Spherical Mapping Based Load Aware Routing for Wireless Sensor Networks

FATIMA MUHAMMAD ANWAR, Ajou University
MUHAMMAD TAQI RAZA, LG Electronics
SEUNG-WHA YOO, Ajou University
KI-HYUNG KIM, Ajou University

Energy management and routing issues are correlated in most of the applications for Wireless Sensor Networks (WSNs). Routing in WSNs mainly focuses on Geographical/Shortest Path Routing (SPR) which takes linear shortest distance as the routing metric. These shortest traveled distances mostly traverse through the centre of the network. This leads to an uneven distribution of the traffic load causing higher energy consumption in the core of the network. As a solution, in this paper, the energy usage fairness is introduced by proposing a type of routing which balances the load in the entire network. We propose Spherical Mapping based Routing (SMR) which maps a grid onto a sphere using spherical coordinates and takes circular shortest distance on the surface of the sphere as the routing metric. Our protocol not only balances the load but also makes sure that the average load and stretch factor for the network do not go out of bounds.
In our analytical and simulation evaluation, we have compared our routing protocol with the state of the art work in the load balancing routing. We have deduced an improvement of 26% in the stretch factor, and 33% in the average load per node – that results in fair energy distribution per node.

Categories and Subject Descriptors: **C.2.2 [Computer-Communication Networks]**: Network Protocols

General Terms: Performance, Protocol, Design, Measurement, Experiment

Additional Key Words and Phrases: Wireless sensor networks, routing protocols, applications, energy efficiency

1. INTRODUCTION

Recently, Wireless Sensor Networks (WSNs) have been emerged with its wide spread applications in the buildings, under water [Jun-Hong Cui et al. 2006], outer space and smart grids. In these applications, if the transmission range of an individual sensor node is less than the total height of the network (earth surface as the reference for height), then the scope of that network is 3D. Traditional approaches like Shortest Path Routing (SPR) are not inherently developed for 3D network deployments. These techniques mostly route the packets through middle nodes because most of the shortest paths traverse through the centre of the network. This creates congested regions called 'hot spots' in the core of the network and the phenomenon is called the 'crowed centre effect'. The nodes lying in the hot spot region are over burdened and may quickly run out of battery power. On the other hand, the nodes at the periphery remain rich in power as they are underutilized. This unfair distribution of energy makes the network heterogeneous in terms of power.

Moreover, the congested nodes at the centre could cause many other problems, such as, 1) unnecessary delays in the communication, 2) high packet loss at the central nodes, 3) energy depletion resulting in shorter network life time, and 4) security threats compromising the network core. Balancing load in the entire network solves almost all of these problems.

Much effort has been put to the solution of these issues through 'load balancing' approaches. Most of these well known techniques also take 3D network deployments

into account. For example, Circular path routing techniques adopt circular path over the sphere by calculating circular distances among the nodes. This behavior diverges the traffic from the congested region to the periphery of the network. However, it also introduces an overhead of increased traveled distance in terms of number of hops. Therefore, in later work, more efforts were put to minimize the increased traveled distance between a source and destination node. For example, Circular Sailing Routing (CSR) [Fan Li et al. 2008] and Curve Ball Routing [L. Popa et al. 2007] are two well known approaches for load balancing that use circular path routing. Both [Fan Li et al. 2008] and [L. Popa et al. 2007], first map 3D Cartesian domain to the sphere by stereographic projection, then they use Cartesian coordinates' projection and its geometry on the sphere to route the traffic. The mapping in [Fan Li et al. 2008] and [L. Popa et al. 2007] concentrate the projected node points on the upper or lower half of the sphere. Because of these node concentrations in fixed regions, the node points have lower connectivity in terms of number of neighboring nodes. That's why, in some cases, there is only one possible path between a source and destination node. This possible path could be a much longer route. Moreover, the scarcity of alternate shorter paths forces the network to rely on a single path. This single path could be a longer path option – hence the load on the nodes gradually increases. Therefore, the average load in the network is not only increased but the traveled distance between the source and the destination is also observed to be out of bounds for some cases.

In this paper, we focus on designing a routing algorithm for load balancing using circular path routing. We map the 3D grid to the sphere and use spherical coordinates instead of cartesian coordinates for mapping and routing. Spherical coordinates explain the geometry of the sphere in a better way, as well as help with the calculations in spherical trigonometry. Any SPR [Y. Yu et al. 2001], [M. Zorzi et al. 2003], [Rao A. Ratnasamy S. et al 2003], [J.Bruck et al. 2005] or Localized SPR [B. Karp et al. 2000] approach may be used for routing on the sphere with spherical distance as the routing metric instead of the euclidean distance. The nodes only need to compute their spherical coordinates and this overhead is negligible. Moreover, there is no extra communication overhead since traditional routing techniques are applied on the sphere.

Spherical coordinates based mapping spreads the mapped nodes all over the spherical surface. Our spherical distance calculation between two mapped nodes on the sphere produces a network interconnection in such a way that most of the mapped nodes find more than two neighbors to forward their packets. Thus, each node has more than two paths for each destination. This helps in finding the shorter path among all of the available choices. Therefore, the stretch factor of our protocol is improved, where,

$$Stretch\ Factor\ = \frac{Number\ of\ hops\ taken\ by\ Circular\ Path\ Routing}{Number\ of\ hops\ taken\ by\ Shortest\ Path\ Routing}$$

Furthermore, we assume an all-to-all communication scenario where each node sends a packet to all other nodes in the network. All of these packets traverse over minimum possible number of hops, as stated earlier. This reduces the total packet load on an individual node which implies that the average load of the network is decreased.

We have provided analytical and simulation based results for our proposed protocol. By comparing our proposed technique with other well known approaches, we can observe an improvement in the existing work in terms of throughput and network load etc.

In short, the main contributions of our work are as follows.

- We introduce a technique for mapping the coordinates in a 3D grid to a sphere. We use spherical coordinates with proper equations for mapping. This mapping approach maps every point to its unique position and preserves the neighbors for each node.
- We put forward a new routing metric, called spherical distance. This distance is not linear, rather it is circular. It is calculated through a formula using the spherical coordinate positions of source and destination nodes. We will also discuss different cases for spherical distance calculation.
- We present a routing algorithm which uses the spherical distance as the routing metric. Routing is done on the surface of a sphere circularly, using the spherical distance. This approach finds the optimal shortest circular path on the surface of the sphere. When the circular path on the surface of the sphere is mapped to a linear path in the 3D grid, the resultant path is not the shortest path in the grid. We make sure that our resultant path in the grid is as close to the shortest path as possible.

The rest of the paper is organized as follows. In Section 2, we review the related work. In Section 3 and Section 4, we describe the proposed spherical coordinates based mapping and routing scheme; whereas analytical evaluation of our approach is discussed in Section 5. We compare our protocol with Circular Sailing Routing and Shortest Path Routing in Section 6. Section 6 also provides discussion on simulation results. Finally in Section 7, we conclude the paper.

