Parabolic Yao-type Geometric Spanners in Wireless Ad Hoc Networks

Ojenie Artoun

A Thesis
in
The Department
of
Computer Science and Software Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Computer Science at
Concordia University
Montreal, Quebec, Canada

January 2010

CONCORDIA UNIVERSITY

School of Graduate Studies

This is to certify that the thesis prepared

By: Ojenie Artoun

Entitled: Parabolic Yao-type Geometric Spanners in Wireless Ad Hoc Networks

and submitted in partial fulfillment of the requirements for the degree of

Master of Computer Science

complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the final examining committee:

____ Chair
Dr. N. Shiri

____ Examiner
Dr. H. Harutyunyan

____ Examiner
Dr. J. Opatrny

____ Supervisor
Dr. T. Fevens

Approved by ____
Chair of Department or Graduate Program Director

____
Dr. Robin A. L. Drew, Dean
Faculty of Engineering and Computer Science

Date ____

Abstract

Parabolic Yao-type Geometric Spanners in Wireless Ad Hoc Networks

Ojenie Artoun

Geometric graphs are frequently used to model a wireless ad hoc network in order to build efficient routing algorithms. The network topology in mobile wireless networks may often change therefore position-based routing that uses the idea of localized routing has an advantage over other types of routing protocols. Since a wireless network has limited memory and energy resources, topology control has an important role in enhancing certain desirable properties of these networks. To achieve the goal of topology control, spanning subgraphs of the UDG graph such as Relative Neighborhood Graph, Gabriel Graph and Yao Graph are constructed, which are then routed upon rather than the original UDG.

In this thesis we introduce a spanning subgraph of the UDG graph that is a variation of the Displaced Apex Adaptive Yao ($DAAY$) graph. In this subgraph an exclusion zone based on a parabola is defined with respect to each non-excluded nearest neighbor and positioned on each node, instead of a cone as with the $DAAY$ subgraph, such that each nearest neighbor is inside the parabola. It also lets the apex of the parabola to move along the line segment between the node and its neighbor. The subgraph has two adjustable parameters, one each for the position of the apex with respect to the nearest neighbor, and the width of the parabola. Thus a directed or undirected spanning subgraph of a UDG is constructed. We show that this spanning subgraph is connected, has a conditional bounded out-degree, is a t-spanner with bounded stretch factor, and contains the Euclidean minimum spanning tree as a subgraph. Experimental comparisons with related spanning subgraphs are also presented.

Acknowledgements

First and foremost, I want to express my gratitude to my supervisor Dr. Fevens for his guidance and motivation. I am sincerely in debt to him for his patience and for allowing me to complete many courses before starting this project, and for his continuous support. I was really lucky to have him as my supervisor and it was for his valuable advices that today I am able to finish this thesis.

I would also like to extend my thanks to Dr. Harutyunyan and Dr. Opatrny for taking the time to review my thesis and serve on my committee.

Although I have been far from my family, they have always been with me in my mind, pushing me forward. My sister, Artemis and my brother, Narbeh, I am grateful for their support.

I would like to thank my friends at Concordia who have helped me with the course projects. I wish them all the success in the future.

To my best friend in the world, Peyman.
I would not achieve this goal without your support.

Contents

List of Figures

List of Tables

Chapter 1

Motivations

1.1 Wireless ad hoc networks

A *wireless ad hoc network* consists of a collection of wireless hosts that can communicate with each other without the use of a network infrastructure or any centralized administration. Since wireless ad hoc networks can be easily and inexpensively set up as needed, they offer solutions to a variety of applications, especially in situations with geographical or logistical constraints such as battlefields, military applications, and other emergency and disaster situations.

Wireless ad hoc networks can be subdivided into two classes: *static* and *mobile*. In static networks, the position of a wireless node does not change or changes very slowly once the node is deployed; an example is sensor networks. In mobile networks or MANETs (Mobile Ad hoc NETworks), wireless nodes may move freely. Mobile wireless networks change their topology frequently and often without any regular pattern, therefore routing in these networks is challenging and of great importance. In this thesis our focus is on MANETs and for simplicity we assume that the nodes are quasi-static during the short period of topology reconstruction or route finding.

In addition, we assume that the position information of each node can be obtained by a low-power Global Position System (GPS) receiver at the node. There are many techniques to estimate the position of the nodes or the distance between neighboring nodes if GPS is not available [9].

We consider a wireless ad hoc network consisting of a set V of N wireless nodes distributed in the Euclidean two-dimensional plane $\mathbb{R}^2$. Each node has a transmission range r, which we assume is a fixed number, that can be represented as a disk centered at the node. These wireless nodes can be modeled by a *unit disk graph* $UDG(V)$ in which there is an edge between two nodes if and only if their Euclidean distance is at most r, the transmission range of the node [2, 3]. Node v can receive the signal from node u if node v is within the transmission range of node u. If nodes u and v are not able to communicate directly then node u uses multi-hop wireless links through intermediate nodes to send packets to node v.

Topology control in wireless ad hoc networks is to maintain a connected topology among the network nodes by reducing number of the active nodes and links. Topology control in a wireless ad hoc network may require construction of a *t-spanner* subgraph of the network. A *spanning subgraph* of unit disk graph is a subgraph that connects all the nodes in the graph. If the shortest path between any two nodes in the subgraph is not t times, $t \in \mathbb{R}$, longer than the shortest path between those two nodes in the original unit disk graph then that spanning subgraph is called a t-spanner of the unit disk graph. A t-spanner with linear number of links is called a *sparse spanner*. Such spanning subgraphs are used in many routing algorithms.

1.2 Routing in wireless ad hoc networks

Routing in wireless ad hoc networks generally involves multiple hops because of the limited transmission range. In such a network, each node operates not only as a host but also as a router, forwarding packets for other nodes which may not be within direct wireless transmission range of their destination node. Routing in ad hoc networks can be challenging because of node mobility, lack of a predefined infrastructure, the peer-to-peer mode of communication and limited transmission range. Also some or all of the nodes in a wireless ad hoc network might rely on batteries, and power and computation constraints could force a node to exhaust its energy. Because of the mentioned limitations, it is more challenging to design a network topology based on a subgraph for wireless ad hoc networks that is suitable for designing or using a routing algorithm which uses less power and has less memory consumption, than the traditional wired networks.

The goal of routing in wireless ad hoc networks is to find a path from the source node to the destination node, and to deliver packets to the destination node. Since nodes in the network may move freely, a path discovered by a source may not exist after a short interval of time. This requires the nodes to discover and maintain routes in the network. There are different routing protocols for wireless ad hoc networks depending on how nodes establish and maintain paths. They mainly may be devided into *topology-based* routing protocols and *position-based* or *location-based* routing protocols [27, 14] categories. Topology-based routing protocols are mainly subcategorized as *pro-active* [30], *reactive* [16, 31] and *hybrid* [15] routing protocols.

Pro-active routing protocols are table-driven protocols. They maintain a routing table using the routing information learnt from their neighbors and they update it continuously. Examples of this type of protocols are the Destination-Sequenced Distance-Vector (DSDV) protocol [30] and the Wireless Routing Protocol (WRP)[28]. The main drawback to pro-active routing protocols is that because of the node mobility, routing tables are updated frequently and these protocols suffer from overhead due to route maintenance.

Reactive routing protocols are demand-driven protocols. They find a path when it is necessary. Examples of this type of protocol are the Ad-hoc On-demand Distance Vector Routing (AODV)[31] and the Dynamic Source Routing (DSR) protocols [3, 16]. The main drawback of reactive routing is that because of the on-the-fly route discovery there is a delay in delivering the packets to the destination.

Hybrid protocols combine the advantages of various approaches of local pro-active routing and global reactive routing into a single protocol which achieves a more scalable and efficient protocol. An example of this type of protocol is the Zone Routing Protocol (ZRP)[15]. The drawback to hybrid routing is that route maintenance is still needed.

Position-based routing protocols use position information of nodes in the network and the global topology information of the network has very little use. Therefore frequent changes of the network topology has no effect on such routing protocols, thus making position-based routing a good candidate for wireless ad hoc networks. Position-based routing in wireless ad hoc networks uses the idea of *localized routing* where each node maintains only local e.g., one or two hop, topology information. This is an advantage since nodes in wireless ad hoc networks frequently have scarce storage resources and limited power. More importantly when the network topology changes no more computation is needed to maintain a routing table. Localized routing has the following desirable characteristics. The number of nodes in a network can increase since there is no routing table to be updated. Therefore localized routing is scalable. Every node in the network performs the same routing protocol in deciding to which neighboring node to forward the packet. Therefore localized routing is uniform.

1.3 Thesis contribution

When modeling a wireless ad hoc network using the unit disk graph, the size of the graphs could be as large as the square order of the number of network nodes. Since the goal of designing a network is to achieve desirable network features such as bounded node degree, low-stretch factor, and also facilitate the use of attractive routing schemes such as localized routing with guaranteed delivery, we want to construct a subgraph of the unit disk graph which is sparse, can be constructed locally, and is relatively good in terms of the length of possible paths compared with the original unit disk graph.

We introduce a new spanning subgraph of a unit disk graph, that is a variation of Displaced Apex Adaptive Yao Graph $DAAY$ [12]. This new graph replaces the cones in $DAAY$ with parabolas and locally constructs a directed or undirected spanning subgraph of a UDG. We show that this new subgraph is connected, has a conditionally bounded out-degree, is a t-spanner with bounded stretch factor, and contains the Euclidean minimum spanning tree as a subgraph.

1.4 Thesis organization

This thesis consists of five chapters. The present chapter (Chapter 1) provided the background and motivations of this study.

In Chapter 2, a systematic literature review on the most relevant contributions to the spanner graphs is presented. In addition, advantages and disadvantages of the use of these graphs to model wireless ad hoc networks are discussed.

In Chapter 3, a new geometric subgraph is introduced which is a variation of $DAAY$ graph. The properties of the new subgraph are also proved.

Chapter 4 presents a large set of numerical examples for the new subgraph and the results are compared with some of other known subgraphs.

Finally, in Chapter 5, conclusions, contributions and future work are presented.

Chapter 2

Spanning graphs

2.1 Preliminaries

A graph G is defined as $G = (V, E)$ where V is the set of N vertices $v_1, v_2, \ldots, v_N$ and E is the set of pairs of vertices, called edges, and is a subset of cartesian product $V \times V$. A node, or vertex, u is a neighbor of node v if there exists a pair $(u, v) \in E$, such that there is an edge between u and v in the graph G. $N(u)$ is the set of all neighboring nodes of node u. A *geometric graph* is a graph in which the vertices are points in the Euclidean plane and edges are straight line segments between vertices. A geometric graph is *planar* if no edges cross each other.

A *Unit Disk Graph* or $UDG(V)$ is a geometric graph such that for each pair of nodes $u, v \in V$ there exists an edge $(u, v) \in UDG(V)$ if and only if $dist(u, v) \leq 1$. In our wireless model there exists an edge between two nodes in $UDG(V)$ if and only if $dist(u, v) \leq r$ where r is the transmission range of the nodes in V.

The *degree* of a node u is the number of edges connecting u to its neighbors, or simply $|N(u)|$. A *Euclidean graph* is a weighted graph $G = (V, E)$ in which V is a set of points in the plane, and each edge $(u, v) \in E$ is assigned a weight equal to the

Euclidean distance between u and v, $dist(u, v)$, which we denote as $|uv|$.

$$dist(u, v) = \sqrt{(u_x - v_x)^2 + (u_y - v_y)^2}$$

Weight of a graph G is defined as the sum of edge lengths for all edges in E.

A *subgraph* of a graph is a graph with vertices and edges as subsets of vertices and edges of the original graph. A path from a source node, s, to a destination node, d, in a graph is a sequence of nodes $s, v_1, v_2, \ldots, v_k, d$ where v_i and v_{i+1} are neighbors; for $i = 1, \ldots, k - 1$. A graph is *connected* if there is a path between any two nodes in the graph. A *strongly connected* graph is a directed graph in which there is a path between any two nodes by traversing the edges in the directions that they point. A *spanning subgraph* of a graph is a graph has all vertices in the original graph and its edges is a subset of edges in the original graph. For a geometric graph G a *Euclidean Minimum Spanning Tree* or $EMST(G)$ is a spanning tree of G with minimum weight. A *t-spanner* of graph G is a spanning subgraph on V that connects all nodes and the length of the shortest path between any two nodes in S is at most t times the length of the shortest path between the same nodes in G. Parameter t is called the *stretch factor*.

There are several possible definitions of the length of a path. An important length definition is based on considering the power consumed during routing along a path. Suppose $u = v_0, \ldots, v_i, \ldots, v_m = v$ is the shortest path from u to v in subgraph S of original graph G, define $p_S(u, v) = \sum_{i=0}^{m} |v_{i-1}v_i|^\beta$ as the total transmission power consumed by the path. Let $u = w_0, \ldots, w_j, \ldots, w_n = v$ be the shortest path from u to v in graph G, define $p_G(u, v) = \sum_{j=0}^{n} |w_{j-1}w_j|^\beta$ as the total transmission power consumed by the path in G. Here β is a constant between 2 and 5 depending on the wireless transmission range ([24]) and is called path loss exponent. The *power stretch factor* (distance dilation ratio), $\rho_S^p(G)$, of the subgraph S is defined as follows ([21]):

$$\rho_S^p(G) = \max_{u,v \in V} \frac{p_S(u,v)}{p_G(u,v)} \tag{2.1}$$

If $\beta = 1$ in the above, we obtain the *length stretch factor* based on simply summing the Euclidean lengths of the edges of the path.

