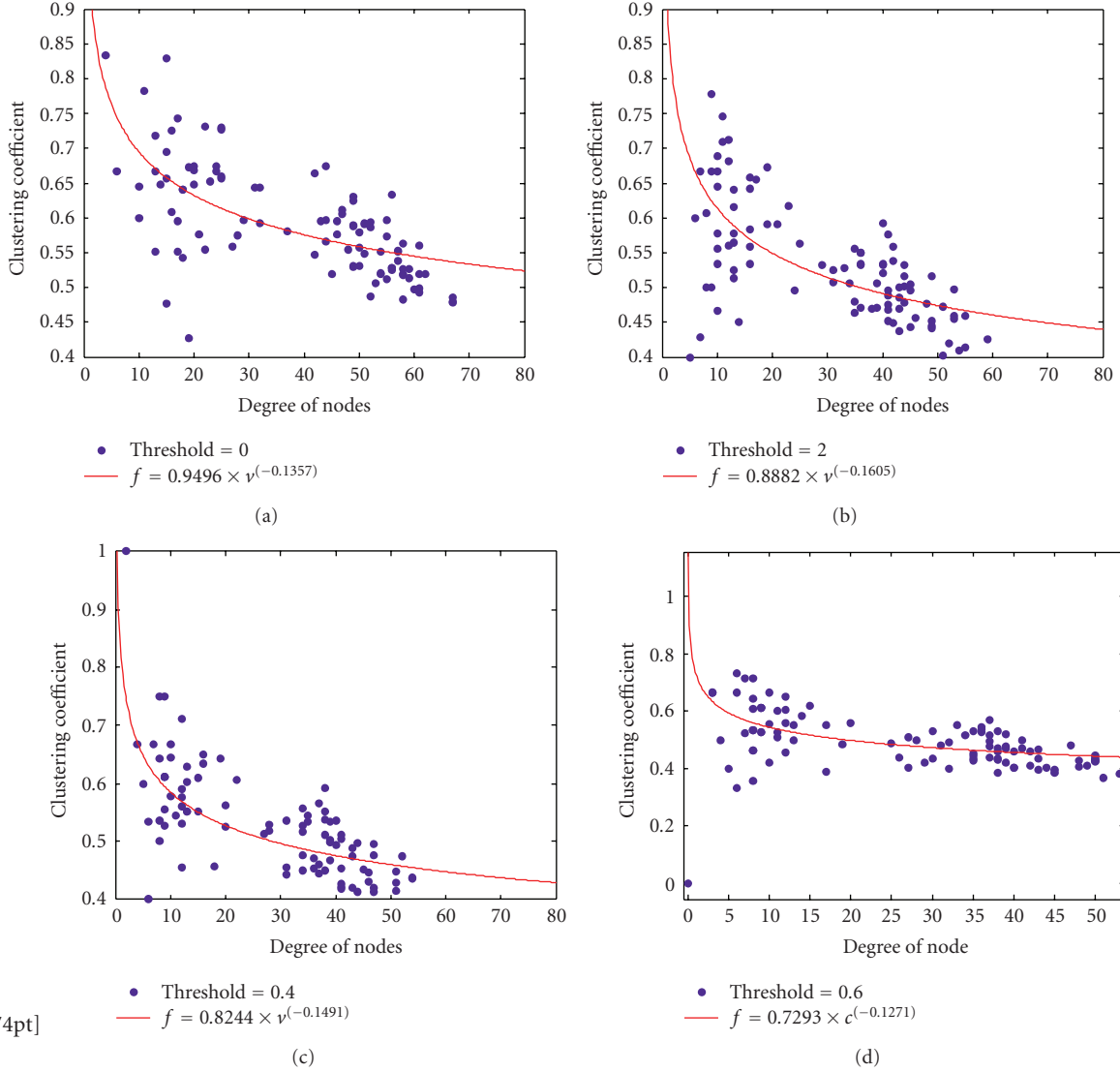


FIGURE 16: Power law for distribution of clustering coefficient versus degree of nodes: dataset A.

TABLE 2: Confidence intervals for parameters of power law distribution of clustering coefficients (confidence level = 95%).

$f_c = \beta c^\alpha$		Threshold			
Confidence interval		0.0	0.2	0.4	0.6
Simulation data	β	0.8835 (-.224, 1.991)	2.432 (.7948, 4.07)	0.3751 (-.291, 1.041)	0.1833 (-.3859, .731)
	α	-7.335 (-9.867, -4.803)	-3.571 (-4.244, -2.897)	-5.019 (-6.929, -3.109)	-6.08 (-9.182, -2.303)
Real data A	β	0.5912 (-1.078, 2.26)	1.194 (.07214, 2.351)	1.29 (-.8332, 3.414)	4.834 (-2.975, 12.64)
	α	-6.914 (-13.13, -.6997)	-4.596 (-5.998, -3.193)	-4.485 (-7.076, -1.893)	-2.135 (-4.195, -.0761)
Real data B	β	0.01378 (-.02385, .0514)	0.4498 (-.8301, 1.709)	0.2986 (-.7255, 1.298)	0.4522 (-.6654, 1.57)
	α	-11.25 (-15.28, -7.227)	-4.944 (-8.309, -1.631)	-5.275 (-9.397, -1.252)	-4.452 (-7.038, -1.866)

It therefore appears that positive curvature can be characterized by a well-defined power law for the clustering coefficient versus the degree. This observation is consistent with [15], where for the positively curved World Wide Web $\alpha \approx -1$. Negative curvature on the other hand can be

characterized by a “flatter” and statistically somewhat less reliable clustering versus degree curve.