2. RELATED WORK

The Underwater Acoustic Sensor Network [I. F. Akyildiz, et al. 2005] is a perfect example of a 3D network. The sensors in such a network should be able to relay information to the surface station via multi-hop paths. Thus, these sensors should coordinate their depths in such a way that the network topology is always connected, i.e., at least one path from every sensor to the surface station should always exist. Therefore, congestion should be avoided from these paths. This paper just formulates a problem of congestion but didn't go in depth of the solution to this problem.

To mitigate congestion in a network, there are many load balancing routing techniques in the literature. Most of these routing techniques take load balancing into account after the node has already been congested. For example, the approach in [Sung-Ju Lee et al. 2001] uses the amount of load on the node as the routing metric. Once some paths in the network are found to be congested, measures are taken to avoid routing the traffic through those paths. Also, [Lei Yung et al. 2008] introduces back pressure algorithm where nodes only need the queue length information of neighboring nodes to make routing decisions, and packets are adaptively routed in the network according to the congestion information. Moreover, Hotspot Mitigation Protocol (HMP) [S.-B. Lee et al. 2003] monitors local buffer occupancy, packet loss, MAC contention and delay conditions, and take local actions in response to the emergence of hotspots, such as, suppressing new route requests and rate controlling TCP flows. Since these algorithms are resilient to traffic and topology changes, the network is not scalable.

The concept of multiple path routing is used in [M.R. Pearlman et al. 2000], [Y. Ganjali et al. 2004], [S. Kwon et al. 2007] and [W. Wang et al. 2004] i.e. the traffic on the overloaded paths is diverged in multiple directions using multiple paths. Hence, for the same source or the same destination, the routing paths could be different. But

load balancing is achieved only if large number of paths is used; otherwise the performance is equivalent to the performance of single path routing. The concept of Oblivious routing for geometric networks [Costas Busch et al. 2005] is similar to multipath routing. A packet do not follow optimal paths, rather they chose paths with lower stretch factor. They argue that the motivation for minimizing congestion and stretch simultaneously is that some packet scheduling algorithms, which deliver the packets along the given paths in time, are very close to the optimal. The problem is that the oblivious routing [Costas Busch et al. 2005] chooses the next hop nodes randomly in order to avoid congestion.

Curve Ball Routing [L. Popa et al. 2007] addressed the problem of balancing the traffic in multihop wireless networks to increase energy usage fairness and reduce congestion. The main idea is to mitigate congestion by routing on curved paths rather than the shortest ones. Similarly, the double rulings scheme in [R. Sarkar et al. 2006] proposes to store data replica at a curve instead of one or multiple isolated sensors. The client node travels along a curve which guarantees to intersect with the host node. These protocols provide reduced communication costs and more balanced traffic load on the sensors. But there is no exact method or procedure mentioned in [L. Popa et al. 2007] and [R. Sarkar et al. 2006] to map the nodes on the curved surface.

J. Gao and L. Zhang [Jie Gao et al. 2004] describes load balancing together with shortest path routing by considering a single traffic pattern, where nodes are placed in a narrow strip of constant width. It achieves good stretch factor and load balancing ratio and deals with dynamic changes efficiently. However, it needs to deal with other node distributions as well. For example it argues that load balanced routing is difficult for nodes placed in the regular grid.

In Circular Sailing Routing (CSR) [Fan Li et al. 2008], wireless nodes in a 2D network are mapped to a sphere using reversed stereographic projection and the routing decision is made based on the circular distance on the sphere instead of the Euclidean distance in 2D plane. By doing so, the hot spots could be eliminated but it results in increased average load in the whole network. Also [Fan Li et al. 2008] argues that besides stereographic projections, there are area-preserving map projections, such as the Lambert azimuthal equal-area projection, which maintains the size at the expense of shape. But they all face a common problem that the routing distance travelled is much greater than the distance traveled by shortest path routing techniques.

Routing in outer space [Alessandro Mei et al. 2000] propose that every node of the network is responsible for relaying the same number of messages, on expectation. Hence, the message flow is homogeneously distributed over all the network area, the network does not have the associated security-related issues and does not encourage selfish positioning. Furthermore, with routing in outer space, the load among network nodes is equally balanced, with the intuition that this routing protocol also brings significant improvement in energy-efficiency issues. This technique offers the same problem as [Fan Li et al. 2008] where the routing path length in terms of hops is out of limits. In case of [Alessandro Mei et al. 2000], routing path length goes severely out of bounds.

3. SPHERICAL COORDINATES BASED MAPPING

We will describe the spherical coordinates based mapping in a step by step manner. First, we will highlight some preliminaries of our proposed mapping algorithm followed by the motivation of our work. After that, the procedure of the proposed mapping algorithm will be described in detail.

3.1 Preliminaries

In this paper, the sensor nodes in a network are referred as '*points'* in a 3D grid. Each point in the grid has its own coordinate system *(x, y, z).* These points in the grid are mapped to the points on the surface of the sphere named as *'projected points'* in this paper.

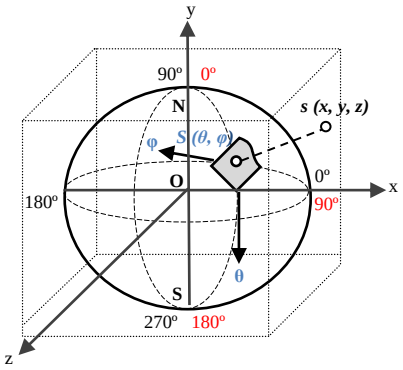


Figure 1. Spherical coordinates based mapping from a cube to a sphere: one to one mapping from a node s in cube to a node S on sphere

The type of mapping, we use, is the *spherical coordinates based mapping*.

The spherical coordinates based mapping uses spherical coordinates on the surface of the sphere. As we know that three variables are used to identify a point in the spherical domain. These three variables are *Rho (ρ), Theta (θ)* and *Phi (φ).* Since we need to map points on the *surface of a single sphere*, we set ρ as constant and only two variables, θ and φ are required for mapping.

We will now describe Figure 1 in detail. The 3D grid is shown in the form of a cube in this figure because the boundary points of the grid are arranged equidistantly from all the three x, y and z axes, thus forming a cube. Therefore, the terms 'cube' and 'grid' are used interchangeably further in this paper. For simplicity of explanation, a sphere is placed inside the cube. The center of both the sphere and the cube is same i.e. at *(x, y, z) = (0, 0, 0).* The North Pole of the sphere is at the coordinate, *(x, y, z) = (0, 0, ρ)* or *(θ, φ) = (0, arbitrary),* while its antipodal point is at the South Pole with coordinate *(x, y, z) = (0, 0, -ρ)* or *(θ, φ) = (π, arbitrary).* The vertical curve shown in Figure1, joining the North and the South Pole, denotes the values and direction of θ and is called *'constant-phi curve',* while the horizontal circle specifies the values and direction of φ and is called *'constant-theta circle'.* Direction of θ is from North to South Pole while direction of φ is anticlockwise. In Figure1, the values of θ are shown in red; whereas the values of φ are shown in black. The range of θ is from 0º to 180 º whereas; it is 0º to 360º for φ. A point *s(x, y, z)* in the grid is mapped to a point *S(θ, φ)* on the surface of the sphere.