The stretch factor can also be measured by number of hops in the path. Let $l_S(u,v)$ be the number of edges in the shortest path from u to v in subgraph S of original graph G and let $l_G(u,v)$ be the number of edges in the shortest path from u to v in G. Then *hop stretch factor* (hop dilation ratio), parameter t above, is defined as follows:

$$\rho_S^l(G) = \max_{u,v \in V} \frac{l_S(u,v)}{l_G(u,v)} \tag{2.2}$$

A t-spanning path from u to v is *strong* if the length of every edge in the path is at most $|uv|$. The graph G is a *strong t-spanner* if there is a strong t-spanning path between every pair of nodes in G.

2.2 Topology control

Since wireless ad hoc networks may be powered by a limited power supply like batteries, typically have small memory, and the network topology may change due to node mobility, an important goal of topology control is to maintain connectivity and to increase network lifetime. A basic requirement to achieve these goals is to construct a spanning subgraph of UDG. Because unit disk graphs can be very dense depending on the number of nodes in the network, constructing subgraphs that, in particular, are sparse, power-efficient which is measured by stretch factor, and are locally constructed (with the information of neighbors at one or two hops distance) so that changes of network topology are easily handled, are of great interest.(see [22, 32] for surveys).

We are particularly interested in spanning subgraphs that are sparse with low number of edges in the graph, have low stretch factor to be power-efficient, have bounded degree meaning each node in the subgraph has a degree less than a constant fixed number, are planar or nearly planar and can be constructed locally. Many of spanning subgraphs have been introduced (see surveys [27],[32]) and each one achieves one or some of those objectives.

2.3 Geometric structures

There have been an extensive study on geometric spanners of the UDG for ad hoc networks and many subgraphs have been introduced. Dobkin et al. [11] showed that the Delaunay graph is a planar geometric spanner of the Euclidean graph. Bose et al. [7] introduced a subgraph of the Delaunay graph that is a geometric spanner of the Euclidean graph. Recently Kanj and Perkovic [18] improved this subgraph and constructed a subgraph of Delaunay graph with bounded degree and stretch factor which implies a planar geometric spanner of a Euclidean graph. Li et al. [23] gave a distributed algorithm that constructs a planar geometric spanner of a unit disk graph. Wang and Li [35] introduced a bounded-degree planar spanner of a unit disk graph. A comprehensive survey on different methods for constructing t-spanners with various properties is given in [29].

In the following we present some of the geometric spanning subgraphs of a UDG that are commonly used with position-based routing and are the most relevant to our work.

2.3.1 Relative neighborhood graph

Consider two circles, one centered at node u, C_u, and other at node v, C_v, with radius $|uv|$. An edge (u, v) from graph G is in the relative neighborhood graph of G[60], denoted $RNG(G)$, if and only if the intersection of two circles does not contain any other node $w \in V$. See Figure 2.1(a). This spanning graph has been introduced by Toussaint ([34]) and it has been shown that $RNG(G)$ is planar and a t-spanner with power stretch factor of $N - 1$.

2.3.2 Gabriel graph

Let $disk(u, v)$ be a circle with diameter $|uv|$ and centered at the midway point of edge (u, v). An edge (u, v) from the graph G is in the Gabriel graph of G[45], denoted $GG(G)$, if and only if $disk(u, v)$ does not contain any other node $w \in V$. See Figure 2.1(b). Bose *et al.*[5] have shown that $GG(G)$ is planar and a $\frac{4\pi\sqrt{2N-4}}{3}$-spanner. Furthermore Li *et al.*[24] prove that the power stretch factor of any Gabriel subgraph is one.

2.3.3 Yao graph

At each node u, define k cones, $k \geq 6$, by k equally-separated rays originating at u. A directed edge $\overrightarrow{(u, v)}$ is in the directed Yao graph of G, denoted $\overrightarrow{YG_k}(G)$, if and only

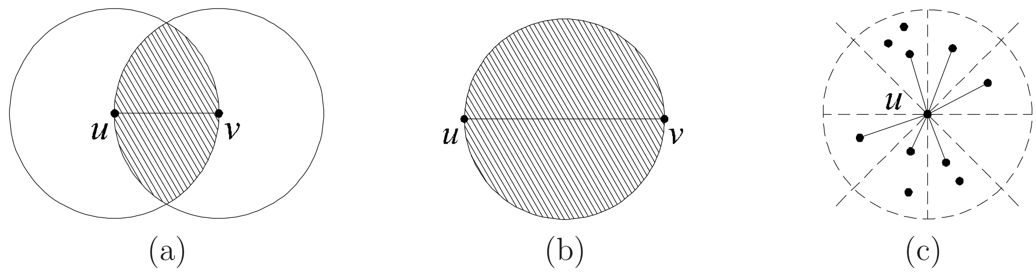


Figure 2.1: (a)Relative Neighborhood Graph ;(b)Gabriel Graph ;(c)Yao Graph

if $\overrightarrow{(u,v)}$ is the shortest edge, if any, in each cone. Ties are broken arbitrarily [37]. Let $YG_k(G)$ be the undirected graph obtained by ignoring the direction of edges in $\overrightarrow{YG_k}(G)$. See Figure 2.1(c). A similar construction of $YG_k(G)$ is also called a *Theta Graph* [6]. $YG_k(G)$ may have crossing edges if G is a UDG. Yao graph has an out-degree of at most k, contains the $EMST(G)$ as a subgraph, is a $\frac{1}{1-2\sin\frac{\pi}{k}}$-spanner and is connected if $UDG(G)$ is connected [20, 24]. Li *et al.* [24] prove that the power stretch factor of Yao graph is at most $\frac{1}{1-(2\sin\frac{\pi}{k})^{\beta}}$.

2.3.4 Gabriel-Yao graph and Yao-Gabriel graph

Li *et al.*[25] proposed two structures by combining Gabriel graph and Yao graph that are sparser than the original two graphs. In $GYG_k(G)$ Gabriel graph is applied on top of Yao graph structure and in $YGG(G)$ Yao graph is applied on top of Gabriel graph structure. Both graphs are connected if the $UDG(G)$ is connected, have out-degree of at most k and have power stretch factor of $\frac{1}{1-(2\sin\frac{\pi}{k})^{\beta}}$.

2.3.5 Half space proximal graph

For every node $u \in V$ define an open half plane $H(u,z)$ containing z, neighbor of u, with the boundary line perpendicular to and at the midway point of $\overrightarrow{(u,z)}$. A directed Half Space Proximal graph[31], denoted $\overrightarrow{HSP}(G)$ is constructed by the

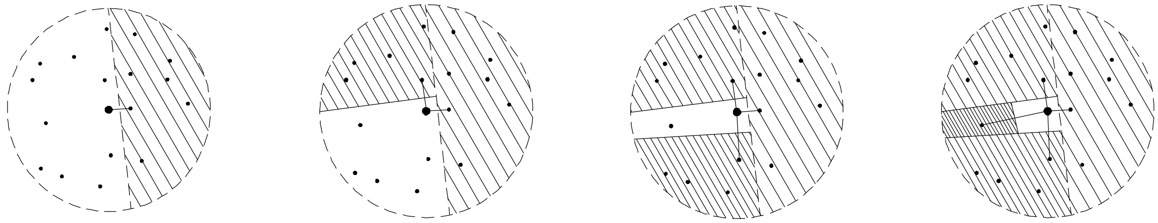


Figure 2.2: Constructing the HSP graph in a UDG. (a) An edge is selected to the first closest neighbor, (b) and (c) next closest neighbors are selected from outside the half plane, (d) until list of neighbors is empty.

following iterative steps [10]:

for all $u \in V$ **do**

Create a list of neighbors of u: $LN(u) = N(u)$.

repeat

Find the nearest neighbor z and add edge $\overrightarrow{(u,z)}$ to $\overrightarrow{HSP}(G)$.

Define $H(u, z)$ and discard all the nodes in it from $LN(u)$.

until $LN(u)$ is empty

end for

Let $HSP(G)$ be the undirected graph obtained by ignoring the direction of edges. See Figure 2.2. $HSP(G)$ is strongly connected, has an out-degree of at most six and contains $EMST(G)$ as its subgraph. An $HSP(G)$ is not planar and may have crossing edges.

2.3.6 The family of G_λ^θ graphs

Bose *et al.*[4] introduces a family of G_λ^θ graphs that are related to the HSP, and that depend on two parameters λ and θ. Let $0 \le \theta < \pi/2$ and $1/2 < \lambda < 1$. Define $C(u, z, \theta)$, the cone of angle 2θ with apex u and edge (u, z) as its bisector. Define

$H(u, z, \lambda)$, the half plane containing z with the boundary line perpendicular to (u, z) and at the distance of $\frac{1}{2\lambda}|uz|$ from u. Now define the $K(u, z, \theta, \lambda)$, *destruction region of z with respect to u* as the intersection of $C(u, z, \theta)$ and $H(u, z, \lambda)$. G^{θ}_{λ} is constructed by the following iterative steps:

for all $u \in V$ **do**

Create a list of neighbors of u: $LN(u) = N(u)$.

repeat

Find the nearest neighbor z and add edge (u, z) to G^{θ}_{λ}.

Define $K(u, z, \theta, \lambda)$ and discard all the nodes in it from $LN(u)$.

until $LN(u)$ is empty

end for

G^{θ}_{λ} has an out-degree of at most $\lfloor(2\pi/(\min(\theta, \arccos(1/2\lambda)))\rfloor$ and it is a strong t-spanner, with $t = 1/((1 - \lambda)\cos(\theta))$.

2.3.7 Displaced apex adaptive Yao graph

The Displaced Apex Adaptive Yao graph [12], denoted $DAAY(G, \alpha, p)$, is a class of orientation-invariant Yao-type graphs that includes the HSP graph as a special case. For every $u \in V$ define cone $C(u, z, \theta)$ the cone of interior angle 2θ with edge $\overrightarrow{(u, z)}$ as its bisector. The parameter α, $0 \leq \alpha \leq 1$, determines θ as a fraction of a maximum cone angle, which we define below. Let $(1 - p)u + pz$ be a parametrization of line segment between u and z, $0 \leq p \leq 1$. The position of apex of the cone is represented by choosing a particular p. The directed displaced apex adaptive Yao graph, $\overrightarrow{DAAY}(G, \alpha, p)$ is constructed by the following iterative steps:

for all $u \in V$ **do**

Create a list of neighbors of u: $LN(u) = N(u)$.

repeat

Find the nearest neighbor z and add edge $\overrightarrow{(u,z)}$ to $\overrightarrow{DAAY}(G, \alpha, p)$.

Define $C(u, z, \theta)$ and discard all the nodes in it from $LN(u)$.

until $LN(u)$ is empty

end for

Let $DAAY(G, \alpha, p)$ be the undirected graph obtained by ignoring the direction of edges. See Figure 2.3. Since this thesis work is a variation of $DAAY$, we describe it in more detail.

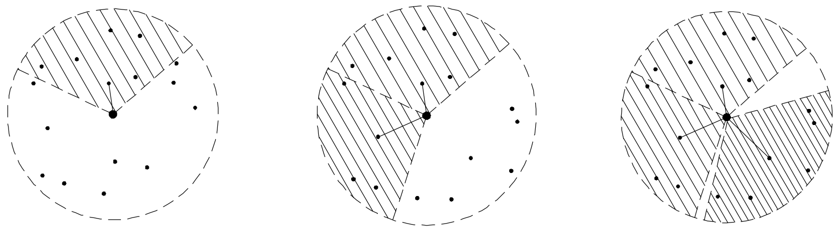
The maximum cone angle, $\theta_m(p, |uz|)$, is a function of p and $|uz|$, and is defined as follows:

$$\begin{cases} \dfrac{\sin\left(\theta_m(p,|uz|) - \frac{\pi}{3}\right)}{\sin(\theta_m(s,|uz|))} = p & \text{if } 0 \le p < 0.5 \\ \dfrac{\sin\left(\theta_m(p,|uz|) - \cos^{-1}\left(\frac{1}{2}\frac{|uz|}{r}\right)\right)}{\sin(\theta_m(s,|uz|))} = \dfrac{p|uz|}{r} & \text{if } 0.5 \le s \le 1 \end{cases}$$

For $p = 0.5$, maximum cone angle is $\theta_m(0.5, |uz|) = \pi/2$ and if $\alpha = 1$ then $\theta = \theta_m(p, |uz|)$, therefore we obtain the Half Space Proximal graph. The main properties of DAAY are listed below. Proofs are presented in [12, 1].

Property 2.3.1. *Consider a node set V and $UDG(V)$ defined on V. If $UDG(V)$ is connected and the cone angle θ is less than or equal to $\theta_m(p, |uz|)$ then $DAAY(UDG(V, \alpha, p)$ is strongly connected.*

Property 2.3.2. *The out-degree of any node in $DAAY(UDG(V), \alpha, p)$, $\theta = \alpha \cdot \theta_m(p, |uz|)$, $0 \le \alpha \le 1$, is at most $\left\lfloor \dfrac{2\pi}{\phi} \right\rfloor$ where ϕ is defined by $\dfrac{\sin(\theta - \phi)}{\sin(\theta)} = p$.*



(a) (b) (c)

Figure 2.3: Constructing DAAY graph in a UDG. (a)An edge is selected to the first closest neighbor, (b) next closest neighbor is selected from outside the cone, (c) until list of neighbors is empty.