The fact that in negative curvature the clustering is nearly constant relative to the degree provides experimental confirmation of our earlier assertion that clustering

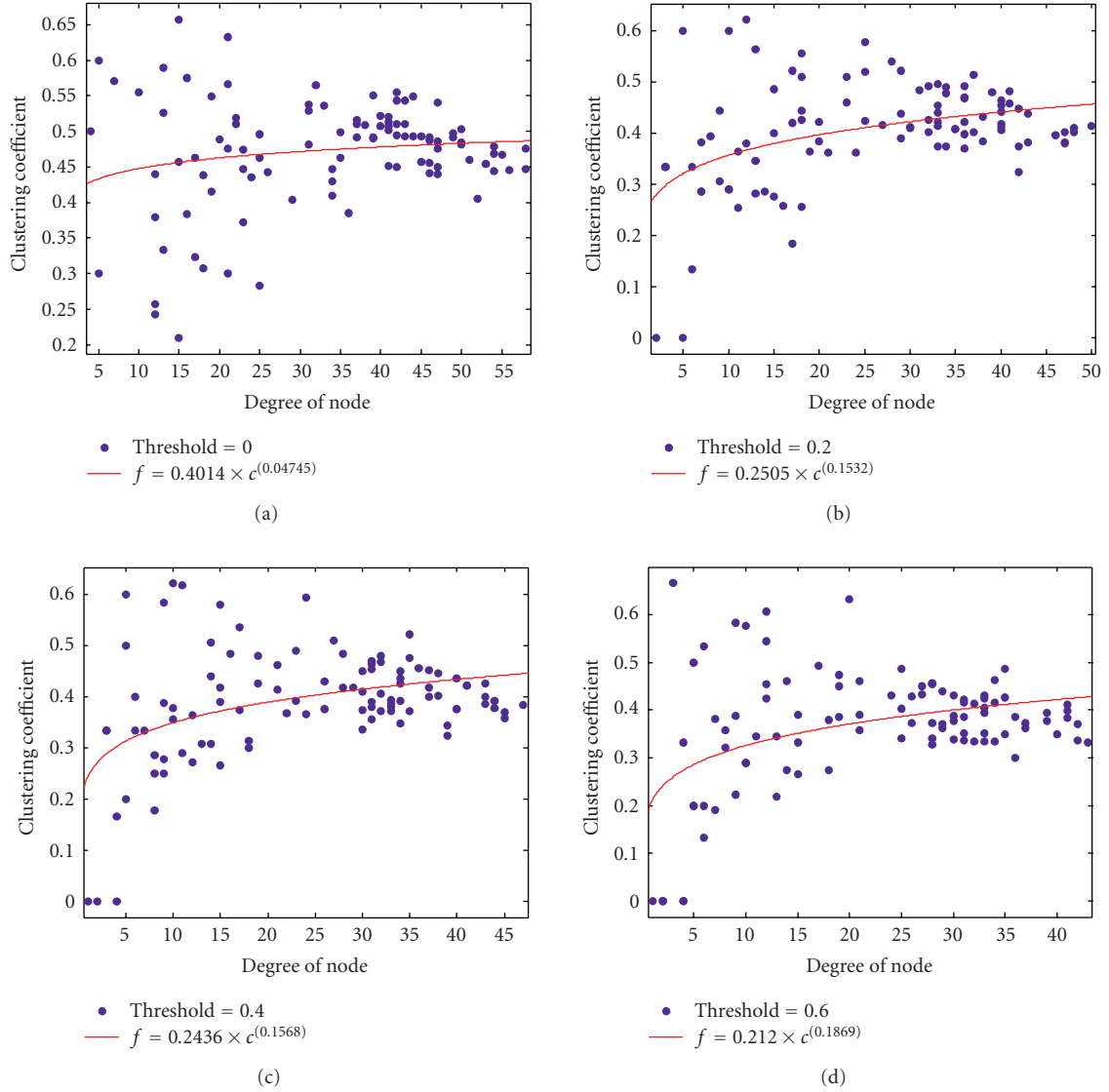


FIGURE 17: Power law for distribution of clustering coefficient versus degree of nodes: dataset B.

TABLE 3: Average of degree of nodes and threshold.

Threshold	Average of degree of nodes		
	Simulation data	Real data A	Real data B
0.0	53.049	38.82	33.2
0.1	7.7156	32.64	27.76
0.2	7.4933	30.84	26.52
0.3	7.2622	29.98	25.2
0.4	6.9867	28.88	24.06
0.5	6.7556	27.82	22.84
0.6	6.5422	26.76	21.98
0.7	6.2311	25.80	20.70
0.8	5.8933	24.28	19.28
0.9	5.5111	22.20	17.38
1.0	1.2622	8.04	6.18

is a degree-independent factor contributing to congestion.