3.2 Motivation

The idea of using spherical coordinates based mapping is inspired form the geographical coordinate system of the earth. Earth is almost a sphere with an approximate constant radius. Different locations on the earth are determined by their latitude and the longitude on its surface. Latitude is the angle from a point on the earth's surface (i.e. surface of the sphere) to the equatorial plane, measured from the center of the earth (i.e. sphere). Whereas, longitude is the angle, east or west of a

reference meridian between the two geographical poles to another meridian that passes through an arbitrary point [Geographic coordinate system, Wikipedia]. Latitude is equivalent to the angle θ *(theta)* of spherical coordinates system and Longitude is equivalent to the angle φ *(phi)* of spherical coordinates system. The angle θ and latitude have a phase difference of 90º; whereas φ and longitude have a phase difference of 180 º. The reason why we are comparing both the geographical coordinate system and the spherical coordinate system is because we assume that the sphere in our spherical mapping is the earth, the projected points on the sphere are the distant locations on the earth and the circular distance between two projected points on the sphere surface, is the distance between two locations on the earth. We have used spherical coordinates instead of geographical coordinates because of two reasons. First, the spherical coordinates provide ease of use when dealing with a sphere in our mapping. Second, the spherical geometry and trigonometry are better tools for theoretical considerations. Third, there is not much difference between the two coordinate systems, except for a small phase difference between there angles which could be ignored.

3.3 Mapping Procedure

Let s with coordinate *(x, y, z)* be a point in the cube and it is mapped to a small area element S on the surface of the sphere. We assume that this element S is infinitely small and converges to a point with coordinate *(θ, φ)*. This projected point is shown at the intersection of constant-phi curve and constant-theta circle in Figure 1.

By using the following method, we can find the position of the projected point on the sphere.

3.3.1 Method 1

Given a point s with coordinate *(x, y, z)* in the 3D grid, the coordinate of the projected point S using spherical coordinates based mapping is (θ, φ), where,

$$\theta = \cos^{-1}(\frac{z}{\rho});\ \varphi = \sin^{-1}(\frac{y}{\rho \sin\theta})\ \text{OR}\ \cos^{-1}(\frac{x}{\rho \sin\theta});$$

However, there is one problem in this mapping using Method 1. The centre point O *(0, 0, 0)* in the grid has no definite projected point on the sphere. Thus, we map O to the North Pole of the sphere, by force. The only neighbors of O on the sphere are the other projected points on the North Pole.

The '*spherical routing path'* on the surface of the sphere consists of projected points (to be described in Section 4). Once this spherical routing path is established on the surface of the sphere, we need to map the projected points in the spherical routing path to points in the 3D grid. These points in the 3D grid will form the '*linear routing path'*. In order to get this linear routing path in the 3D grid, we have to convert the spherical coordinates *(θ, φ)* of the projected points in the spherical routing path to the cartesian coordinates *(x, y, z)* of the points in the linear routing path. By using the Reverse Spherical Coordinates based Mapping in Method 2; we can find the route in 3D grid.

3.3.2 Method 2

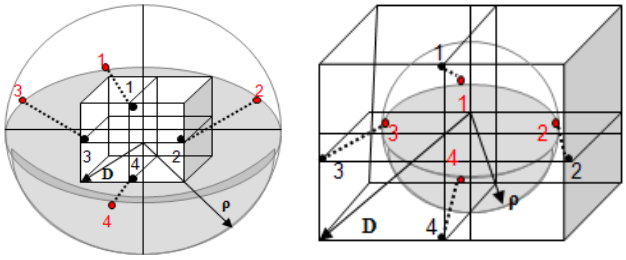
Given a projected point S with coordinate (θ, φ) on the sphere, the coordinate of the point s using reverse spherical coordinates based mapping is *(x, y, z),* where,

$$y = \rho \sin\theta \sin\varphi;\ \ x = \rho \sin\theta \cos\varphi;\ \ z = \rho \cos\theta$$

3.4 Spherical Mapping Properties

There are some key properties of our mapping approach that makes it the most suitable for Circular routing.

- The spherical coordinates based mapping is unique as no two points in a 3D grid map to a single point on the sphere.
- The neighbor's location is preserved through this mapping. For example, a point in the grid and the projected point on the sphere will have the same neighbors.
- There is one special property of spherical coordinates based mapping i.e. the rad ius of the sphere does not have any effect on the mapping of the points on the su rface of the sphere. If we make the radius smaller or bigger, the projected points will always map to the same position with respect to the whole projected networ k. The mapped points and the routing paths remain the same irrespective of wh ether we place the sphere inside or outside the cube. These two cases have been described in Figure 2 (a) and Figure 2 (b). The black points 1, 2, 3 and 4 in the c ube, map to the corresponding red points on the sphere. Note that the red projec tions are same whether the sphere is inside or outside the cube. So there is no op timal radius of the sphere with respect to the grid. But for the sake of simplicity of explanation, we have considered some bounds on the relationship between siz e of grid and the radius of the sphere (to be discussed in Section 5.2).



(a) $\rho > D$ (b) $\rho < D$

Figure 2. The mapping of points from 3D grid to projected points on the surface of various spheres of different sizes: (a) $\rho > D$ (b) $\rho < D$

There are other mapping techniques that could be used besides spherical coordinates based mapping but we argue that due to the properties mentioned above and other reasons to be explained later, the desired results of bounded average load and stretch factor could better be achieved by our proposed approach

4. SPHERICAL MAPPING BASED ROUTING (SMR)

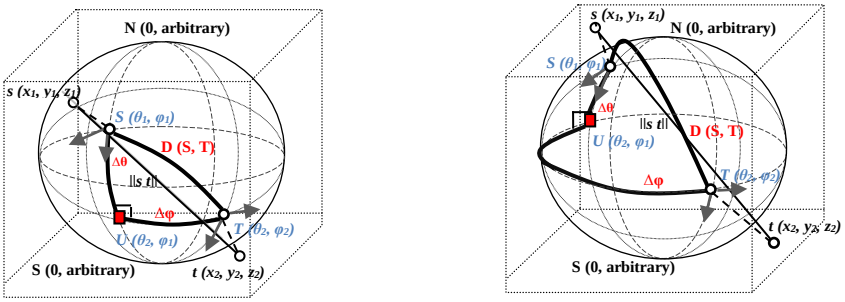
The shortest distance between two points on the earth's surface is a circular distance rather than a linear distance. This circular distance is required for different type of traditional applications like navigation and astronomy [Great-circle distance, Wikipedia]. We will use the same circular distance on the surface of the sphere as our routing metric. This distance is called the '*great circle distance'* or the '*orthodromic distance'*. Between any two points on a sphere which are not directly opposite to each other, there is a unique great circle. The two points separate the great circle into two arcs. The length of the shorter arc is the great circle distance between the points. A great circle endowed with such a distance is the Riemannian circle [Great-circle

distance, Wikipedia]. In our paper, we will use the term *'spherical distance'* instead of great circle distance.