Property 2.3.3. *Let $V \subseteq \mathbb{R}^2$ be a set of N points and let $\theta < \pi/3$ be the cone angle. Then $DAAY(UDG(V), \alpha, p)$ is a spanner with hop stretch factor* $\frac{1}{1 - 2\sin(\frac{\theta}{2})}$.

Property 2.3.4. *Let $V \subseteq \mathbb{R}^2$ be a set of N points and let $\theta < \pi/3$ be the cone angle. Then $DAAY(UDG(V), \alpha, p)$ is a spanner with power stretch factor* $\frac{1}{1 - (2\sin(\frac{\theta}{2}))^{\beta}}$.

Property 2.3.5. *Consider a node set V and $UDG(V)$ defined on V. Assume that $UDG(V)$ is connected. Then $DAAY(UDG(V), \alpha, p)$, $\theta = \alpha \cdot \theta_m(p, |uz|)$, $0 \leq \alpha \leq 1$, contains the Euclidean Minimum Spanning Tree $EMST(UDG(V))$ as a subgraph.*

2.4 Summary

Although the relative neighborhood graph and Gabriel graph are planar graphs, they have high stretch factor. The Yao graph has the advantage of controlling the node out-degree and has lower stretch factor in addition to a sparse subgraph. The GYG and YGG do not make an improvement in node degree but constructed subgraph is sparser. HSP has a bounded node degree and it is a sparse graph. $DAAY$ combines the advantages of both the HSP subgraph and the Yao subgraph by permitting control over the degree of the subgraph while also being orientation-invariant. Therefore our focus is on $DAAY$ and in the next chapter we introduce a modified version.

Chapter 3

Adaptive parabolic Yao graph

In this chapter, we introduce a variation of the $DAAY$ graph called the Adaptive Parabolic Yao graph, which is an undirected subgraph of an undirected unit disk graph. It uses parabolas, instead of cones as with $DAAY$, positioned on each node such that nearest neighbor is centered "inside" the parabola. Like $DAAY$, the apex of the parabola is permitted to move along the line segment between the node and its neighbor.

3.1 Definition

Let V be a set of N points in the Euclidean two-dimensional plane. Consider vertices $u, z \in V$, such that z is a neighbor of u. Without loss of generality (WLOG), we will assume that u is positioned at the origin of the Cartesian coordinate system and its neighbor z is positioned along the positive x-axis at the point $(|x|, 0)$. Let $(1-\alpha)u + \alpha z$ with $0 \leq \alpha \leq 1$ be a parametrization of the line segment between u and z, then any choice of α represents a position b which we will define as apex of the parabola (note that the apex, or vertex, is the intersection point between the parabola and its axis of symmetry). Consider parameter γ, a real number in $\{0\} \cup \mathbb{R}^+$.

We define a to be the *width of the parabola*, with respect to a_m, the *maximum width of the parabola* (a function of α and $|uz|$) which we will define later in Eq. 3.2 on page 22, as follows:

$$a = \begin{cases} (1+\gamma)a_m & \text{if } 0 \leq \alpha < 0.5 \\ \gamma \cdot a_m & \text{if } \alpha = 0.5 \\ (1-\gamma)a_m & \text{if } 0.5 < \alpha \leq 1 \end{cases} \tag{3.1}$$

where variable γ defines different values of a for a given parabola. Now the parabola P can be defined by $x = ay^2 + b$. By changing α the parabola moves along the line segment uz. As the value of γ increases, the value of the width of parabola a increases, towards positive infinity, from its minimum value a_m. A point $v(x_v, y_v)$ is *inside* the parabola if and only if $(x_v - a{y_v}^2 - b) > 0$. See Figures 3.1, 3.2 and 3.3. To define the parabola in the form of $x = ay^2 + b$, positions of all nodes in the network are an affine transformation of the original positions of the nodes such that nodes have first a translation and then a rotation by θ, the angle between the new and original x positive axes. This transformation does not change the topology of the network, with respect to position of the nodes relative to each other.

Algorithm 1 Adaptive Parabolic Yao(G, α, γ) graph algorithm

Input: *A graph G with the node set V, width parameter γ, and parameter α.*
Output: *A list of directed edges L for each node $u \in V$ which represent the Adaptive Parabolic Yao subgraph of G, $APY(G, \alpha, \gamma)$.*
for all $u \in V$ **do**
  Create a list of neighbors of u: $LN(u) = N(u)$.
  repeat
    (a) Remove the nearest neighbor node z from $LN(u)$ (ties broken arbitrarily) and add the directed edge uz to L.
    (b) Determine $a_m(\alpha, |uz|)$ such that a is calculated by equation 3.1.
    (c) Let $b = (1-\alpha)u + \alpha z$ be a point on the line segment uz.
    (d) Consider the parabola P with its apex at b with width a and z in its interior, such that the line uz bisects the parabola P into two equal halves.
    (e) Scan the list $LN(u)$ and remove each node in the interior of P from $LN(u)$.
  until $LN(u)$ is empty
end for

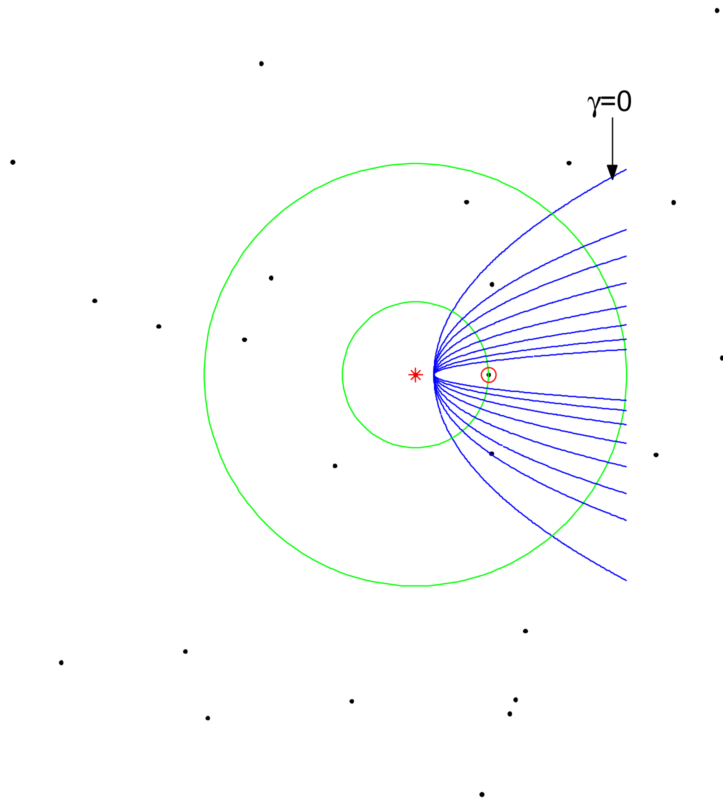


Figure 3.1: Definition of parabola when $\alpha < 0.5$. This picture shows $x = (1+\gamma)a_m y^2 + b$ family of parabolas with $\alpha = 0.25$ and $\gamma = 0, 1, 2, 4, ..., 64$. The value of a is positive and as the value of γ increases the parabola gets narrower.

Definition 3.1.1. *Let G be a UDG with node set V. The directed Adaptive Parabolic Yao subgraph, $\overrightarrow{APY}(G, \alpha, \gamma)$, is defined to be the graph with node set V whose edges are obtained by applying the Adaptive Parabolic Yao(G, α, γ) algorithm, Algorithm 1 given on page 18, on the graph G using parabola width a(Eq. 3.1), and apex position b. The undirected graph $APY(G, \alpha, \gamma)$ is obtained by ignoring the direction of the directed edges in $\overrightarrow{APY}(G, \alpha, \gamma)$.*

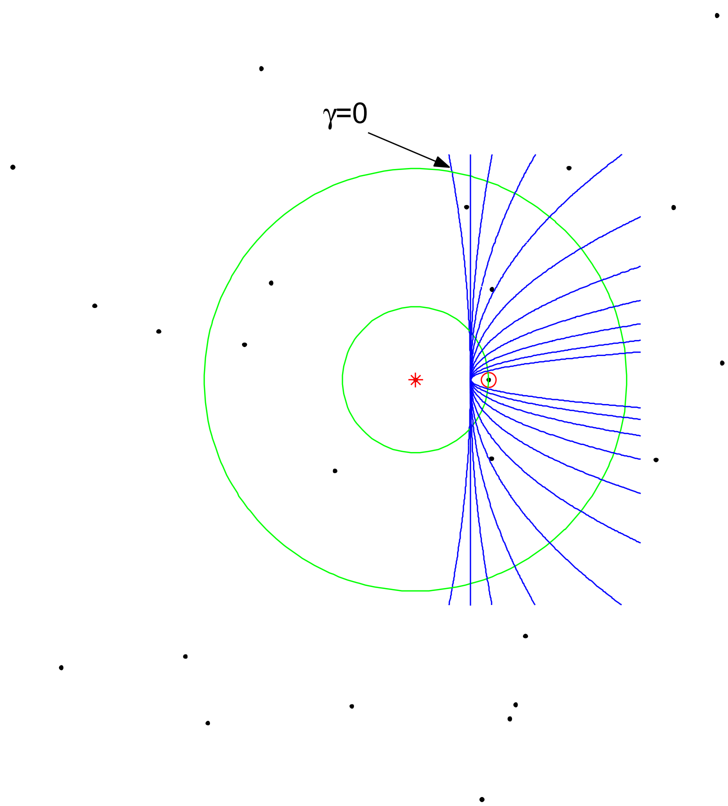


Figure 3.2: Definition of parabola when $\alpha > 0.5$. This picture shows $x = (1-\gamma)a_m y^2 + b$ family of parabolas with $\alpha = 0.75$ and $\gamma = 0, 1, 2, 4, ..., 512$. At the beginning the value of a is negative, hence parabola is inverted. As the value of γ increases the parabola closes up, becomes a line, and then gets narrower.

Lemma 3.1.2. *Consider a node u and a neighbor z of u, where WLOG we assume u is at the origin and z is positioned on the positive x-axis. Consider an arbitrary point $b = (1-\alpha)u + \alpha z$ at the distance $\alpha|uz|$ from u. Define L to be the line that is perpendicular to the line segment uz and that intersects uz at its midpoint m (corresponding to $\alpha = 0.5$). Define a parabola $x = ay^2 + b$ with its apex at b and its axis of symmetry on the line segment uz, oriented such that z is in its interior. Consider the parabola intersecting the line L at a point c (see Figure 3.4). Then the width of parabola is defined by;*

$$a = \frac{2(1-2\alpha)|uz|}{4|uc|^2 - |uz|^2}$$

Proof. Consider the right triangle $\triangle ucm$, we have $|mc|^2 = |uc|^2 - |um|^2$. By substituting c in the parabola's formula we conclude that $a = \dfrac{2(1-2\alpha)|uz|}{4|uc|^2 - |uz|^2}$.

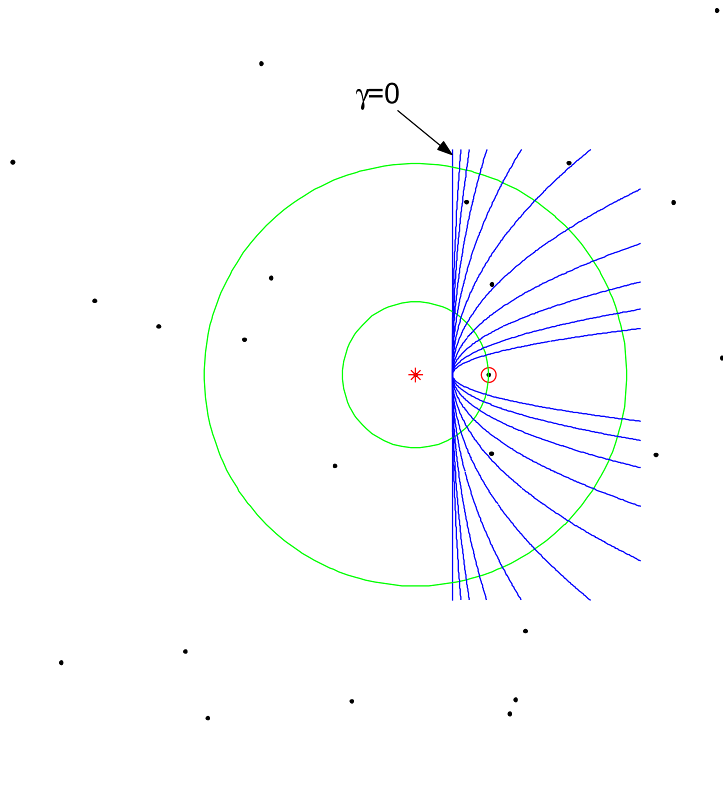


Figure 3.3: Definition of parabola when $\alpha = 0.5$. This picture shows $x = \gamma a_m y^2 + b$ family of parabolas with $\alpha = 0.5$ and $\gamma = 0, 1, 2, 4, ..., 512$. At the beginning the value of a is 0 and parabola starts as a line, then as the value of γ increases parabola gets narrower.