5.3. Spatial distribution of clustering

Figures 18–20 show the spatial distribution of the clustering coefficients across the network. It is quite obvious that, for low threshold, the clustering is nearly constant, whereas, at high threshold, it is much more heterogeneous. The homogeneity at low threshold can be justified on the ground that taking all links into consideration makes the wiring homogeneous. At high threshold, there are isolated areas of high clustering, which might be called “cores.” As shown in the figures, the core of the network (i.e., nodes with higher clustering coefficients) is almost in the center of the graph, and the areas of negative curvature (nodes with low clustering coefficient) are at the periphery. This is more visually obvious from the simulation data. For real networks

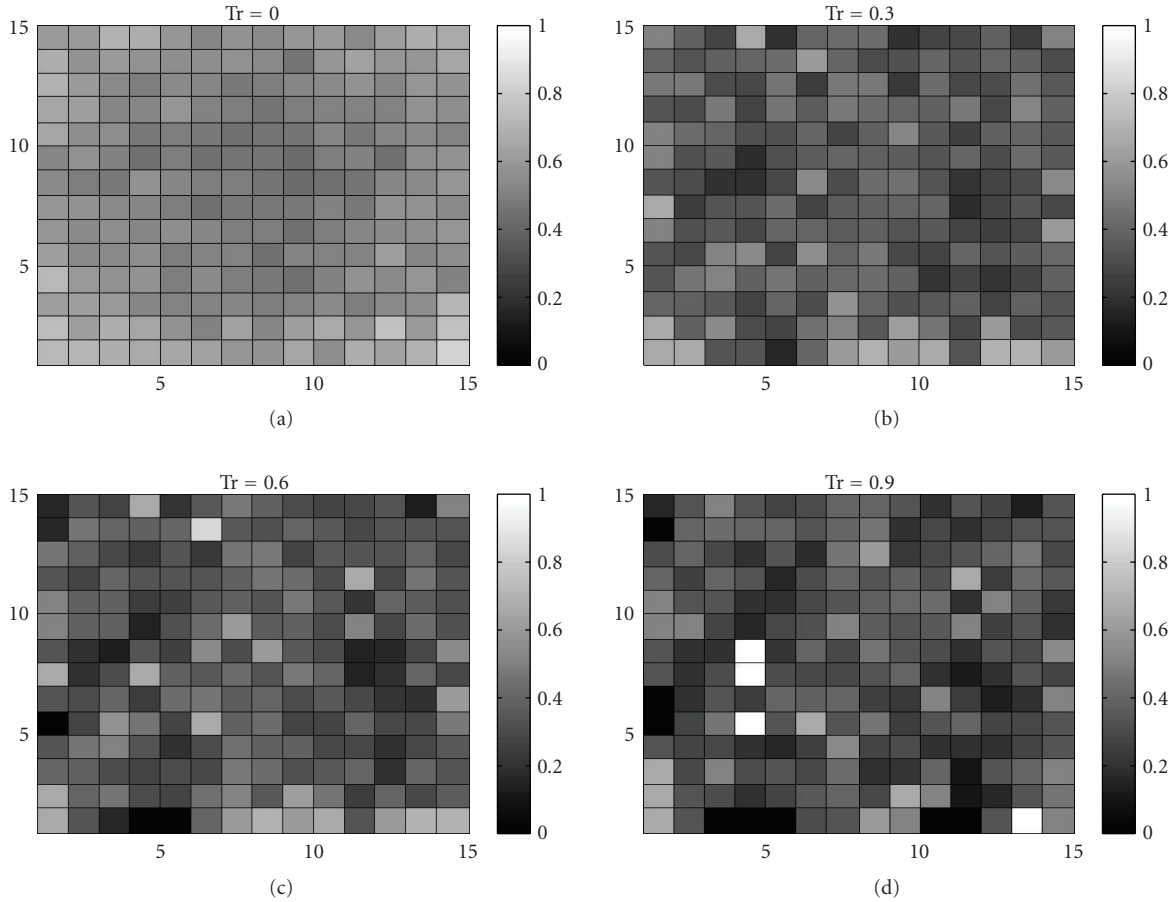


FIGURE 18: 3D illustration of spatial distribution of clustering coefficient across the graph: simulation data.

TABLE 4: Confidence intervals for parameters of clustering versus degree power law (confidence level = 95%).

$f_c = \beta c^\alpha$		Threshold			
		0.0	0.2	0.4	0.6
Simulation data	β	3.201 (2.967,3.435)	0.9801 (.7958,1.164)	0.8852 (.7216,1.049)	0.5582 (.4194, .697)
	α	-0.4417 (-.4607, -.4227)	-0.4616 (-.5614, -.3617)	-0.4337 (-.5358, -.3317)	-0.2187 (-.3572, -.0802)
Real data A	β	0.9496 (.8487,1.051)	0.8882 (.7942, .98230)	0.8244 (.7355, .9134)	0.7293 (.6518, .8067)
	α	-0.1357 (-.1664, -.105)	-0.1605 (-.1942, -.1267)	-0.1491 (-.1844, -.1138)	-0.1271 (-.1624, -.0918)
Real data B	β	0.4014 (.3202, .4827)	0.2505 (.1892, .3118)	0.2436 (.1852, .3019)	0.212 (.1508, .2731)
	α	0.04745 (-.0113, .1062)	0.1532 (.07867, .2278)	0.1568 (.08174, .2319)	0.1869 (.09472, .2791)

A and B, since the networks consist of two different types of sensors, two cores in the center of each group are observed while the nodes with negative curvature are located at boundaries.