4.1 Spherical Distance Calculation

Let us denote the Euclidean distance between points in the grid as $||s\ t||$ and the spherical distance between two projected points as *D(S, T)*. Now we will show how to compute the spherical distance between two projected points on the sphere using spherical trigonometry. See Figure 3 for illustration. Let S and T be two projected points on the surface of the sphere with *D(S, T)* as the spherical distance between them. Given the coordinates of S and T i.e. (θ_1, φ_1) *and* (θ_2, φ_2) respectively, we can find an intermediate projected point U with coordinates (θ_2, φ_1). Using the coordinates of S, T and U, $\Delta\theta$ and $\Delta\varphi$ are calculated, where, $\Delta\theta = |\theta_2 - \theta_1|$; and $\Delta\varphi = |\varphi_2 - \varphi_1|$. Using $\Delta\theta$ and $\Delta\varphi$, *D(S, T)* is calculated as follows,

$$D(S,T) = \tan^{-1} \frac{\sqrt{\sin\Delta\theta^2 + (\sin\Delta\varphi\cos\Delta\theta)^2}}{\cos\Delta\theta\cos\Delta\varphi}$$



Δφ < π (b) Δφ ≥ π

Figure 3. The spherical distance calculation between points S and T on the surface of the sphere.

The two cases in Figure 3 shows, how a spherical distance is calculated when the value of $\Delta\varphi$ is less than π and when its value is larger than equal to π. In either case, the calculation procedure is the same. But figuratively, the two cases look different. For example, in Figure 3 (a), for the case of $\Delta\varphi < \pi$, the arc representing the spherical distance traverses only a small part of the sphere. While for the case, $\Delta\varphi \geq \pi$, the arc of spherical distance almost traverses half of the sphere. Note that the arc for the spherical distance is the smallest possible arc that could be passed between two points on the surface of a sphere. Also, note that the value of $\Delta\theta$ does not have much effect on the spherical distance as compared to $\Delta\varphi$. For a same value of $\Delta\varphi$, whether the value of $\Delta\theta$ is close to 0 or π, it does not alter the length of the spherical distance significantly. But for the same value of $\Delta\theta$, the value of $\Delta\varphi$ significantly affects the spherical distance arc**.**

4.2 Routing Algorithm

The sensor nodes have the information about their cartesian coordinates *(x, y, z)* either through a GPS or the information could be fed to the nodes before the deployment. During the provisioning phase, the sensors may exchange their spherical coordinates (θ, φ), where the spherical coordinates are calculated through

their cartesian coordinates *(x, y, z)* using Method 1. A sensor node utilizes less processing memory and time to calculate its own projection coordinates (spherical coordinates). The nodes exchange their spherical coordinates by broadcasting or by allowing the neighboring nodes to exchange packets. After that, each node calculates the spherical distance with its neighboring nodes. Any shortest path routing approach e.g. dijkstra or greedy routing etc. is applied with spherical distance as the routing metric. Hence, a routing path is obtained on the surface of the sphere as shown in Figure 4 (a).

In Figure 4 (b), the network topology and nodes connectivity for a 3x3x3 grid is shown. This grid is mapped on a surface of a sphere and the new topology and nodes connectivity is shown in Figure 4 (a). The source node is node number 1 and the destination node is node number 24. The red line shows the mutihop routing path from node 1 to node 24, both in the grid and on the sphere.

The detailed algorithm for Spherical Mapping based Routing (SMR) is given as follows:

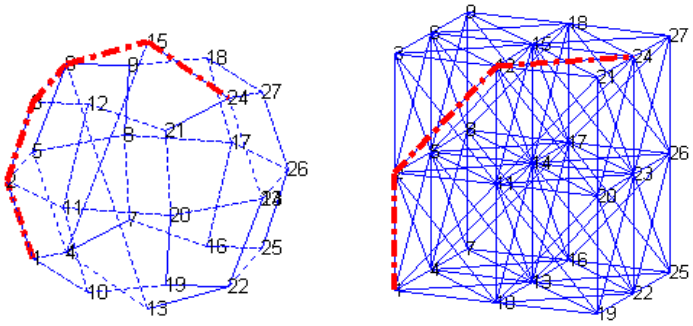
ALGORITHM 1. Spherical Mapping based Routing (SMR)

Mapping: Map each node *s (x, y, z)* in the 3D grid to a node *S (θ, φ)* on the sphere using Spherical Coordinates based Mapping (refer to Section 3.3.1, Method 1).

Routing metric: For every existing link st between nodes *s* and *t* with Euclidean distance $||s\ t||$ in the 3D grid, calculate the spherical distance, *D(S,T)* between their projected nodes *S* and *T*. We use *D(S,T)* as the routing metric (refer to Section 4.1, Figure 3 (a), (b)).

Routing: Apply any general shortest path routing or greedy routing with spherical distance as the routing metric. Chose the route with smallest total circular distance from *S* to *T* on the sphere (refer to Figure 4 (a)).

Routing Path: Let the route on the sphere be, *S, U_1 ,U_2 ,U_3,..., T*. Map the nodes back to the 3D grid using reverse spherical coordinates based mapping and the desired routing path will be *s, u_1, u_2, u_3,..., t* (refer to Section 3.3.2, Method 2 and Figure 4 (b)).



(a) (b)

Figure 4. Blue lines show the network topology and connectivity for a 3D sphere and a 3x3x3 grid. Red line showing the routing path from source node to the destination node: (a) Spherical Mapping based Routing (SMR) (b) Reverse Spherical Mapping based Routing

5. ANALYTICAL EVALUATION

In this section, we will discuss the performance of our routing technique, SMR. We will also discuss two different cases for our evaluation e.g., one special case and one general case.

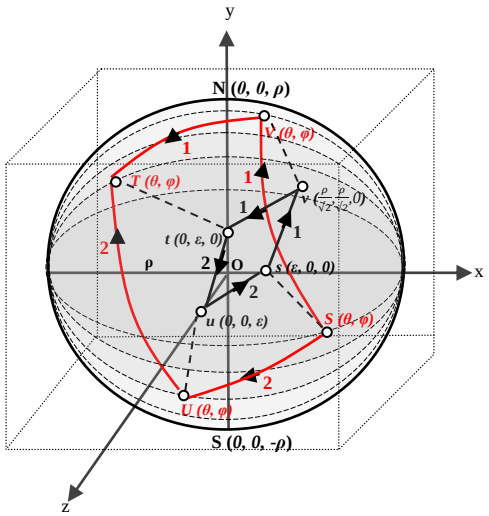


Figure 5 (a). Two Eucledian paths (1, 2) in the grid are shown in black and two Spherical paths (1, 2) on the surface of the sphere are shown in red.