□

Corollary 3.1.3. *Consider a node u and a neighbor z of u, where WLOG we assume u is at the origin and z is positioned on the positive x-axis. Consider an arbitrary point $b = (1-\alpha)u + \alpha z$ at the distance $\alpha|uz|$ from u. Define L to be the line that is perpendicular to the line segment uz and that intersects uz at its midpoint m (corresponding to alpha $= 0.5$). Define a parabola $x = ay^2 + b$ with its apex at b and its axis of symmetry on the line segment uz, oriented such that z is in its interior. Consider the parabola intersecting the line L at a point c (see Figure 3.4). If $|uc| = |uz|$ then*

$$a = \frac{2(1-2\alpha)}{3|uz|}$$

Definition 3.1.4. *Consider a node u and a neighbor z of u, where WLOG we assume u is at the origin and z is positioned on the positive x-axis. Consider an arbitrary point $b = (1-\alpha)u + \alpha z$ at the distance $\alpha|uz|$ from u. Define L to be the line that* *is perpendicular to the line segment uz and that intersects uz at its midpoint m (corresponding to $\alpha = 0.5$). Define a parabola $x = ay^2 + b$ with its apex at b and its axis of symmetry on the line segment uz, oriented such that z is in its interior. Consider the parabola intersecting the line L at a point c (see Figure 3.4). Suppose maximum transmission range is r, define minimum value of a which is the maximum parabola width $a_m(\alpha, |uz|)$ as follows:*

$$a_m(\alpha, |uz|) = \begin{cases} \dfrac{(\frac{1}{2} - \alpha)}{\frac{3|uz|}{4}} & \textit{if } 0 \leq \alpha < 0.5 \\ \dfrac{(\delta - \frac{1}{2})}{\left((\frac{r}{|uz|})^2 - \frac{1}{4}\right)|uz|} & \textit{if } \alpha = 0.5 \\ \dfrac{(\frac{1}{2} - \alpha)}{\left((\frac{r}{|uz|})^2 - \frac{1}{4}\right)|uz|} & \textit{if } 0.5 < \alpha \leq 1 \end{cases} \tag{3.2}$$

where $\delta \in \,]0.5, 1[$ such that $\delta - \alpha < \epsilon$, ϵ small positive number.

Note that when $0.5 < \alpha \leq 1$ the width of the parabola a is negative. This definition describes a class of parabolas that not only move along the line segment uz but also are concave, straight or convex depending on the value of γ. See Figures 3.1, 3.2 and 3.3. When $\alpha = 0.5$ such that $a = 0$, we obtain the Half Space Proximal graph. Since for $\alpha \geq 0.5$ the limiting shape of the parabola is concave, if a fixed value of a is used for all nearest neighbors (regardless of the distance $|uz|$) such that the parabola is convex, the subgraph will be connected if the UDG is connected. This would be analogous to using a cone with a fixed angle in the $DAAY$ subgraph. It may be noted that for $\alpha < 0.5$, from Eq. 3.2, the width of the convex limiting

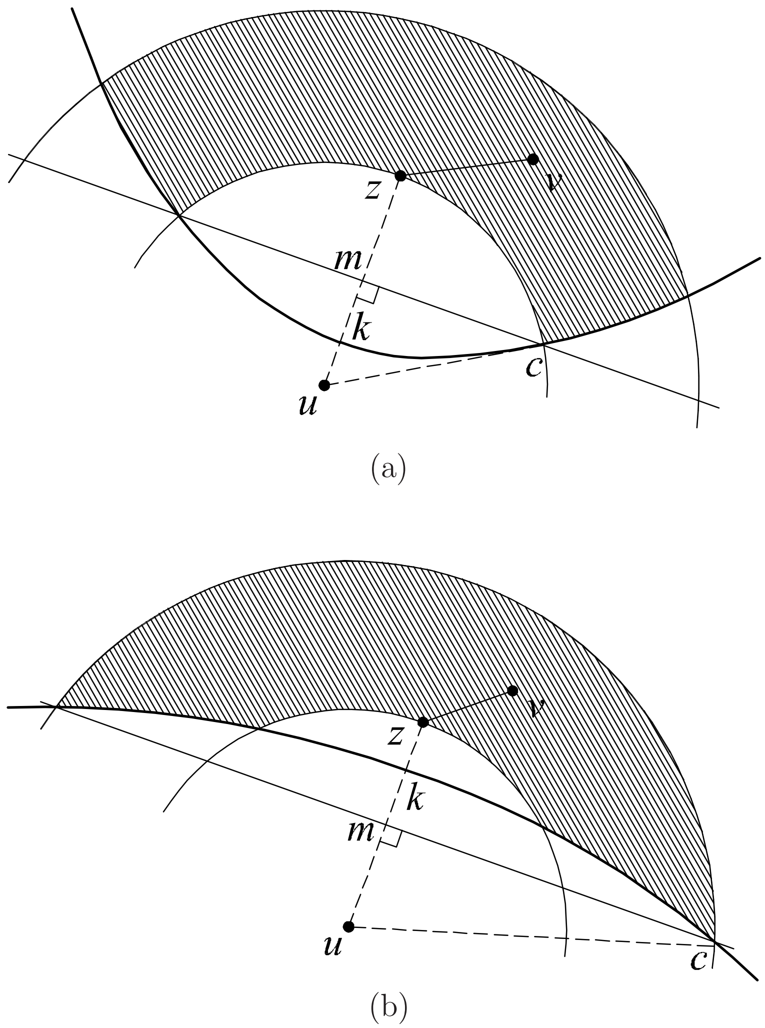


Figure 3.4: Figure for Thm.3.2.1. (a) $0 \leq \alpha < 0.5$ (b) $0.5 < \alpha \leq 1$

parabola depends on the distance between the node and its nearest neighbor and a fixed width a cannot be used for arbitrary UDGs.

3.2 Properties

In this section we will list several properties of the APY subgraph. In the following proofs we use an approach and notation similar to the proofs in [12].

3.2.1 Connectivity

Theorem 3.2.1. *Consider a node set V and $UDG(V)$ defined on V. Let $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$ to be the graph with node set V whose edges are obtained by applying the Adaptive Parabolic Yao(G, α, γ) algorithm, Algorithm 1 given on page 18, on the graph G using parabola width a (Eq. 3.1), and apex position b. If $UDG(V)$ is connected and the parabola width a is greater than or equal to $a_m(\alpha, |uz|)$ defined in Eq. 3.2 then $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$ is strongly connected.*

Proof. Consider a proof by contradiction. Assume that there is at least one edge $uv \in UDG(V)$ such that there is no directed path from u to v in $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$. Let uv be the shortest such edge in $UDG(V)$. This implies that there is an edge $uz \in \overrightarrow{APY}(UDG(V), \alpha, \gamma)$ such that $|uz| < |uv|$, because the edge uv should be in the parabola of uz selected by the Adaptive Parabolic Yao$(UDG(V), \alpha, \gamma)$ algorithm.

Now consider the triangle $\triangle uzv$ in Fig. 3.4. The choice of maximum parabola width in Def. 3.1.4 is based on the idea that it ensures that v is contained in the open half-plane H containing z defined by the line perpendicular to the line segment uz in the middle of uz (the point corresponding to $\alpha = 0.5$ in our parametrization of the uz; labeled as m in Fig. 3.4). Consider the three cases defined by the value of α. First, assume $0 \leq \alpha < 0.5$ (for example, the apex of the parabola would be at the point labeled as k in Fig. 3.4(a)). Then to keep v in the interior of H, the maximum parabola width for a would define a parabola that intersects the boundary of H at a point at distance $|uz|$ from u (such a point is labeled as c in the figure). By Corollary 3.1.3, this limiting parabola width $a_m(\alpha, |uz|)$ as defined in Def. 3.1.4.

Now, assume $0.5 < \alpha \leq 1$. To keep v in the interior of H, the maximum parabola width for a would define a parabola that intersects the boundary of H at a point at distance r from u (such a point is labeled as c in the Fig. 3.4(b)). By Lemma 3.1.2 and noting that $|uc| = r$, this limiting parabola width $a_m(\alpha, |uz|)$ is defined in Def. 3.1.4. The case of $\alpha = 0.5$ implies the HSP graph of $UDG(V)$ and as shown in [10] it is connected.

In all three cases, since $a \geq a_m(\alpha, |uz|)$, then any position of the node v inside the parabola for z such that $|uz| \leq |uv|$ would give $|zv|$ strictly less than $|uv|$. Since uv is an edge in $UDG(V)$, then zv is also an edge in $UDG(V)$. Therefore, there exists a directed path from z to v in $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$, and so there is a directed path from u to v in $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$.

□

3.2.2 Bounded out degree

Theorem 3.2.2. *Consider a node set V and $UDG(V)$ defined on V. Let $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$ to be the graph with node set V whose edges are obtained by applying the Adaptive Parabolic Yao(G, α, γ) algorithm, Algorithm 1 given on page 18, on the graph G using parabola width a(Eq. 3.1), and apex position b. The out-degree of any node in $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$, $0 \leq \alpha \leq 1$, is $\frac{2\pi}{\phi}$ where ϕ is defined by:*

$$\begin{cases} \left(a(r\sin\phi)^2 + \alpha d_0\right)^2 + (r\sin\phi)^2 = r^2 & \text{if } 0 \leq \alpha < 0.5 \\ \left(a(d_0\sin\phi)^2 + \alpha d_0\right)^2 + (d_0\sin\phi)^2 = d_0^2 & \text{if } 0.5 \leq \alpha \leq 1 \end{cases} \tag{3.3}$$

where $d_0 = \min\{dist(u, v) : u, v \in UDG(V)\}$.

Proof. By the definition of Adaptive Parabolic Yao(G, α, γ), the smallest angle between any two edges is when any nearest neighbor that is selected to form an edge is outside, or on the boundary of, the parabola for any other neighbor.

Consider a node u. Let z be the nearest neighbor. The parameters α and γ define a parabola. For $0 \leq \alpha < 0.5$, the smallest angle between z and another nearest

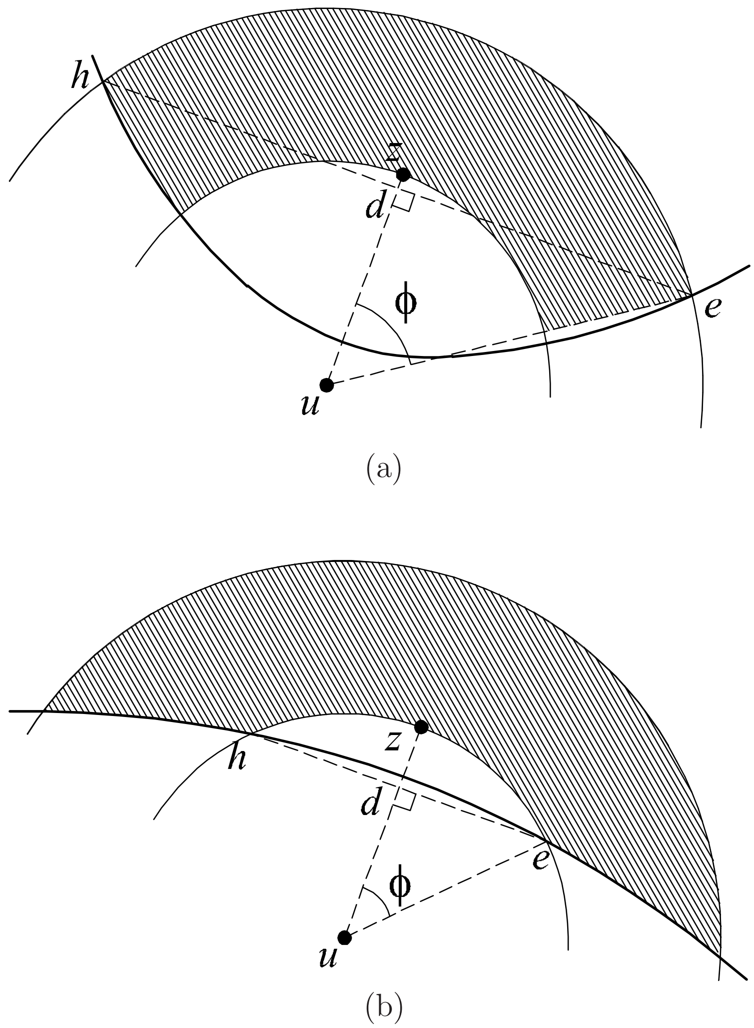


Figure 3.5: Figure for Thm.3.2.2. (a) $0 \leq \alpha < 0.5$ (b) $0.5 \leq \alpha \leq 1$

neighbor w, occurs when w is placed at the intersection e of the parabola boundary and the circle of radius r, transmission range, centered on u (Fig. 3.5(a)). Consider the segment eh intersecting uz at d. We have $|ed| = r\sin\phi$ and from parabola $|ud| = a(r\sin\phi)^2 + \alpha|uz|$. Therefore we have:

$$\big(a(r\sin\phi)^2 + \alpha|uz|\big)^2 + (r\sin\phi)^2 = r^2$$

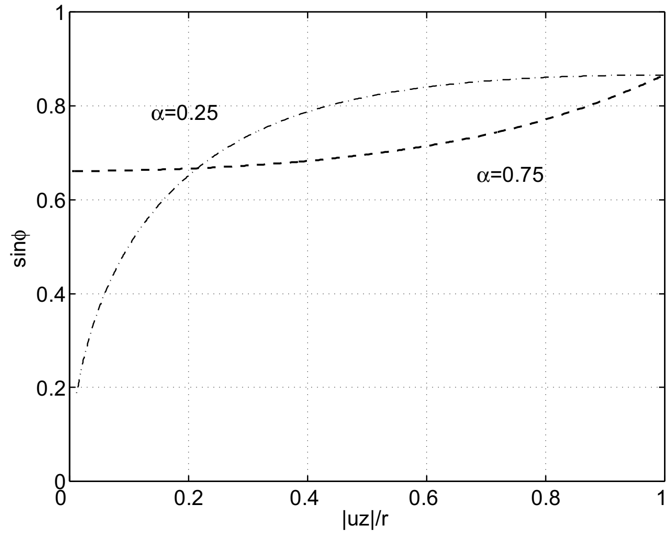


Figure 3.6: An example Figure for Thm.3.2.2. For both cases $0 \leq \alpha < 0.5$ and $0.5 \leq \alpha \leq 1$ as $|uz|$ increases, the smallest angle ϕ also increases