5.4. Clustering curvature versus threshold

The relationship between the clustering coefficient and the curvature of the graph can be established as follows: if the clustering coefficient of the node is closer to 1, then the curvature is positive; otherwise, if it is closer to zero, the curvature is negative. Looking at Table 1, one can see that, as the

value of the threshold increases, the average of the clustering coefficients for the whole graph decreases, pointing toward negative curvature. This can be explained by the fact that, under increasing threshold, only the strong links are taken into consideration and the same strong links interconnect in a tree-like pattern, the perfect example of a negatively curved graph. On the other hand, under diminishing threshold, the clustering coefficient increases, pointing toward positive curvature. Again, this is not surprising, since under small threshold nearly all links are taken into consideration, the graph tends to a *fully* meshed one, the perfect example of a positively curved graph.

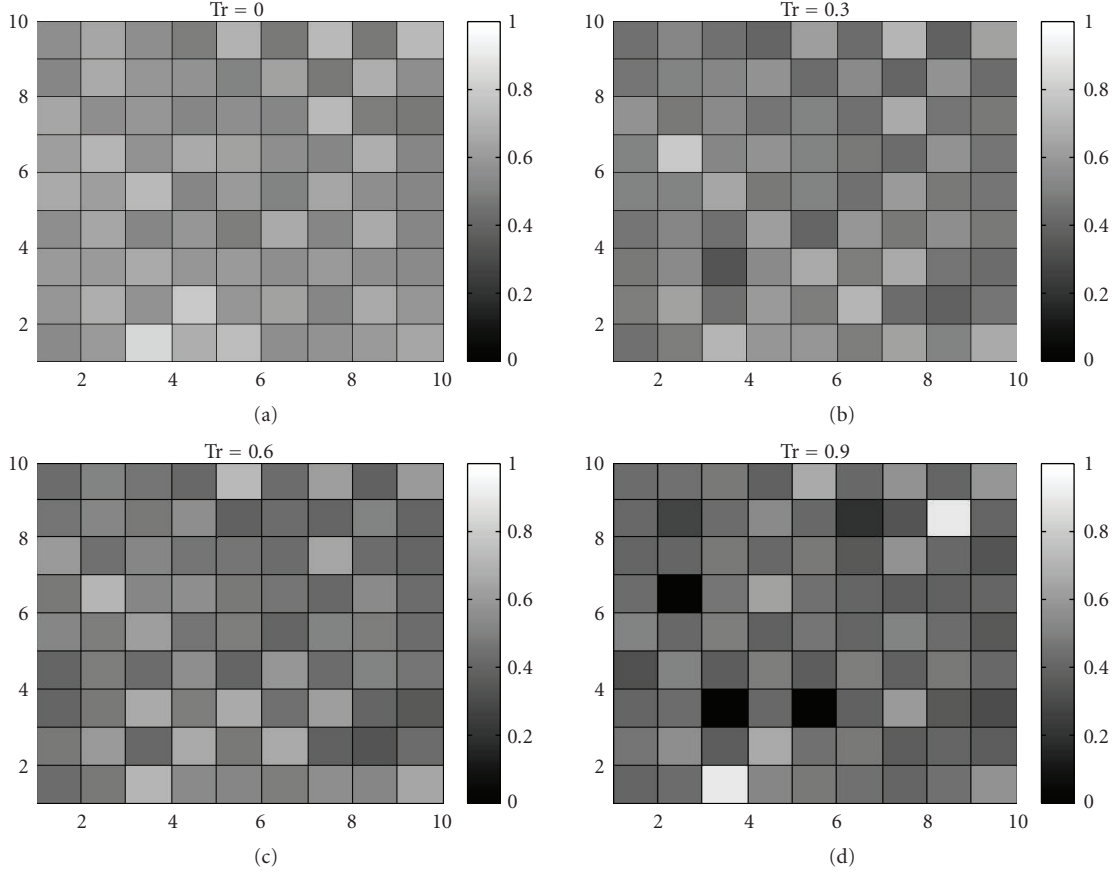


FIGURE 19: 3D illustration of spatial distribution of clustering coefficient across the graph: dataset A.

6. LARGE-SCALE VERSUS SEMILOCAL CONGESTION INTERPRETATION

As said, clustering is a semilocal approach to an inherently large-scale congestion problem. Here we formulate the genuine large-scale congestion issues and illustrate them on the “dataset 06A,” which is generated from a real wireless sensor network, deployed in an indoor basketball court at USC, involving 100 nodes 6 feet apart. Next, we will compare the exact large-scale analysis with the semilocal clustering approach.

Given a network graph, we define the *betweenness* of the node a , $\beta(a)$, to be the number of geodesics passing through a . Betweenness is a pure mathematics concept [29], introduced in disguise in tree networks in [30], and quite explicitly utilized in Protein Interaction Network (PIN) [31]. The *inertia* of the network relative to the vertex a is defined as $\phi(a) = (1/2) \sum_{v \in V(G)} d^2(a, v)$. A *center of mass* or *centroid* of the network is defined as a vertex relative to which the inertia is minimum. Our general large-scale conjecture is that for a negatively curved network graph, the vertex of heaviest congestion (of maximum betweenness) occurs at the centroid (vertex of minimum inertia). Proofs in some specific setups are available in [12]. For a positively curved network, the inertia tends to be uniform and the traffic tends to be uniformly distributed across the network [12].