5.1 Special Case

Refer to Figure 5 for the special case. Assume that the network has only four points: *s, t, u* and *v* with coordinates *(ε, 0, 0), (0, ε, 0), (0, 0, ε),* and $(\frac{\rho}{\sqrt{2}},\frac{\rho}{\sqrt{2}},0)$ respectively. Here, *ε* could be an arbitrary small number. Moreover, the network only has four links: *su, sv, ut,* and *vt*. From *s* to *t*, there are two paths, $s \rightarrow u \rightarrow t$ and $s \rightarrow v \rightarrow t$. Applying the CSR projection method, CSR gets the positions of the projected points: *S'(ρ, 0, 0, ε), T'(0, ρ, 0, ε), U'(0, 0, ρ, ε),* and $V'(\frac{\rho}{\sqrt{2}},\frac{\rho}{\sqrt{2}},0,\rho)$. Since $d(S', U')+d(U', T') = \pi\frac{\rho}{2}$, (where, *d(S, U)* is the circular distance for CSR) which is smaller than $d(S', V')+d(V', T') = \pi\rho$, CSR will chose point *u* to relay packets between *s* and *t*. The total distance of $\boldsymbol{P_{CSR}}(s, t) = ||s\ u|| + ||u\ t|| = \sqrt{\left(\frac{\rho}{\sqrt{2}}-\varepsilon\right)^2 + (\frac{\rho}{\sqrt{2}})^2}$. However, the length of the shortest path $\boldsymbol{P_{SPR}}(s, t) = ||s\ v|| + ||v\ t|| = 2\sqrt{2}\varepsilon$. Thus, the stretch factor, $\boldsymbol{SF_{CSR}} = \frac{\boldsymbol{PCSR}(s,t)}{\boldsymbol{PSPR}(s,t)} = \sqrt{\frac{\sqrt{\left(\frac{\rho}{\sqrt{2}}-\varepsilon\right)^2+(\frac{\rho}{\sqrt{2}})^2}}{2\varepsilon^2}}$. When *ε* is arbitrary small, *SF* can be arbitrary large. CSR may use an arbitrary longer path than the shortest path in the worst case. Fortunately, the worst case seldom occurs in a random network but it occurs very often in the grid network [J. Gao et al. 2004].

We proved that the stretch factor for CSR is very large especially for the worst case in grid network. Now we evaluate SMR analytically by proving that its stretch factor is not that high. Assume the same grid topology for SMR as shown in Figure5 for fair comparison. The 3D points and their respective spherical mapping based coordinates

are shown in Figure 5. The projection of the points *s, t, u* and *v* with coordinates *(ε, 0, 0), (0, ε, 0), (0, 0, ε),* and $(\frac{\rho}{\sqrt{2}},\frac{\rho}{\sqrt{2}},0)$ respectively are calculated and shown in Table.1.

Table 1. Cartesian coordinates to Spherical coordinates mapping

Coordinates in the 3D Grid	Coordinates on the surface of the sphere
$s\ (\varepsilon, 0, 0)$	$S\ (\theta, \varphi) = (\arcsin\ (\frac{\varepsilon}{\rho}, \frac{\pi}{2})$
$t\ (0, \varepsilon, 0)$	$T\ (\theta, \varphi) = (\arcsin\ (-\frac{\varepsilon}{\rho}, \frac{3\pi}{2})\ )$
$u\ (0, 0, \varepsilon)$	$U\ (\theta, \varphi) = (0, \arccos\ (\frac{\varepsilon}{\rho}))$
$v\ (\frac{\rho}{\sqrt{2}},\frac{\rho}{\sqrt{2}},0)$	$V\ (\theta, \varphi) = (\frac{\pi}{4}, \frac{\pi}{2})$

$$Path\ 1 \rightarrow D(S, V) + D(V, T) = \arccos\{\cos(\tfrac{\pi}{4} - \lambda)\cos(0)\} + \arccos\{\cos(-\lambda - \tfrac{\pi}{4})\cos(\pi)\} \approx \frac{\pi}{2}$$

$$Path\ 2 \rightarrow D(S, U) + D(U, T) = \arccos\{\cos(\sigma - \tfrac{\pi}{2})\cos(-\lambda)\} + \arccos\{\cos(-\lambda)^2\} \approx \pi + \sigma$$

where,

$\sigma \rightarrow$ very large number and $\lambda \rightarrow$ very small number.

Therefore, SMR takes the circular routing path $S \rightarrow V \rightarrow T$ which has the shorter spherical distance. Hence, SMR will choose node *v* to relay packets between *s* and *t*. Now, $\boldsymbol{P_{SMR}(s, t)} = ||s\ v|| + ||v\ t|| = 2\sqrt{2}\varepsilon = \boldsymbol{P_{SPR}(s, t)}$. Hence, $\boldsymbol{SF_{SMR} = \frac{PSMR(s,t)}{PSPR(s,t)} = 1.}$

We have proved for the worst case that, $\boldsymbol{SF_{SMR} < SF_{CSR}}$.

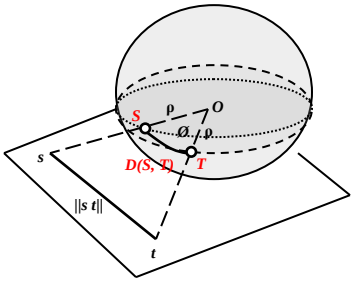


Figure 5 (b). Grid to spherical Projection diagram (Lemma 1)

5.2 General Cases for Grid Networks

Now we will briefly explain some lemmas for routing path length measurement.

5.2.1 Lemma 1

Consider two nodes s and t in 3D grid with projections S and T on a 3D sphere, we have,

$$D(S,T) \leq \frac{\pi\,(n-2)}{n}||s\ t||$$

Proof: We know that the shortest distance in Euclidean plane is always smaller than the circular distance in the spherical plane. In this lemma we will calculate the actual approximation of the two distances. The two point's s and t in the grid are $||s\ t||$ distance apart. Similarly S and T are *D(S, T)* spherical distance apart on the surface of the 3D sphere. According to Figure 5 (b), *D(S, T)* = $\rho\emptyset$, where ρ is the radius of the sphere and its range is $^{L(n-2)}/_{n} \leq \rho \leq {}^{L}/_{2}$, where L is the length and n is the order of the grid. The angle $\emptyset$ is subtended by OS and OT and it ranges from *0* to π. The Euclidean distance $||s\ t|| = \sqrt{Os^2 + Ot^2 - 2\,(Os)(Ot)\cos\emptyset}$, where, *Os* and *Ot* are sides of the triangle *Ost* and their range is ${}^{L}/_{n} \leq Os, Ot \leq \sqrt{3}\,{}^{L}/_{n}$. This range is unique for a grid network only. The angle $\emptyset$ is the angle opposite to $||s\ t||$. Here, note that the angle $\emptyset$ is the same for OST and Ost. Now we will compare the two distances, $\frac{D(S,T)}{||s\ t||} = \frac{\rho\emptyset}{\sqrt{Os^2+Ot^2-2\,(Os)(Ot)\cos\emptyset}}$. We will analyze for the worst case, when any two nodes are exactly opposite on the surface of the sphere. We take $\emptyset = \pi$, $\rho = {}^{L}/_{n}$ and *Os, Ot* = ${}^{L}/_{2}$(Two nodes opposite to each other in the grid, are separated by the length of the grid).

$$\Rightarrow \quad \frac{D(S,T)}{||s\ t||} = \frac{{}^{L}/_{n}\,\pi}{\sqrt{(L/2)^2+(L/2)^2-2\,(L/2)(L/2)\cos\pi}}$$

$$\Rightarrow \quad \frac{D(S,T)}{||s\ t||} = \frac{\pi\,(n-2)}{n}$$

$$\Rightarrow \quad D(S,T) \leq \frac{\pi\,(n-2)}{n}\ ||s\ t||$$

5.2.2 Lemma 2

Distance travelled by SMR is within a constant stretch factor $\frac{\pi\,(n-2)}{n}$ of SPR. i.e.