When $0.5 \leq \alpha \leq 1$, the smallest angle between z and another nearest neighbor w, occurs if w is placed at the intersection e of the parabola boundary and the circle of radius uz centered on u (Fig. 3.5(b)). Consider the segment eh intersecting uz at d. We have $|ed| = |uz| \sin \phi$ and from parabola $|ud| = a(|uz| \sin \phi)^2 + \alpha |uz|$. Therefore we have:

$$\left(a(|uz| \sin \phi)^2 + \alpha |uz|\right)^2 + (|uz| \sin \phi)^2 = |uz|^2$$

As we can see in Fig. 3.6, the smallest angle depends on distance of u and its closest neighbor z that constructs the parabola. As $|uz|$ increases ϕ increases for both $0 \leq \alpha < 0.5$ and $0.5 < \alpha \leq 1$ cases. Therefore smallest ϕ is when nodes u and z are the closest nodes in the network. Call this smallest distance d_0. Then the smallest angle between any two nodes is defined by Eq 3.3 and the maximum out-degree for any node will be $\dfrac{2\pi}{\phi}$.

□

The Adaptive Parabolic Yao graph has a conditional maximum out-degree for each node, since for $0 \le \alpha < 0.5$ it depends on d_0 which is the distance between the closest nodes in the network. Note that ϕ is a fixed angle for $0.5 < \alpha \le 1$ and a fixed γ, and decreasing $|uz|$ does not change the size of the angle ϕ (see Figure 3.5-(b) and also Figure 3.6 for the case $\alpha = 0.75$). This shows that high out-degree happens for the nodes with closest neighbors. As the value of γ increases from 0 the parabola is narrower and the maximum out-degree of the APY subgraph would be that of the original UDG. As an example, based on the Thm. 3.2.2, when $\alpha = 0.25$, for a fixed $|uz|$, $\gamma = 0$ gives $\phi = 49.73\,^\circ$ and therefore the out-degree of u is 7.2, while $\gamma = 8$ gives $\phi = 18.40\,^\circ$ and the out-degree is 19.6. When $\alpha = 0.75$, for the same fixed $|uz|$, $\gamma = 0$ gives $\phi = 42.63\,^\circ$ and therefore the out-degree of u is 8.4, while $\gamma = 8$ gives $\phi = 34.84\,^\circ$ and the out-degree is 10.3. Conversely, the in-degree is not bounded by a constant and could be as large as the number of nodes.

3.2.3 Length stretch factor

Theorem 3.2.3. *Consider a node set V and $UDG(V)$ defined on V. Let $\overrightarrow{APY}(UDG$-$(V), \alpha, \gamma)$ to be the graph with node set V whose edges are obtained by applying the Adaptive Parabolic Yao(G, α, γ) algorithm, Algorithm 1 given on page 18, on the graph G using parabola width a (Eq. 3.1), and apex position b. Then $APY(UDG(V), \alpha, \gamma)$ is a spanner with length stretch factor* $\dfrac{1}{1 - 2\sin\frac{\phi}{2}}$.

Proof. Let uv be an edge in UDG that is not selected by Adaptive Parabolic Yao(G, α, γ) algorithm. Since, by Theorem 3.2.1, $APY(UDG(V), \alpha, \gamma)$ is connected, then there is a shortest path from u to v. Let a "worst" such path from u to v in $APY(UDG(V), \alpha, \gamma)$ be $u_0 = u, u_1, u_2, \ldots, u_m = v$. See Fig. 3.7. By the Adaptive Parabolic Yao(G, α, γ) algorithm, the angle $\angle u_{i+1}u_iv \le \phi$ from Eq. 3.3 on page 25 and $|u_iu_{i+1}| < |u_iv|$ since

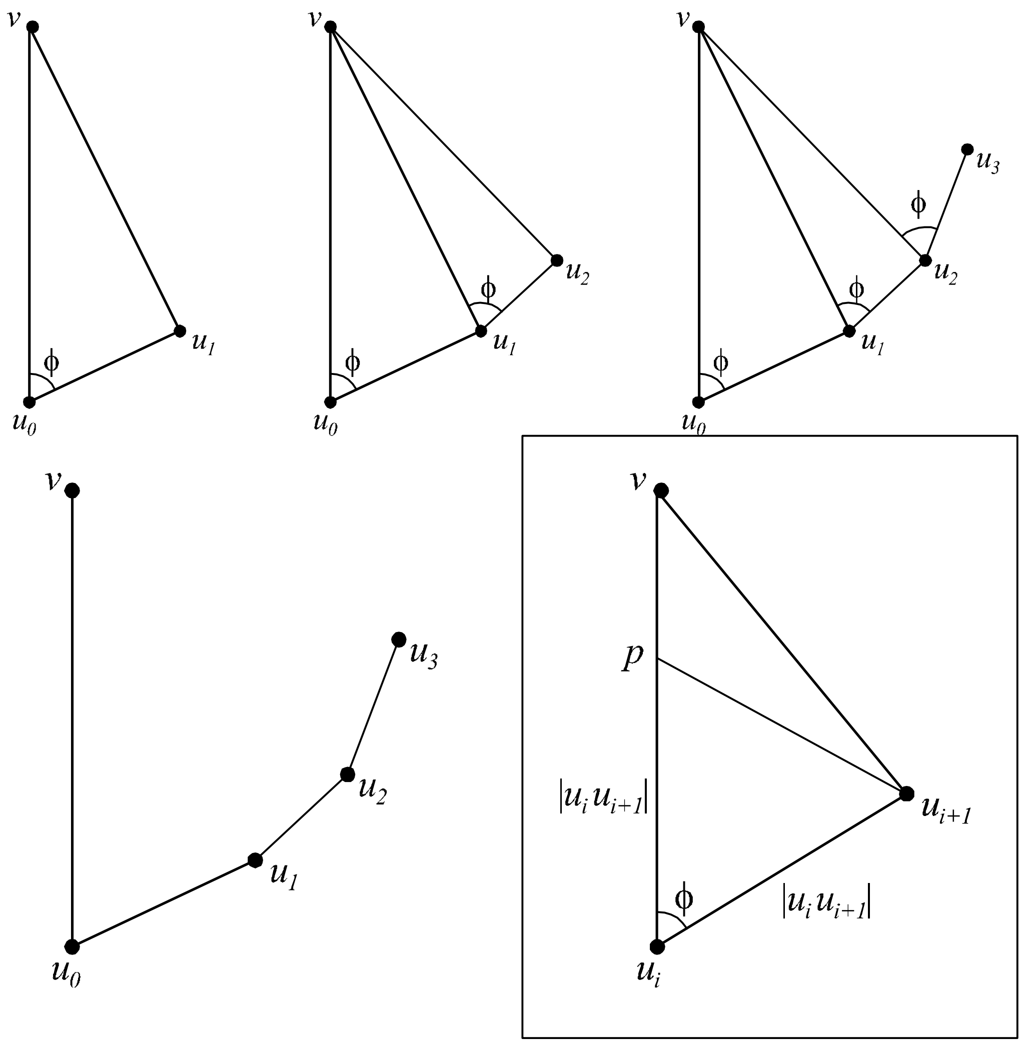


Figure 3.7: Figure for Thm. 3.2.3

otherwise $u_i v$ would be part of $APY(UDG(V), \alpha, \gamma)$ and part of the path. Also, by Theorem 3.2.1, $|u_{i+1}v| < |u_i v|$ since we can always decrease the distance to v from each u_i along the path.

Now consider the triangle $\triangle u_i u_{i+1} v$. See Fig. 3.7. Let p be the point on $u_i v$ such that $|u_i p| = |u_i u_{i+1}|$. By the triangular inequality $|u_{i+1}v| \leq |u_{i+1}p| + |pv|$. Note that $|u_{i+1}p| = (2 \sin \frac{\phi}{2})|u_i u_{i+1}|$, and $|pv| = |u_i v| - |u_i u_{i+1}|$. Applying these two latter equations to the triangular inequality, we obtain

$$|u_{i+1}v| \leq |u_iv| - |u_iu_{i+1}|(1 - 2\sin\frac{\phi}{2}).$$

Applying the previous analysis iteratively on the entire path, we have

$$\sum_{0\leq i<m} |u_{i+1}v| \leq \sum_{0\leq i<m} \left(|u_iv| - \left(|u_iu_{i+1}|(1 - 2\sin\frac{\phi}{2})\right)\right).$$

Therefore,

$$\begin{aligned}\sum_{0\leq i<m} (|u_iu_{i+1}|) &\leq \left(\frac{1}{1 - 2\sin\frac{\phi}{2}}\right) \sum_{0\leq i<m} (|u_iv| - |u_{i+1}v|) \\ &\leq \left(\frac{1}{1 - 2\sin\frac{\phi}{2}}\right) |u_0v|.\end{aligned}$$

The larger the angle ϕ, the larger the stretch factor along the path. Consider when u_{i+1} is a nearest neighbor of u_i defining a parabola by the Adaptive Parabolic Yao (G, α, γ) algorithm. For $0 \leq \alpha < 0.5$, the largest angle possible between u_i, v, and u_{i+1} is when a node is placed at c, the intersection of the parabola boundary and the boundary of the circle of radius u_iu_{i+1} centered at u_i (Fig. 3.4(a), page 23). In this case the largest angle is $\frac{\pi}{3}$. For $0.5 < \alpha \leq 1$, the largest angle possible between u_i, v, and u_{i+1} is when a node is placed at c, the intersection of the parabola boundary and the boundary of the circle of radius r centered at u_i (Fig. 3.4(b), page 23). In this case as $|u_iu_{i+1}| \to 0$, the angle ϕ is maximized to $\frac{\pi}{2}$.

For the stretch factor to be bounded by the above inequality, then ϕ must be restricted by $\phi < \dfrac{\pi}{3}$ thus for any i, $d_0 \leq |u_i u_{i+1}|$ and from Eq. 3.3:

$$\Big(a(d_0 \sin\frac{\pi}{3})^2 + \alpha|u_i u_{i+1}|\Big)^2 + (d_0 \sin\frac{\pi}{3})^2 < d_0^2$$

Therefore for any node u the width of parabola constructed by its nearest neighbor z, should be restricted by $a < \dfrac{(\frac{1}{2} - \alpha)}{\frac{3d_0}{4}}$.

□

3.2.4 Power stretch factor

Theorem 3.2.4. *Consider a node set V and $UDG(V)$ defined on V. Let $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$ to be the graph with node set V whose edges are obtained by applying the Adaptive Parabolic Yao(G, α, γ) algorithm, Algorithm 1 given on page 18, on the graph G using parabola width a (Eq. 3.1), and apex position b. Then $APY(UDG(V), \alpha, \gamma)$ is a spanner with power stretch factor $\dfrac{1}{1-(2\sin\frac{\phi}{2})^\beta}$. Where ϕ is defined as in Thm. 3.2.2 and $\phi < \pi/3$.*

Proof. Here we use a similar approach to the one presented in [24]. For a constant δ, $\rho^p_{APY} \leq \delta$ if and only if for any link $uv \in UDG\backslash APY$, $p_{APY}(u,v) \leq \delta|uv|^\beta$ (see [24]), where $p_{APY}(u,v) = \Sigma_{i=1}^m |u_{i-1}u_i|^\beta$ is the total transmission power by path $u_0 = u, u_1, u_2, \ldots, u_m = v$ from u to v. Let δ be $1/(1-(2\sin\frac{\phi}{2})^\beta)$, we show that for any pair of nodes $u, v \in UDG\backslash APY$ there is a path connecting u and v in APY with $p^p_{APY}(u,v) \leq \delta|uv|^\beta$. Since APY is connected, if edge $(u,v) \notin APY$ then there is a path $u_0 = u, u_1, u_2, \ldots, u_m = v$ that connect u and v in APY.