It is easy to illustrate the conjecture in the simple setting of a graph of vertex set $a \cup N(a)$ along with the concept of clustering. If we include in the traffic of a the traffic transiting through a as well as the traffic departing from a and arriving to a , the total traffic under normalized demand is $\Lambda_t(a) = 2\deg(a) + \binom{\deg(a)}{2}(1 - c(a))$. On the other hand, for $b \in N(a)$, $\Lambda_t(b) = 2\deg(b)$. Hence $\Lambda_t(a) \geq \Lambda_t(b)$. Regarding the inertia, it is easily seen that $\phi(a) \leq \phi(b)$. Thus, as the conjecture says, traffic and inertia are going in opposite directions. More specifically, traffic is maximum at a , $\Lambda_t(a) > \Lambda_t(b)$, when $c(a) = 0$, that is, when the graph has local negative curvature, under the same conditions, $\phi(a) < \phi(b)$. If, on the other hand, $c(a) = 1$, that is, the case of a positively curved graph, $\phi(a) = \phi(b)$ and $\Lambda_t(a) = \Lambda_t(b)$.

Since our conjecture is that the mass center will have the heaviest traffic congestion, we simulated both the traffic distribution and the distance squared distribution (or inertia) as the threshold is set to 0.1 (blue line) and 0.5 (black line) in Figure 21, where we set all edges to be of length one after threshold. (note: the node numbering of Figures 21–24 is by scanning Figures 19 and 20 columnwise, with node number 1 in Figures 21–24 corresponding to the point of coordinates (0, 0) in Figures 19 and 20).

The congestion point (node number 88) and the low inertia point (node number 88) are matching perfectly with threshold 0.5 (clustering coefficient 0.50095); they are not

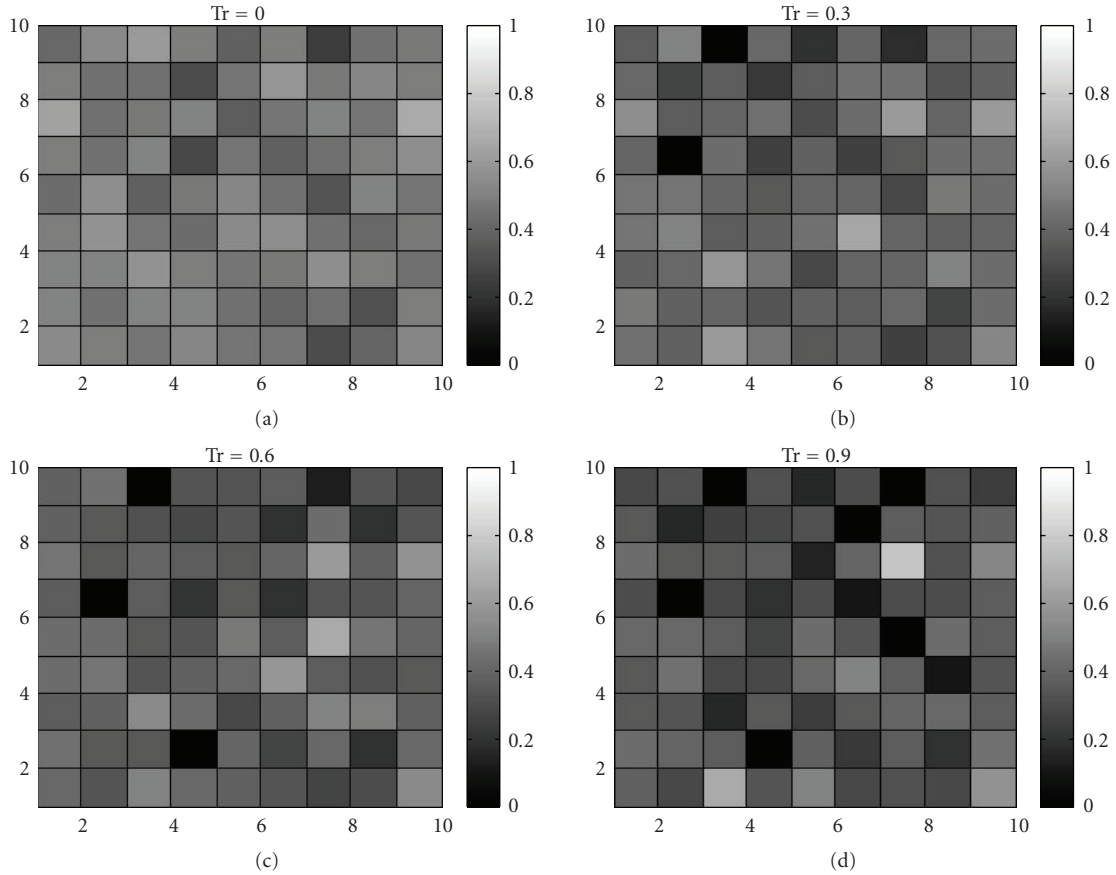


FIGURE 20: 3D illustration of spatial distribution of clustering coefficient across the graph: dataset B.