$P_{SMR}(s,t) \leq \frac{\pi\,(n-2)}{n} P_{SPR}(s,t)$

Proof: Let the path traversed from S to T by SMR; on the surface of the sphere be, S, U_1, U_2, U_3,…, U_p, where, $U_p = T$ and the corresponding path in the grid will be, s, u_1, u_2, u_3,…, u_p, where $u_p = t$. The path traversed by SPR from s to t in the grid be, s, v_1, v_2, v_3,…, v_q, where, $v_q = t$ and the corresponding path on the sphere will be, S, V_1, V_2, V_3,…, V_q. where, $V_q = T$. Note that $p \neq q$. From lemma 1, $\sum_{i=1}^{p} \mathrm{D(U\,i-1, U\,i)} \leq \frac{\pi\,(n-2)}{n} \sum_{i=1}^{p} ||\,u\ i-1\ u\ i\,||$ and $\sum_{i=1}^{q} \mathrm{D(V\,i-1, V\,i)} \leq \frac{\pi\,(n-2)}{n} \sum_{i=1}^{q} ||\,v\ i-1\ v\ i\,||$ As, $P_{SMR}(S,\ T) = \sum_{i=1}^{p} D(U\ i-1, U\ i)$, $P_{SMR}(s,\ t) = \sum_{i=1}^{p} ||\,u\ i-1\ u\ i\,||$, $P_{SPR}(S,\ T) = \sum_{i=1}^{q} D(V\ i-1, V\ i)$, $P_{SPR}(s,\ t) = \sum_{i=1}^{q} ||\,v\ i-1\ v\ i\,||$, $P_{SPR}(s,\ t) \leq P_{SMR}(s,t)$ and $P_{SMR}(S,T) \leq P_{SPR}(S,T)$. It implies that, $P_{SMR}(s,t) \leq \frac{\pi\,(n-2)}{n} P_{SPR}(s,t)$ i.e. SMR is within a constant stretch factor $\frac{\pi\,(n-2)}{n}$ of SPR.

5.2.3 Lemma 3

Distance travelled by SMR is less than equal to CSR by a constant factor of $\frac{n\,(1+\varepsilon)}{2\,(n-2)}$ .i.e.

$P_{CSR}(s,t) \leq \frac{n\,(1+\varepsilon)}{2\,(n-2)} P_{SMR}(s,t)$.

Proof: From Lemma 2, $P_{SMR}(s,t) \leq \frac{\pi(n-2)}{n} P_{SPR}(s,t)$*…(1).* Also, $P_{CSR}(s,t) \leq \frac{\pi(1+\varepsilon)}{2} P_{SPR}(s,t)$*…(2)* [Fan Li et al. 2008]. From *eq(1),* $P_{SPR}(s,t) \geq \frac{n}{\pi(n-2)} P_{SMR}(s,t)$. Putting this in *eq(2)*, we get, $P_{CSR}(s,t) \leq \frac{n(1+\varepsilon)}{2(n-2)} P_{SMR}(s,t)$.

As we know that $n > 2$ and ε is a very small positive number, it implies from lemma 3 that $\frac{n(1+\varepsilon)}{2(n-2)} \geq 1$. So we can conclude that the total path length in SMR approach is less than CSR. We can imply further from the above lemma that as the order of the grid increases, the path length in SMR approaches the path length is CSR.

5.2.4 Lemma 4

The ratio of the path lengths of CSR and SMR approaches $\frac{1+\varepsilon}{2}$ as the order of the grid, n approaches to infinity.

Proof: From lemma 3, $P_{CSR}(s,t) \leq \frac{n(1+\varepsilon)}{2(n-2)} P_{SMR}(s,t)$. As n approaches to infinity, i.e. the order of the grid is very high, so $\frac{PCSR(s,t)}{PSMR(s,t)} \leq \frac{1+\varepsilon}{2}$. The ratio $\frac{1+\varepsilon}{2}$ is less than 1.

From the above lemma, it is proved that the path length in CSR is less than SMR when the order of the grid is very high, theoretically approaching infinity. Fortunately, this case does not occur in the practical applications and lemma 3 applies to most of the real world scenarios.

6. SIMULATION

We evaluate our Spherical Mapping based Routing protocol using two metrics: load distribution and stretch factor. Load Distribution gives the average amount of traffic load on each node in the network. We assume that the traffic load consists of routing traffic only. Stretch factor is a constant used to compare the average path length (interms of hops) of circular routing to the traditional shortest path routing protocols in WSNs.

We take a look at the effect of mapping and routing in the spherical domain on our performance metrics. SMR claims to distribute the load evenly in the network while making sure that its circular routing path length does not exceed the shortest path routing length by a small constant factor.

The circular sailing routing [Fan Li et al. 2008] technique and the traditional shortest path routing techniques are used as refernces for the comaprison with SMR. We use Dijkstra as the shortest path routing algorithm for our comparison results.

6.1 Simulation Setup

We deploy 125 sensor nodes on a 5x5x5 3D grid network. The nodes are arranged in the form of a cube in a 1200x1200x1200 area. The *(x, y, z)* coordinates of the 3D grid range from -600 to 600, where each node is 300 units away from its neighboring nodes.

In order to map this 3D grid onto the surface of the sphere, the radius ρ of the sphere should be kept fixed. We take ρ = 480 so that it could fit inside the cube. We have already proved that the radius of the sphere does not affect our mapping so there is no specific reason for this choice. The centre of the 3D grid and the sphere is same i.e. (0, 0, 0). The transmission range for all nodes is set to 450 which is a little less than the radius of the sphere.

We consider the scenario of all to all communication where, every node has a single packet destined for every other node in the network. In this way, a total of 15625 packets are sourced and sinked. In our results, we calculate the traffic load in the network by ommitting the generated packets at the source nodes and the sink packets at the destination nodes. This experiment is repeated five times and the average of the results is reported.

6.2 Load Distribution

Our main motive is to balance the load in the network so that the nodes at the centre do not run out of their battery power. In SPR the nodes at the periphery are under utilized. But in CSR and SMR, the load is diverged away from the centre towards the periphery in order to balance out the load in the whole network. Although, the load is balanced in CSR, but it also increases the average traffic load in the network. On the other hand, SMR not only balances the traffic evenly among the nodes but also makes sure that the average load on the nodes do not exceed by a certain amount.