We use induction. For $m = 1$ it is clear that $|uv|^\beta \leq \delta|uv|^\beta$. Now suppose $p_{APY}(u,v) \leq \delta|uv|^\beta$ for any path from u to v with $(m-1)$ edges. Consider a path with m edges, containing edge uw and path from w to v with $(m-1)$ edges. We consider two cases. Case 1: If angle $\angle uwv < \dfrac{\pi}{2}$ (see Fig. 3.8(a)) we know that $|uw| \leq |uv|$ and $\angle wuv \leq \phi$. Consider point p along edge uw such that $|up| = |uv|$. It is clear that $|vw| \leq |vp|$ and $|vp| = (2\sin\frac{\phi}{2})|uv|$. Therefore $|wv| \leq (2\sin\frac{\phi}{2})|uv|$, then:

$$p_{APY}(u,v) = |uw|^{\beta} + p_{APY}(w,v) \leq |uw|^{\beta} + \delta|wv|^{\beta}$$

$$\leq |uv|^{\beta} + \delta(2\sin\frac{\phi}{2})^{\beta}|uv|^{\beta} = \delta|uv|^{\beta}$$

Case 2: If angle $\angle uwv \geq \pi/2$ we have $|uw|^2+|wv|^2 \leq |uv|^2$ (see Fig. 3.8(b)), therefore $\left(\frac{|uw|}{|uv|}\right)^{\beta} + \left(\frac{|wv|}{|uv|}\right)^{\beta} \leq \left(\frac{|uw|}{|uv|}\right)^{2} + \left(\frac{|wv|}{|uv|}\right)^{2} \leq 1$ since $\beta \geq 2$, which implies that $|uw|^{\beta} + |wv|^{\beta} \leq |uv|^{\beta}$. Now we have:

$$p_{APY}(u,v) = |uw|^{\beta} + p_{APY}(w,v) \leq |uw|^{\beta} + \delta|wv|^{\beta}$$

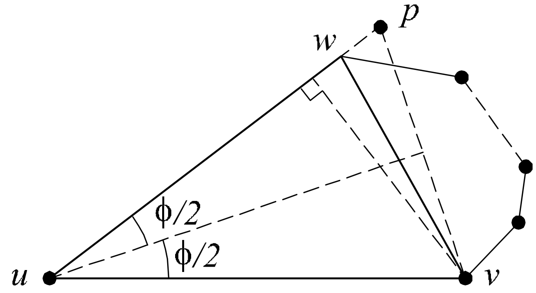
$$\leq \delta|uw|^{\beta} + \delta|wv|^{\beta} \leq \delta|uv|^{\beta}$$

This proves that $\rho^{p}_{APY} \leq \delta$.

□

By using the same logic presented in proof of Thm. 3.2.4, we can also prove that power stretch factor of Displaced Apex Adaptive Yao graph is $\dfrac{1}{1-(2\sin\frac{\theta}{2})^{\beta}}$.

Theorem 3.2.5. *Let $V \subseteq \mathbb{R}^2$ be a set of N points and let $\theta < \pi/3$ be the cone angle. Then $DAAY(UDG(V), \alpha, s)$ is a spanner with power stretch factor $\dfrac{1}{1-(2\sin\frac{\theta}{2})^{\beta}}$.*



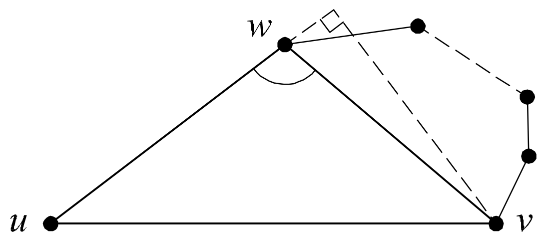


Figure 3.8: Figure for Thm. 3.2.4 (a) Case 1, $\angle uwv < \pi/2$; (b) Case 2 $\angle uwv > \pi/2$.

3.2.5 Containing EMST

Theorem 3.2.6. *Consider a node set V and $UDG(V)$ defined on V. Let $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$ to be the graph with node set V whose edges are obtained by applying the Adaptive Parabolic Yao(G, α, γ) algorithm, Algorithm 1 given on page 18, on the graph G using parabola width a(Eq. 3.1), and apex position b. Assume that $UDG(V)$ is connected. Then $APY(UDG(V), \alpha, \gamma)$, $0 \leq \alpha \leq 1$, contains the Euclidean Minimum Spanning Tree $EMST(UDG(V))$ as a subgraph.*

Proof. This proof has a similar approach as in proof for the $HSP(UDG(V))$ (see [10]). Let $EMST(UDG(V))$ be a Euclidean Minimum Spanning Tree of $UDG(V)$ that contains the maximum number of edges of $APY(UDG(V), \alpha, \gamma)$. Consider a proof by contradiction. Assume there is an edge uv in $EMST(UDG(V))$ that is not in $APY(UDG(V), \alpha, \gamma)$. This implies that there is an edge $uz \in APY(UDG(V), \alpha, \gamma)$

such that $|uz| \leq |uv|$, because the edge uv should be in the cone of another shorter edge selected by the Adaptive Parabolic Yao $(UDG(V), \alpha, \gamma)$ algorithm, and $|vz| < |uv|$ (otherwise, by Theorem 3.2.1, the $\overrightarrow{APY}(UDG(V), \alpha, \gamma)$ would not be strongly connected). Since $EMST(UDG(V))$ is a spanning tree, there is a path from v (or u) to z. If the path is from v to z, then removing uv from the graph and adding the edge uz we obtain a spanning tree with equal or less weight with an additional edge from $APY(UDG(V), \alpha, \gamma)$, a contradiction. If the path is from u to z, then removing uv from the graph and adding the edge vz we obtain a spanning tree with less weight, again a contradiction.

□

Chapter 4

Simulation and results

In all our experiments we have chosen connected unit disk graphs by randomly generating node coordinates on an area of 100×100 units. The graphs that were not connected were not used. Also, the transmission range of 15 units has been used. We varied N, the number of nodes, between 65, 75, 85, 95 and 105 nodes. We have used similar parameters as those used in the simulations presented in similar published research. All results are averaged over 70 or 24 graphs, as will be indicated in each case, for each value of N. Note that for all our simulations we use $\epsilon = 10^{-1}$ and $\delta = \alpha + 10^{-1}$ in Eq. 3.2. To calculate power stretch factor, we have considered $\beta = 2$ in Eq. 2.1.

4.1 Adaptive parabolic Yao graph

The following are the implementation results of the Adaptive Parabolic Yao graph. Here we study the behavior of APY graph with respect to two parameters α and γ. We have considered different values of $\alpha = 0, 0.125, 0.25, \ldots, 0.875, 1$ and $\gamma = 0, 1, 2, 4, \ldots, 2^d$. The case of $\gamma = 0$ is when the parabola is defined with its maximum width, a_m. All the values are averaged over 70 random graphs. As we can see from contour lines, APY graph has a strong dependency on both α and γ. When $\gamma = 0$ and $0.5 < \alpha \leq 1$, APY is behaving more like HSP graph.

From Figures 4.1, 4.2, 4.3 , 4.4, 4.5, 4.6, the behavior of APY graph is different for values less than $\alpha = 0.5$ and for that of greater than $\alpha = 0.5$. For both cases as γ increases average node degree, maximum node degree, number of crossing edges and total weight of the graph increases while hop stretch factor and power stretch factor decreases. Minimum value, as a lower bound, of average node degree, maximum node degree, number of crossing edges and total weight of the graph is where $\alpha = 0.5$ and $\gamma = 0$. Although for all γ the values decrease when α goes from 0.5 to 1, it is in a lower rate than when α goes from 0.5 to 0 (Figures 4.1, 4.2, 4.3 , 4.4). From Fig. 4.3, number of crossing edges for graphs with $\alpha > 0.5$ and $\gamma < 3$ is zero and also for the same values of γ the graphs with $\alpha < 0.5$ have low number of crossing edges. These graphs are more suitable for routing algorithms that use planar graphs.

Power stretch factor and hop stretch factor are calculated by formulas 2.1 and 2.2 from chapter 2.1. Figures 4.5 and 4.6 show that a maximum value, as an upper bound, occurs where $\alpha = 0.5$ and $\gamma = 0$. For all γ the hop stretch factor is higher when $\alpha > 0.5$ than to the values when $\alpha < 0.5$. As γ increases graphs with $\alpha < 0.5$ have a very close value for hop stretch factor $(2.01 - 2.04)$.The power stretch factor also has a maximum, as an upper bound, at $\alpha = 0.5$ and $\gamma = 0$, but with the reversed situation. Values for graphs with $\alpha > 0.5$ are lower than those with $\alpha < 0.5$. For $\gamma = 0$, power stretch factors for $\alpha = 0$ and $\alpha = 1$ are very close (1.61 and 1.62).

Although we have mentioned in Section 3.2.3 that the maximum angle ϕ should be bounded from above by $\pi/3$, the experimental results show no drastic change in stretch factors when this value is exceeded. So, in practice it would appear that such a restriction on a is not necessary.