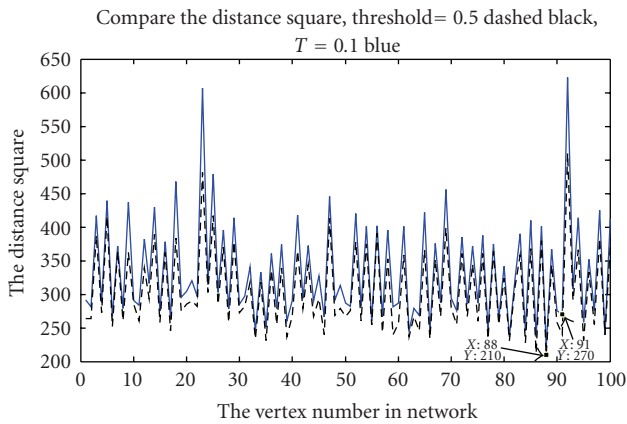


FIGURE 21: Inertia (distance squared) of the sensor networks with threshold 0.1 (solid blue line, positive curvature) and threshold 0.5 (dashed black line, negative curvature) versus vertex index. Observe that the inertia of the negatively curved network (dashed black) is lower than that of the positively curved network (solid blue).

quite matching once the network tends to be positively curved with threshold 0.1 (clustering coefficient 0.54088). However, more strikingly consistent with our conjecture is the fact that, as seen from Figures 22 and 23, traffic congestion is heavier around a limited number of nodes (of

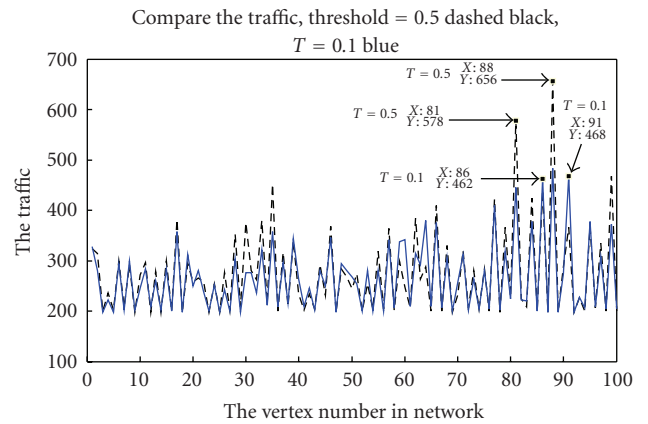


FIGURE 22: Traffic with threshold 0.1 (solid blue line, positive curvature) and threshold 0.5 (dashed black line, negative curvature) versus vertex index. Observe that the dashed black traffic curve (negative curvature) has higher spikes.

minimum inertia) in negatively curved network (threshold of 0.5) than in a positively curved one (threshold of 0.1).

Another way to see the results is through Figure 23, which shows that the mean and the standard deviation of the graph inertia increase with the threshold. In case of a positively

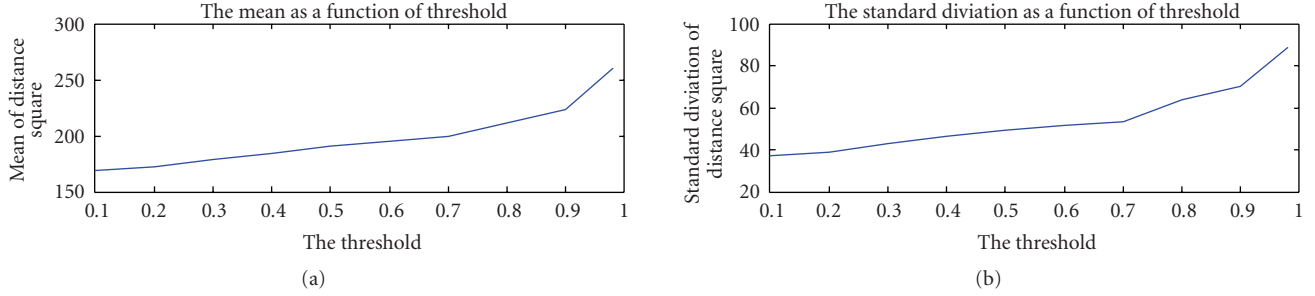


FIGURE 23: The mean and the standard deviation of graph inertia as a function of the threshold. Observe that both of them increase as the curvature becomes more and more negative.

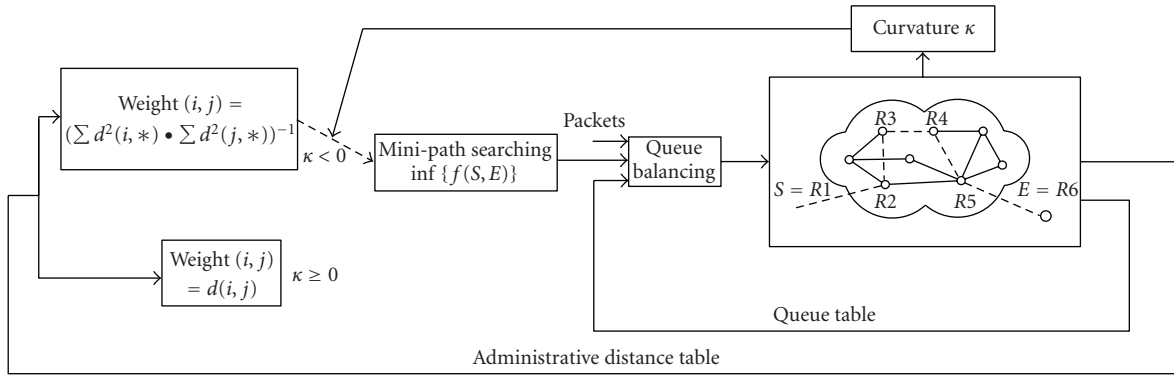


FIGURE 24: System-level diagram of curvature-based load balancing.

curved graph, the inertia is nearly constant, there are no obviously identifiable minima of the inertia, and no vertices stand out as heavily congested relative to the others. The situation is quite different as the threshold increases; the standard deviation of the inertia of the graph increases, some points stand out as minima of the inertia and are hence candidates for congestion.