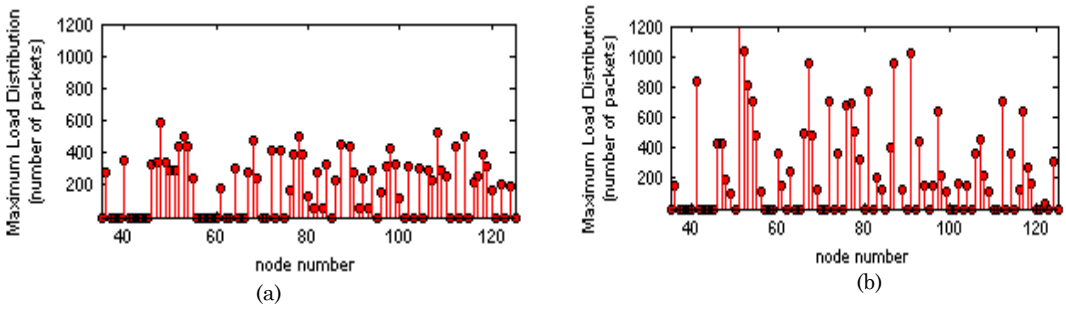


Figure 6. Maximum load distribution in terms of number of packets in the grid network: (a) Load Distribution for SMR (b) Load Distribution for CSR

As shown in Figure6 (a), the maximum routing traffic is no more than 600 packets for SMR and the load is evenly distributed among the nodes. Whereas, in Figure6 (b), it is shown that some nodes need to process a maximum of 1000 packets load in CSR. The reason for this behavior is that, in SMR, the spherical mapping and the distance calculation is done using spherical trignometry. Therefore, the projected nodes on the sphere surface have three or more than three neighbors to forward their routing traffic. We find that only 6 out of 125 nodes have one or two neighbors within their transmission range. On the other hand, CSR mapping and distance calculation for routing results in almost 20 out of 125 projected nodes, with just one or two neighbors within their transmission range. Since the nodes have less choices for routing their traffic, the routing path takes a long route towards the destination, hence, traversing many hops. As a result, most of the nodes in CSR are heavily loaded. In short, SMR balances the traffic more evenly keeping the average load within the limit.

Also the average load at the centre of the network, offered by all the three techniques, is smallest for SMR and greatest for SPR. This is shown in Figure 7. Since the nodes at the centre have least amount of load in the case of SMR, there is a less chance for these nodes to lose their energy completely. Hence, SMR proves to be energy efficient.

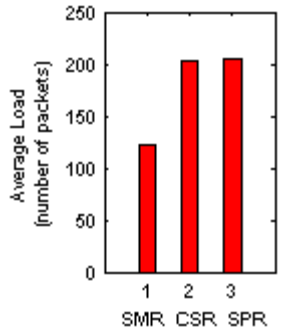


Figure 7 (a). Average load in the centre of the grid network for: (1) SMR (2) CSR (3) SPR

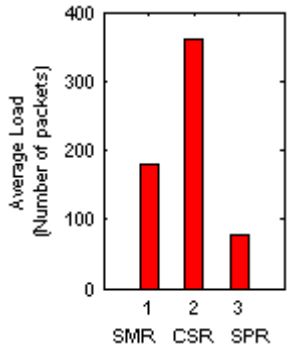
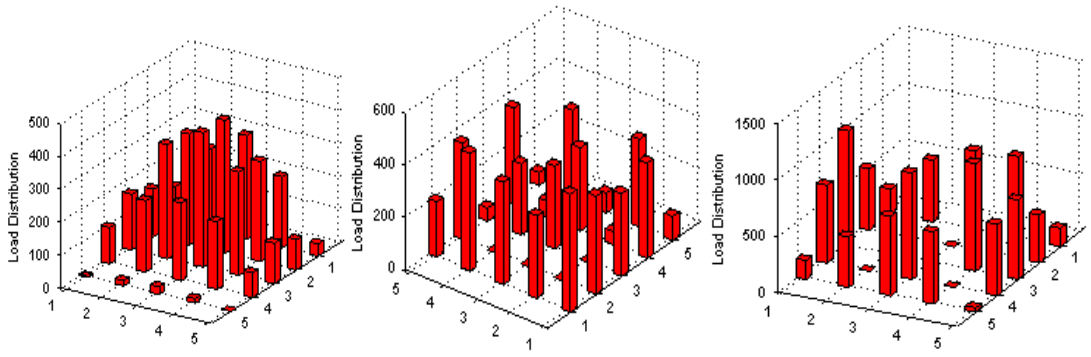


Figure 7 (b). Average load in the boundary of the grid network for: (1) SMR (2) CSR (3) SPR



(a) (b) (c)

Figure 8. Load Distribution in a 2D plane of a grid, obtained by slicing the 3D grid from the centre: (a) Load distribution for SPR (b) Load distribution for SMR (c) Load distribution for CSR

On the contrary, if we look at the load distribution on the boundary nodes of the grid, we come to know that the nodes at the periphery are underutilized in case of SPR. This can easily be shown in Figure 7 (b), where the load offered by SPR on the boundary nodes is just a total of 77 packets which is very less as compared to a total of 213 packets on the nodes at the centre of the grid (refering to Figure 7 (a)). For SMR, both the load on centre nodes and periphery nodes is ccomparable i.e. within a ratio of 120 to 179 packets respectively. But CSR increases the overall average load of the network while balancing the load. For example, it offers a load of almost 362 packets on the periphery nodes and 205 packets on the middle nodes. Thus, it is established that SMR is a better load balancing technique which not only balances the load but also keeps the average load on the overall network within a certain limit.

For Figure 8 (a) to (c), we have cut our grid from the centre, thus obtaining a 2D plane. This plane is a square consisting of 25 nodes. The periphery of the plane has boundary nodes of grid and centre of the plane has center nodes of the grid. We have shown that the load distribution on this plane in Figure 8 (a) to (c) for SPR, SMR and CSR. As we can see in Figure 8 (a), the nodes at the centre of the plane are congested in case of SPR with a maximum load of 400 packets, while the edge nodes have least load on them. On the other hand, SMR and CSR claims to balance the load but in return, they offer higher load at the boundary than the middle region. The maximum load for SMR is approximately 500 packets and there is a small difference between the centre and the edge loads for SMR. However, this difference becomes large enough in case of CSR. Moreover, the maximum load for CSR is highest among all, almost reaching 1200 packets. Therefore, we conclude that SMR outshines CSR in maintaining a reasonable average load in the network.

6.3 Path length and Stretch Factor

In Figure 9, we have shown the comparison of path length versus the node number. The path length is actually the sum of the routing path lengths (in terms of hop count) when a particular node communicates with all the other nodes in the network. It is shown in Figure 9, that in case of SMR, the path length for a single node is 600 hops on average for all to all communication scenario. This is slightly greater than the SPR (where the path length is observed to be 400 hops on average). While in CSR, the path length fluctuates a lot from 600 to 1200 hops on average. We will discuss these results based on our observations in section 6.3. As discussed earlier,the nodes in SMR have more neighbors within their transmission range resulting in small travelled distances (path length) from source to destination. Whereas, the nodes in CSR have less number of neighbors within their transmission range. Since a packet has to traverse many hops in order to reach the destination, the routing path length is increased.