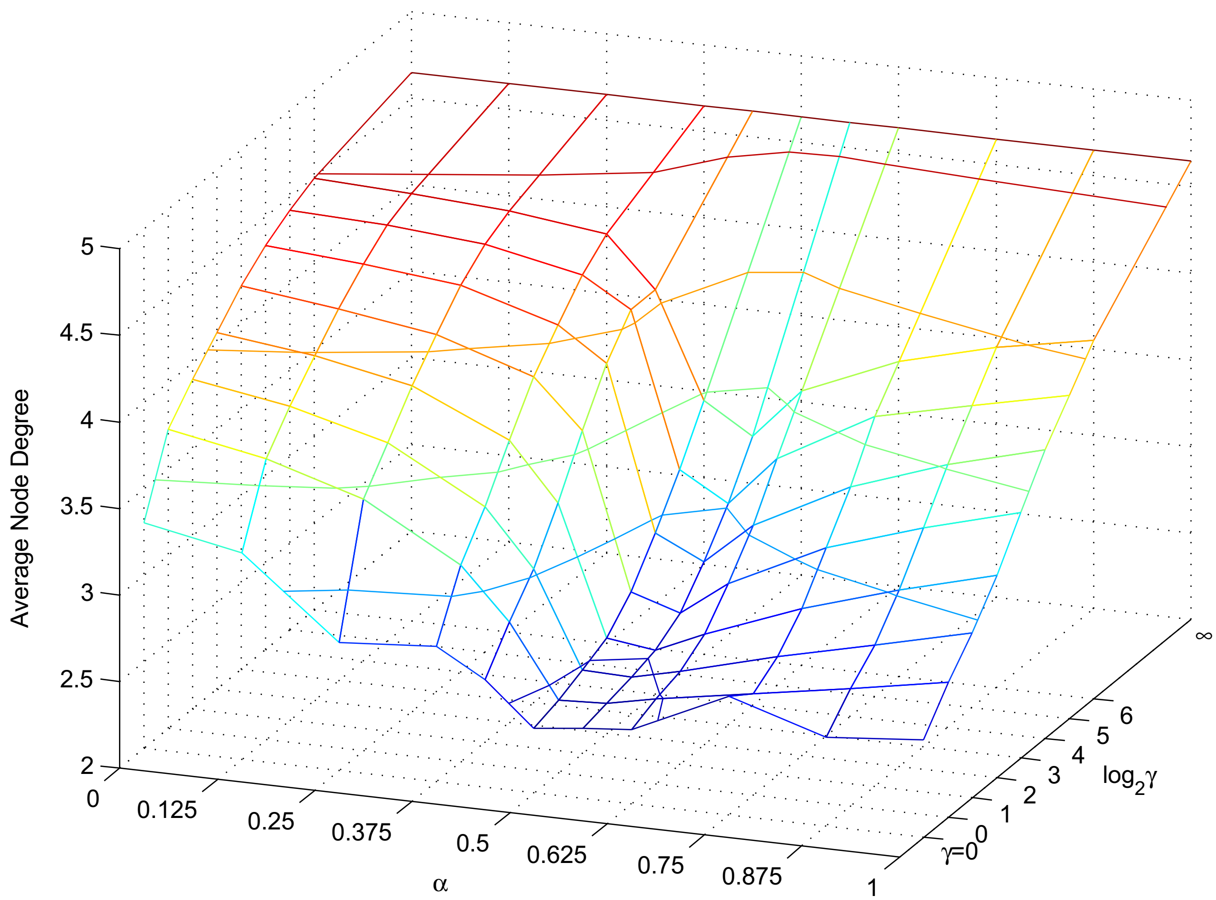


Figure 4.1: Average Node Degree for APY graph

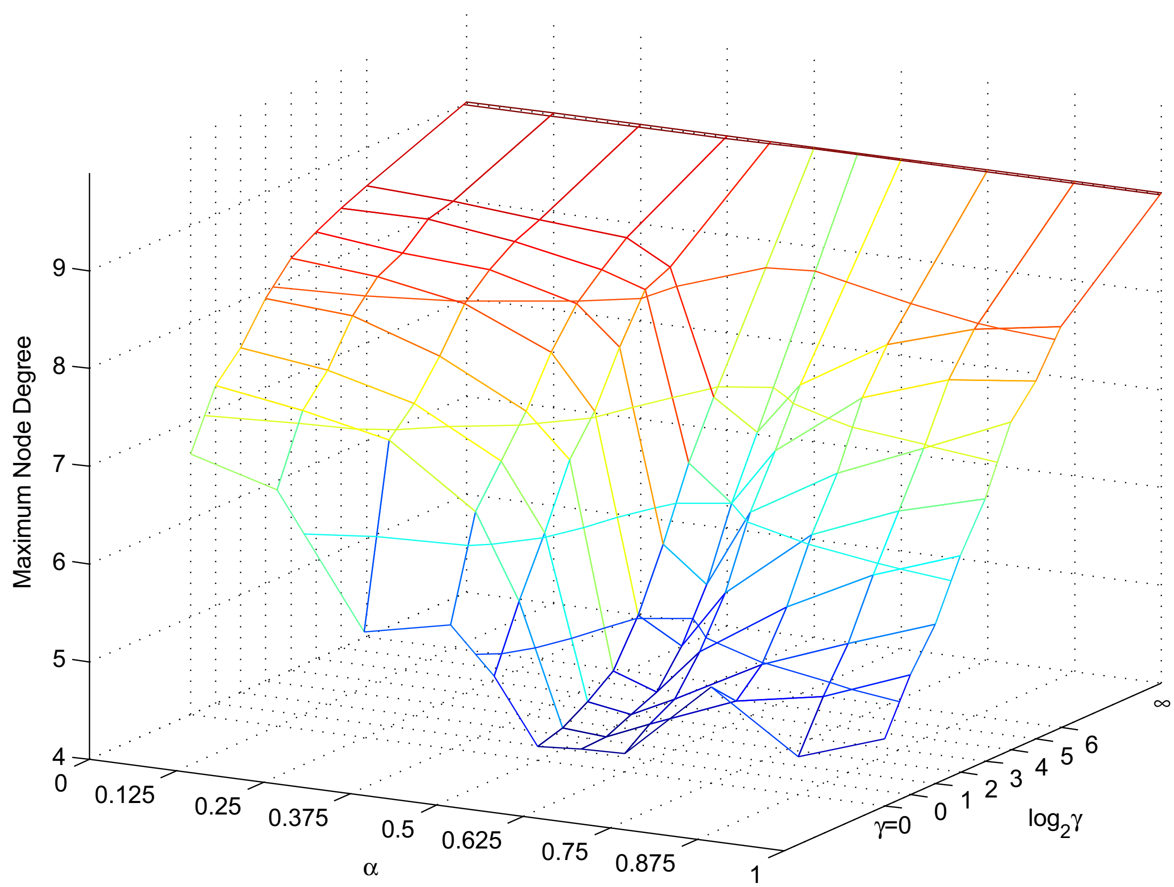


Figure 4.2: Maximum Node Degree for *APY* graph

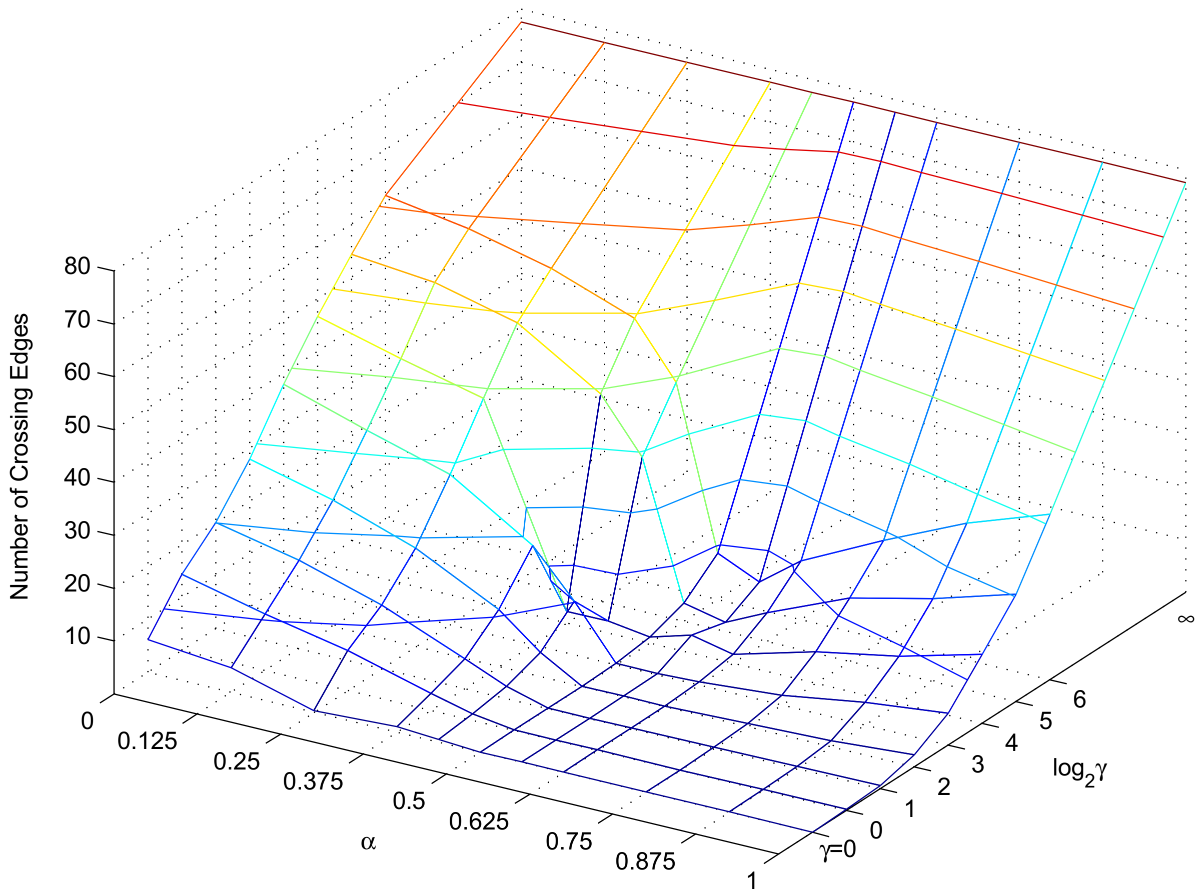


Figure 4.3: Number of Crossing Edges for *APY* graph

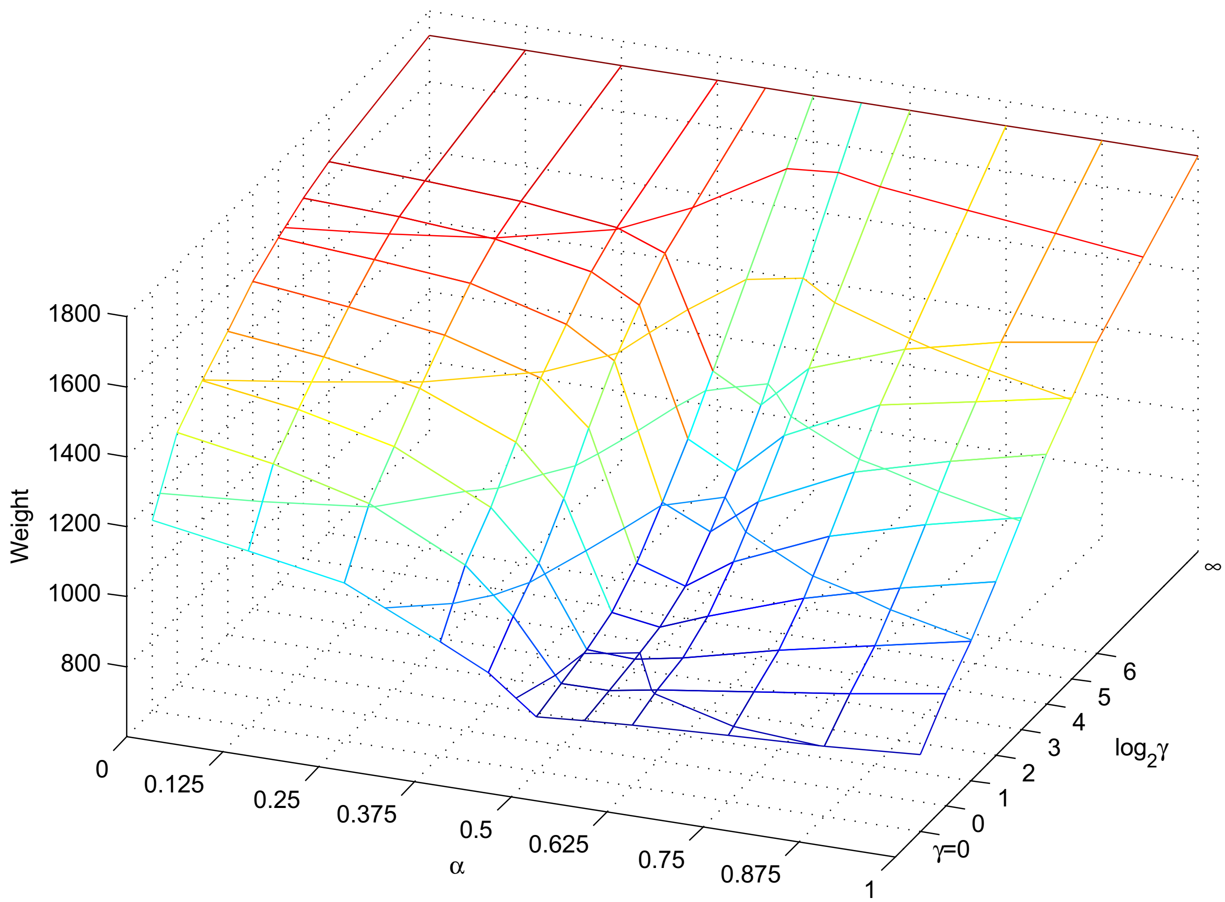


Figure 4.4: Total Euclidean Weight for *APY* graph

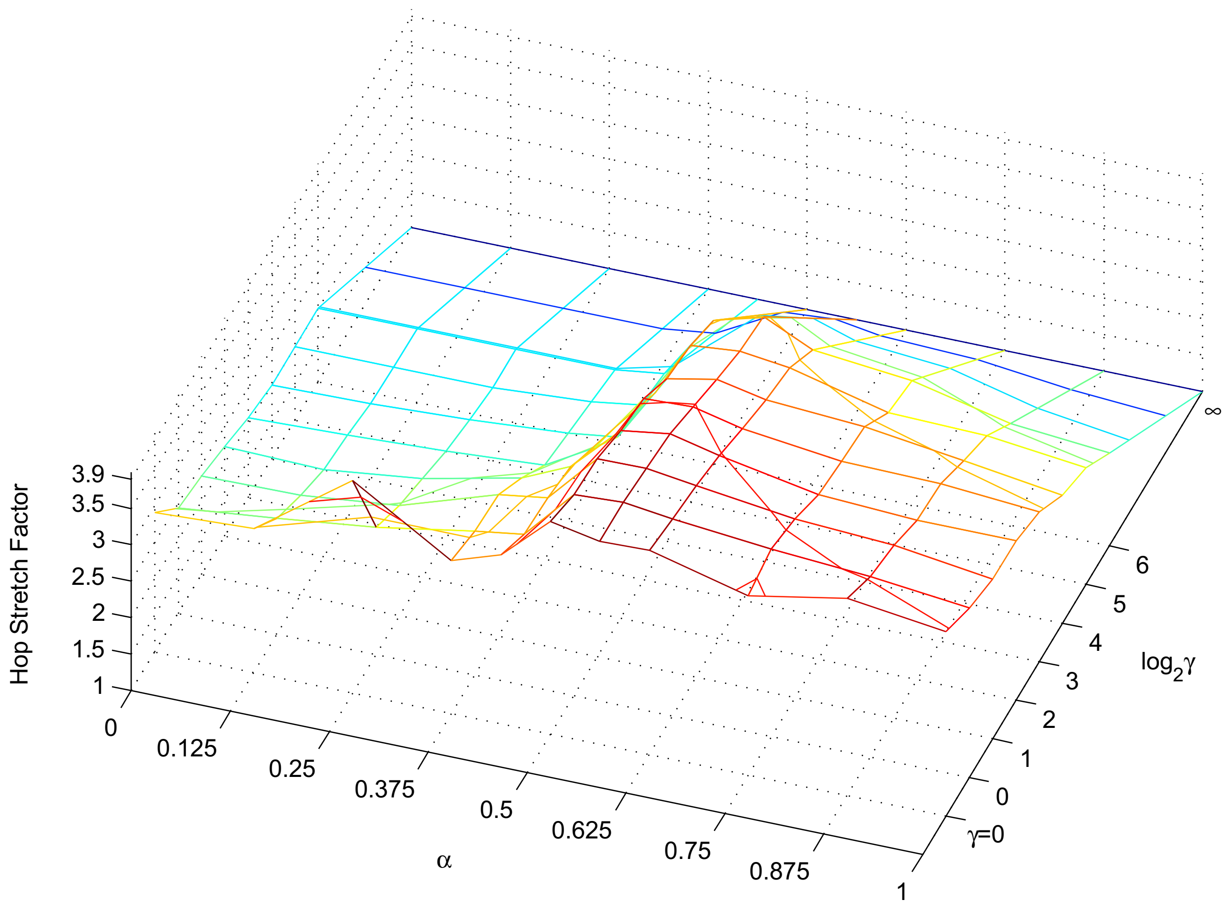


Figure 4.5: Hop stretch Factor for *APY* graph

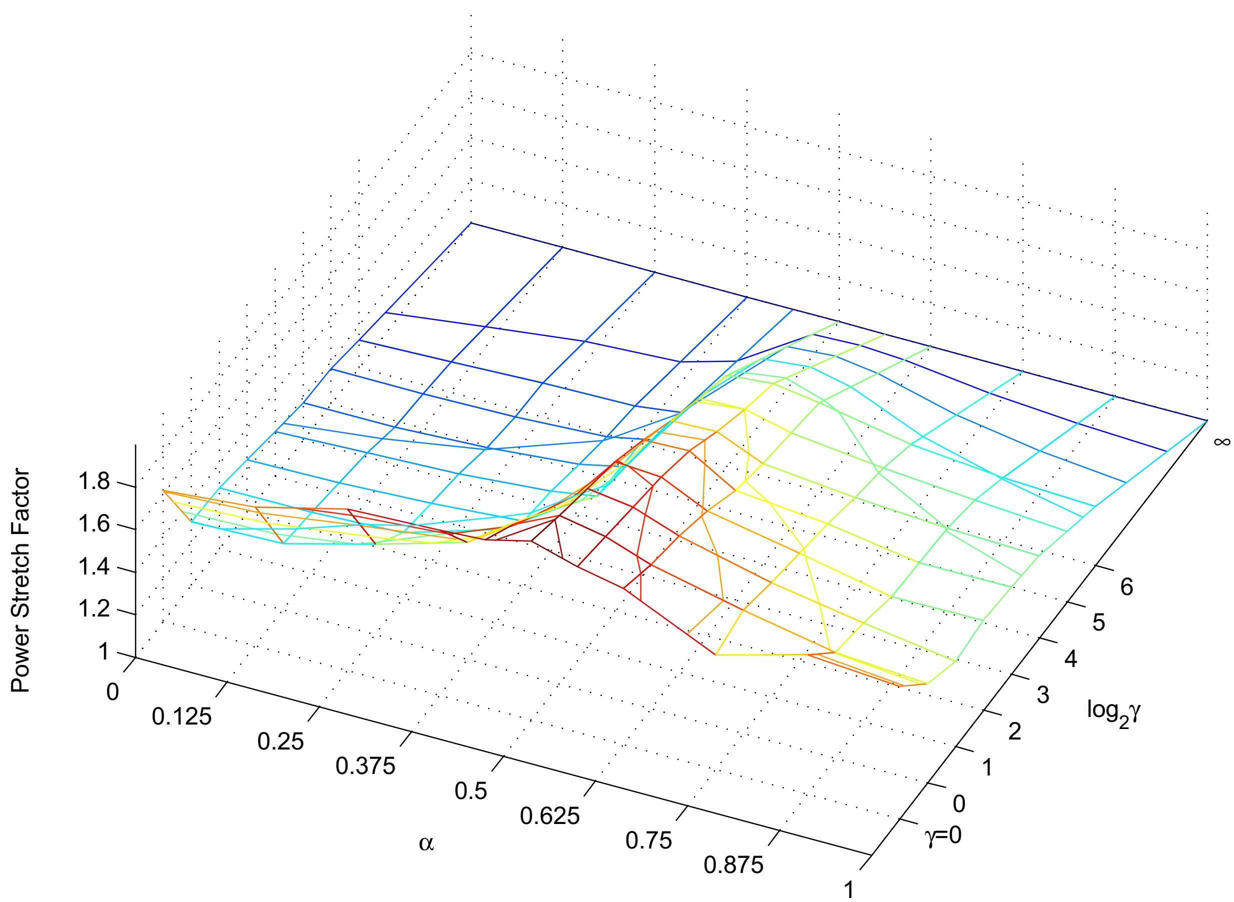


Figure 4.6: Power Stretch Factor for *APY* graph

4.2 Comparisons

In this section we compare Adaptive Parabolic Yao graph to the spanning subgraphs mentioned in chapter 2. All the values are averaged over 24 random graphs. Number of nodes, N, varies between 65, 75, 85, 95 and 105 nodes.

4.2.1 APY and $DAAY$ graphs

In the following we show a comparison of the Adaptive Parabolic Yao graph with $DAAY$ and HSP as a special case for both APY and $DAAY$ graphs.

From Figures 4.7, 4.8, 4.9 and 4.10 we see that APY graph has relatively higher node degree and number of crossing edges than $DAAY$ graph. Also, the total weight of the graph in APY is larger than in $DAAY$. But as the number of nodes in the network increases, the APY graph has the ability to greatly reduce the degree of nodes and number of crossing edges to a lower limit than the $DAAY$ graph, where those values change in a small range. This is made possible by varying the value of α.