The connection with the clustering coefficient can be seen from Figure 19 (and to a lesser extent from Figures 18 and 20 dealing with different datasets). Figure 19 indeed shows an area of low clustering (high congestion) around position number 88. Furthermore, by mere visual inspection of those figures, it is clear that the variance of the clustering varies consistently with the variance of the inertia under varying threshold.

To summarize, the higher the threshold, the smaller the clustering, the more the graph is negatively curved, the more is the tendency to have nodes standing out as heavily congested relative to the others.

7. DISCUSSION AND CONCLUSIONS

This paper has provided a detailed analysis of the curvature of a sensor network using the semilocal, but easily computable, concept of clustering coefficient. The latter provides a snapshot at the exact Riemannian curvature of the network. As far as the benchmark sensor network examples are concerned, numerical investigations have shown that, in case of high threshold, that is, when only the strong links are taken

into consideration, the curvature is negative. On the other hand, taking all links into consideration, including those of very small PRR, yields a network of positive curvature.

What is not completely obvious is the fact that such a *local* concept as clustering yields such a *global* insight as congestion. The explanation is to be found in the Riemannian geometry approach that this paper strives to justify. Probably the most important paradigm of Riemannian geometry is that the curvature, which can be defined very locally by computing various partial derivatives, yields global properties. Examples include the “sphere theorem,” saying that a Riemannian manifold with its sectional curvature uniformly bounded from below by $\kappa_{\min} > 0$ has its diameter bounded by $\pi/\sqrt{\kappa_{\min}}$. Since the clustering emulates the local sectional curvature, it provides a safe gateway to global properties.

From a practical networking perspective, the background motivation of this study has been congestion. The latter can be rephrased, a bit simplistically, as the fact that greedy routing on a negatively curved network creates very heavy congestion around a limited number of nodes. Since congestion can be traced to negative curvature, load balancing must somehow get around it. This leads to a curvature-based load balancing algorithm in which the link weights are deliberately distorted to create a virtual network of positive curvature. Dijkstra’s algorithm with random pick on the virtual network leads to paths, which, mapped back to the real network, provide better load balancing. This concept is illustrated in Figure 24 (see [12] for details).

Another significant feature that emerges is that positive curvature incurs a higher cost due to weak links and negative curvature incurs higher costs due to longer paths. Hence, it is fair to conjecture that there is an optimal threshold value between both extremes. This bears further study. We also speculate that there may be some other significant connections between the curvature and the performance of certain wireless sensor network algorithms. For instance, the convergence of distributed localization algorithms (such as iterative multilateration techniques [32]) and gossip-based algorithms for distributed aggregate computation [33] are likely to be impacted in a nontrivial manner by the graph curvature. This is because these iterative algorithms require local neighborhood message passing and computations (for which intuitively positive curvature may be helpful), but at the same time, they require rapid global dissemination (for which negative curvature may be beneficial). Last but not least, greedy geographic routing on hyperbolic plane embedding of the graph [34] should have better properties (e.g., smaller stretch and congestion) when applied to a graph that is negatively curved in the first place.