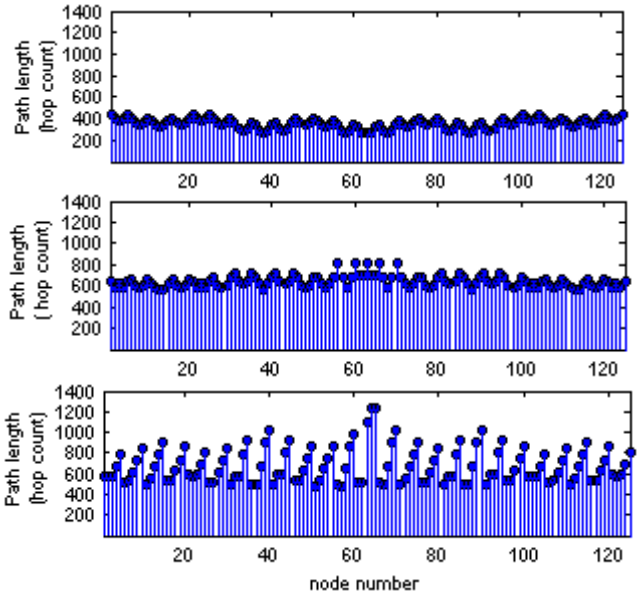


Figure 9. Average path length, interms of hop count, for the grid network, in case of: SPR, SMR, CSR (from top to bottom)

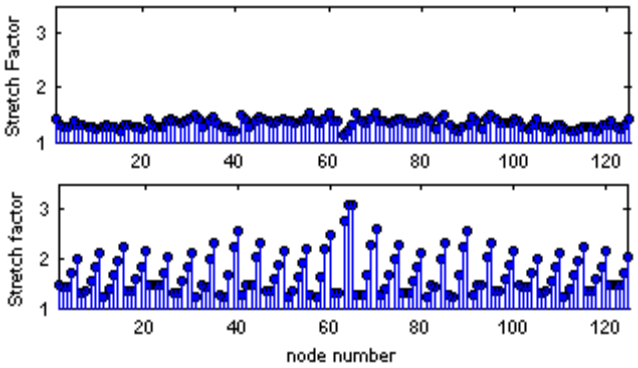


Figure 10. Average stretch factor in case of: SMR, CSR (from top to bottom)

Similarly, in Figure 10, the stretch factor is compared versus the number of nodes. We have already discussed that the number of hops taken by SMR are silghtly greater than the number of hops taken by SPR. Therefore, the stretch factor is within a constant average limit of 1.4 for SMR. Since the path length of CSR is greater and highly fluctuating as compared to SPR, CSR offers an average stretch factor of 1.9.

7. CONCLUSION

The load balancing issue in WSNs is addressed in this paper because uneven load distribution in a network could cause some serious problems e.g., packet delay, packet loss, node battery failure and network security related threats. Some protocols have already been put forward in this regard but they solve the problem of load baalncing with a cost of increased path length and increased overall average load in the network. We have addressed all of the above issues by proposing *Spherical Mapping based Routing (SMR)* approach. Our protocol, SMR first maps the entire 3D grid to a sphere and then routes in the spherical domain using a spherical distance as the routing metric. Also, it not only distributes the traffic evenly in the network but also manages to keep the overall average load in the network below a certain level. Moreover, it offers bounded stretch factor which makes it a better choice for load balancing than the previous protocols.

We have evaluated SMR, both analytically and through simulation. The approaches we used for comparison are SPR and CSR. Our results show that in SPR, the load at the centre nodes of the network is highest and it gradually decreases towards the boundary nodes of the network. Whereas, the load on the nodes in CSR is almost balanced but this load is almost double than the maximum load in SPR. On the other hand, SMR balances the load in the network with just a small overhead of increased average load. The path length in CSR, for some cases, is very large as compared to SPR shortest path length. But the path length for SMR closely follows the path length of SPR within a constant limit. Therefore, both types of our results show that SMR achieves its desired goals to maintain the average load within the limit and provides a constant in bound stretch factor.

REFERENCES

Jun-Hong Cui, Jiejun Kong,Mario Gerla, Shengli Zhou, "Challenges: Building Scalable Mobile Underwater Wireless

Sensor Networks for Aquatic Applications", IEEE Network 2006

Fan Li, Yu Wang, “Circular Sailing Routing for Wireless Networks”, in the IEEE INFOCOM 2008 proceedings.

L. Popa, A. Rostamizadeh, et al., “Balancing traffic load in wireless networks with curveball routing,” in Proc. of ACM MobiHoc, 2007.

Y. Yu, R. Govindan, and D. Estrin, “Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks,” UCLA Computer Science Department, Tech. Rep. UCLA/CSD-TR-01-0023, May 2001.

M. Zorzi and R. R. Rao, “Geographic random forwarding (geraf) for ad hoc and sensor networks: Energy and latency performance”, IEEE Transactions on Mobile Computing, vol. 2, no. 4, pp. 349–365, 2003.

Rao A. Ratnasamy S., Papadimitriou C., Shenker S., Stoica I., "Geographic Routing without Location Information", appeared in the proceeding of ACM Mobicom, 2003.

J.Bruck, J. Gao, A. Jiang, “MAP: Medial Axis Based Geometric Routing in Sensor Networks”, in the Proc. Of ACM Mobicom, 2005.

B. Karp and H. T. Kung, “GPSR: Greedy perimeter stateless routing for wireless networks,” in Proceedings of the 6th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom’00), Boston, MA, USA, August 2000.

I. F. Akyildiz, D. Pompili, and T.Melodia, “Underwater acoustic sensor networks: research challenges,” Ad Hoc Networks, vol. 3, no. 3, pp. 257–279, 2005.

Sung-Ju Lee and Mario Gerla, “Dynamic load-aware routing in ad hoc networks,” in Proc. of IEEE ICC, 2001.

Lei Yung, R. Srikant, Don Towsley, “Cluster-Based Back-Pressure Routing algorithm” in the IEEE Infocom 2008 proceedings.

S.-B. Lee, J. Cho, and A. T. Campbell, “A hotspot mitigation protocol for ad hoc networks,” Ad Hoc Networks, vol. 1, no. 1, pp. 87–106, 2003.

M.R. Pearlman, et al., “On the impact of alternate path routing for load balancing in mobile ad hoc networks,” in Proc. of ACM MobiHoc, 2000.

Y. Ganjali and A. Keshavarzian, “Load balancing in ad hoc networks: single-path routing vs. multi-path routing,” in Proc. Of IEEE Infocom, 2004.

W. Wang, X. Li, and Y. Wang, “Truthful multicast routing in selfish wireless networks,” in Proceedings of the 10th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom’04), Philadelphia, PA, USA, September 2004.

S. Kwon and N. B. Shroff, “Paradox of shortest path routing for large multi-hop wireless networks,” in the IEEE INFOCOM 2007 Anchorage.

Wan C.Y. Eisenman S.B., Campbell A.T., "CODA: Congestion Detection and Avoidance in Sensor Networks", appeared in the proceedings of ACM SenSys 2003.

R. Sarkar, X. Zhu, and J. Gao, “Double rulings for information brokerage in sensor networks,” in Proceedings of the 12th Annual International Conference on Mobile Computing and Networking (MOBICOM ’06), pp. 286–297, Los Angeles, USA, September 2006.

Jie Gao and Li Zhang, “Load balanced short path routing in Wireless Networks”, in the IEEE INFOCOM 2004 proceedings.

Alessandro Mei, Julinda Stefa, “Routing in Outer Space”, in the IEEE INFOCOM 2008 proceedings.

Geographic coordinate system, DOI: http://en.wikipedia.org/wiki/Geographic_coordinate_system.

Great-circle distance DOI: http://en.wikipedia.org/wiki/Great-circle_distance.

J. Gao and L. Zhang, “Tradeoffs between stretch factor and load balancing ratio in routing on growth restricted graphs,” in Proc. Of ACM PODC, 2004.