REFERENCES

- [1] M. D. Penrose, *Random Geometric Graphs*, Oxford University Press, Oxford, UK, 2003.
- [2] P. Gupta and P. R. Kumar, "Critical power for asymptotic connectivity," in *Proceedings of the 37th IEEE Conference on Decision and Control (CDC '98)*, vol. 1, pp. 1106–1110, Tampa, Fla, USA, December 1998.
- [3] J. Dall and M. Christensen, "Random geometric graphs," *Physical Review E*, vol. 66, no. 1, Article ID 016121, 9 pages, 2002.
- [4] C. Avin and G. Ercal, "On the cover time of random geometric graphs," in *Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP '05)*, vol. 3580 of *Lecture Notes in Computer Science*, pp. 677–689, Lisbon, Portugal, July 2005.
- [5] A. Goel, S. Rai, and B. Krishnamachari, "Monotone properties of random geometric graphs have sharp thresholds," *Annals of Applied Probability*, vol. 15, no. 4, pp. 2535–2552, 2005.
- [6] D. Ganesan, B. Krishnamachari, A. Woo, D. Culler, D. Estrin, and S. Wicker, "Complex behavior at scale: an experimental study of low-power wireless sensor networks," Tech. Rep. CSD-TR 02-0013, UCLA, Los Angeles, Calif, USA, February 2002.
- [7] J. Zhao and R. Govindan, "Understanding packet delivery performance in dense wireless sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 1–13, Los Angeles, Calif, USA, November 2003.
- [8] A. Woo, T. Tong, and D. Culler, "Taming the underlying challenges of reliable multihop routing in sensor networks," in *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys '03)*, pp. 14–27, Los Angeles, Calif, USA, November 2003.
- [9] A. Cerpa, J. L. Wong, L. Kuang, M. Potkonjak, and D. Estrin, "Statistical model of lossy links in wireless sensor networks," in *Proceedings of the 4th International Symposium on Information Processing in Sensor Networks (IPSN '05)*, pp. 81–88, Los Angeles, Calif, USA, April 2005.
- [10] M. Zuniga and B. Krishnamachari, "Analyzing the transitional region in low power wireless links," in *Proceedings of the 1st Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON '04)*, pp. 517–526, Santa Clara, Calif, USA, October 2004.
- [11] F. Ariaei and E. Jonckheere, "Cooperative "curvature-driven" control of mobile autonomous sensor agent network," in *Proceedings of the 46th IEEE Conference on Decision and Control (CDC '07)*, pp. 2540–2545, New Orleans, La, USA, December 2007.
- [12] M. Lou, *Traffic pattern analysis in negatively curved network*, Ph.D. dissertation, Department of Electrical Engineering, University of Southern California, Los Angeles, Calif, USA, 2008, <http://eudoxus.usc.edu/IW/Mingji-PhD-Thesis.pdf>.
- [13] E. A. Jonckheere, M. Lou, J. Hespanha, and P. Barooah, "Effective resistance of Gromov-hyperbolic graphs: application to asymptotic sensor network problems," in *Proceedings of the 46th IEEE Conference on Decision and Control (CDC '07)*, pp. 1453–1458, New Orleans, La, USA, December 2007, paper WePI20.12.
- [14] P. Collet and J.-P. Eckmann, "Dynamics of triangulations," *Journal of Statistical Physics*, vol. 121, no. 5-6, pp. 1073–1081, 2005.
- [15] J.-P. Eckmann and E. Moses, "Curvature of co-links uncovers hidden thematic layers in the World Wide Web," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 9, pp. 5825–5829, 2002.
- [16] M. P. do Carmo, *Riemannian Geometry*, Birkhäuser, Boston, Mass, USA, 1992.
- [17] M. Gromov, "Hyperbolic groups," in *Essays in Group Theory*, S. M. Gersten, Ed., vol. 8 of *Mathematical Sciences Research Institute Publication*, pp. 75–263, Springer, New York, NY, USA, 1987.
- [18] E. Jonckheere, "Worm propagation and defense over hyperbolic graphs," in *Proceedings of the 43rd IEEE Conference on Decision and Control (CDC '04)*, vol. 1, pp. 87–92, Atlantis, Bahamas, December 2004.
- [19] E. Jonckheere and P. Lohsoonthorn, "A hyperbolic geometry approach to multi-path routing," in *Proceedings of the 10th Mediterranean Conference on Control and Automation (MED '02)*, Lisbon, Portugal, July 2002.
- [20] E. Jonckheere and P. Lohsoonthorn, "Geometry of network security," in *Proceedings of the American Control Conference (ACC '04)*, vol. 2, pp. 976–981, Boston, Mass, USA, June-July 2004.
- [21] D. Burago, Y. Burago, and S. Ivanov, *A Course in Metric Geometry*, vol. 33 of *Graduate Study in Mathematics*, American Mathematical Society, Providence, RI, USA, 2001.
- [22] E. Jonckheere, P. Lohsoonthorn, and F. Bonahon, "Scaled Gromov hyperbolic graphs," *Journal of Graph Theory*, vol. 57, no. 2, pp. 157–180, 2008.
- [23] E. Jonckheere and P. Lohsoonthorn, *Coarse Geometry of Complex Networks*, book project draft, 2006, <http://eudoxus.usc.edu/IW/AIA.html>.
- [24] E. A. Jonckheere, P. Lohsoonthorn, and F. Ariaei, "Upper bound on scaled Gromov-hyperbolic δ ," *Applied Mathematics and Computation*, vol. 192, no. 1, pp. 191–204, 2007.
- [25] M. Lou, F. Ariaei, E. Jonckheere, and B. Krishnamachari, "Curvature of indoor sensor network: Alexandrov angles," submitted to *Journal of the ACM*.
- [26] E. Jonckheere, "Scaled Gromov-hyperbolic graphs: 4-point condition," submitted, <http://eudoxus.usc.edu/IW/AIA.html>.

- [27] M. K. Reiter, A. Samar, and C. Wang, "Distributed construction of a fault-tolerant network from a tree," Tech. Rep. CMU-CS-05-114, Carnegie Mellon University, Pittsburgh, Pa, USA, 2005.
- [28] S. A. Çamtepe, B. Yener, and M. Yung, "Expander graph based key distribution mechanisms in wireless sensor networks," in *Proceedings of IEEE International Conference on Communications (ICC '06)*, vol. 5, pp. 2262–2267, Istanbul, Turkey, July 2006.
- [29] L. M. Blumenthal, *Theory and Applications of Distance Geometry*, Oxford, Clarendon Press, London, UK, 1953.
- [30] L. Zhao, Y.-C. Lai, K. Park, and N. Ye, "Onset of traffic congestion in complex networks," *Physical Review E*, vol. 71, no. 2, Article ID 026125, 8 pages, 2005.
- [31] M. P. Joy, A. Brock, D. E. Ingber, and S. Huang, "High-betweenness proteins in the yeast protein interaction network," *Journal of Biomedicine and Biotechnology*, vol. 2005, no. 2, pp. 96–103, 2005.
- [32] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in Ad-Hoc networks of sensors," in *Proceedings of the 7th Annual International Conference on Mobile Computing and Networking (MOBICOM '01)*, pp. 166–179, Rome, Italy, July 2001.
- [33] J.-Y. Chen, G. Pandurangan, and D. Xu, "Robust computation of aggregates in wireless sensor networks: distributed randomized algorithms and analysis," in *Proceedings of the 4th IEEE International Conference on Information Processing in Sensor Networks (IPSN '05)*, pp. 348–355, Los Angeles, Calif, USA, April 2005.
- [34] R. Kleinberg, "Geographic routing using hyperbolic space," in *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM '07)*, pp. 1902–1909, Anchorage, Alaska, USA, May 2007.