On the other hand, hop stretch factor and power stretch factor of APY graph are clearly less than the values for $DAAY$ graph, for any number of nodes N. See Figures 4.11 and 4.12. Although large number of crossing edges could be a restriction for some routing algorithms, APY graph enables to model a large network with low value for both hop stretch factor and power stretch factor.

Figure 4.13 demonstrates a histogram for percentage of nodes with respect to degree of nodes. As we can see, for $\alpha = 0$ it is more probable that the node degree for nodes in APY graph to vary in a wider range of values while in $DAAY$ a higher percentage of nodes have low degrees. When α increases from 0 to 1, the values for APY graph monotonically increases towards the values for $\alpha = 0.5$ which is the HSP graph.

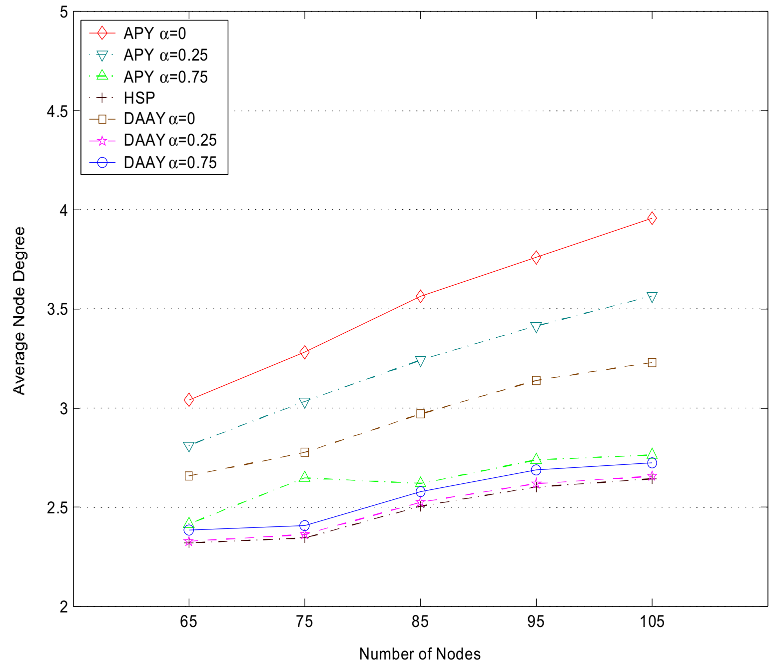


Figure 4.7: Average Node Degree for *APY* and *DAAY* graphs

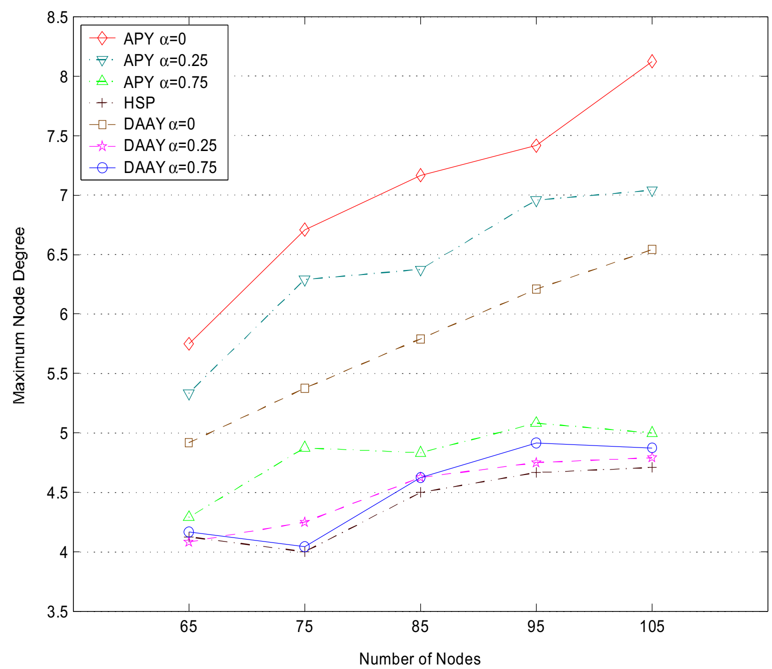


Figure 4.8: Maximum Node Degree for *APY* and *DAAY* graphs

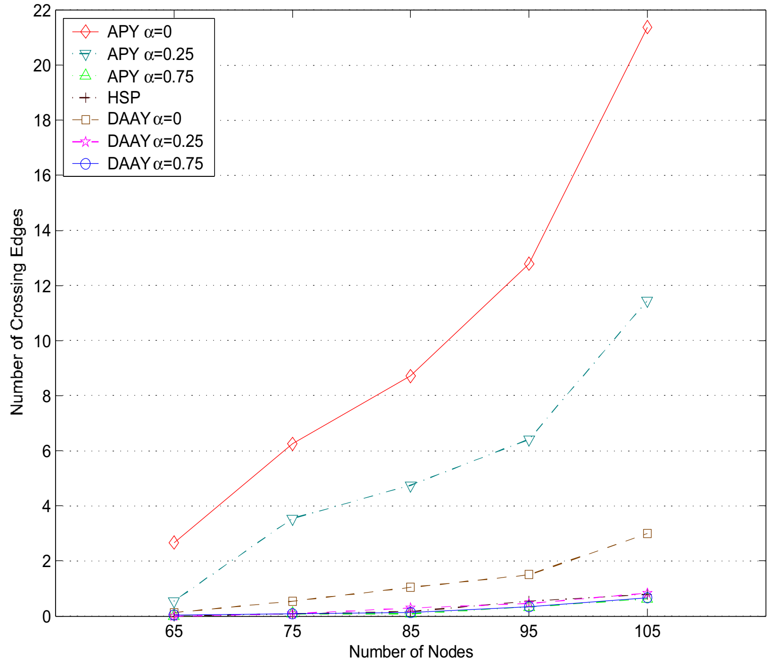


Figure 4.9: Number of Crossing Edges for *APY* and *DAAY* graphs

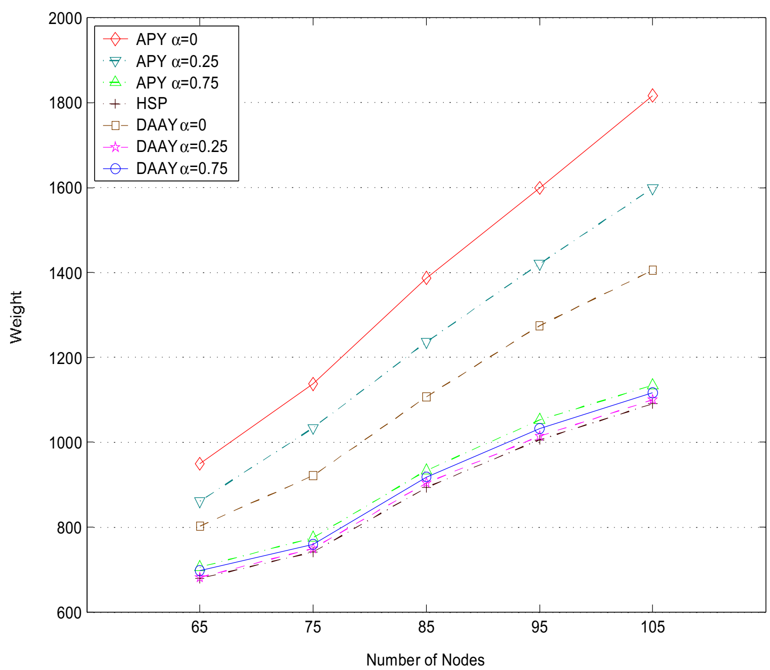


Figure 4.10: Total Euclidean Weight for *APY* and *DAAY* graphs

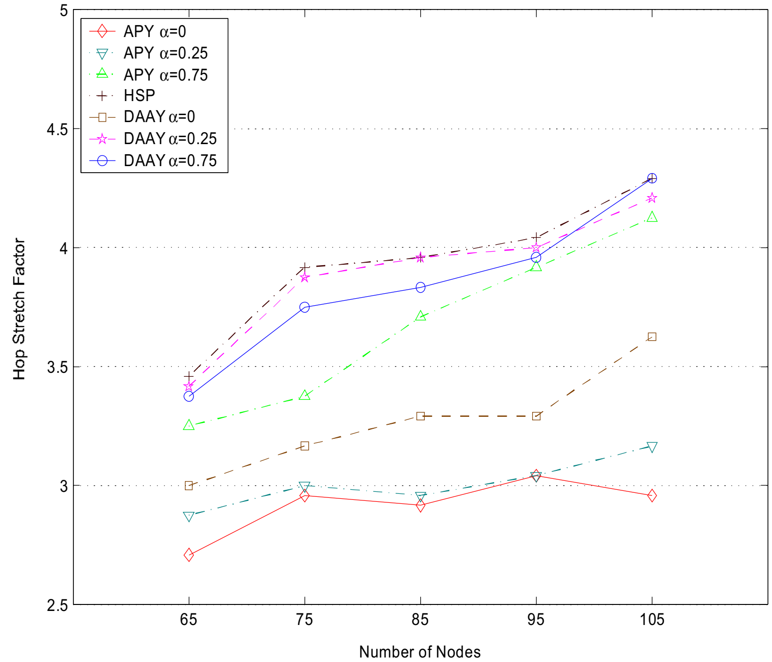


Figure 4.11: Hop stretch Factor for APY and $DAAY$ graphs

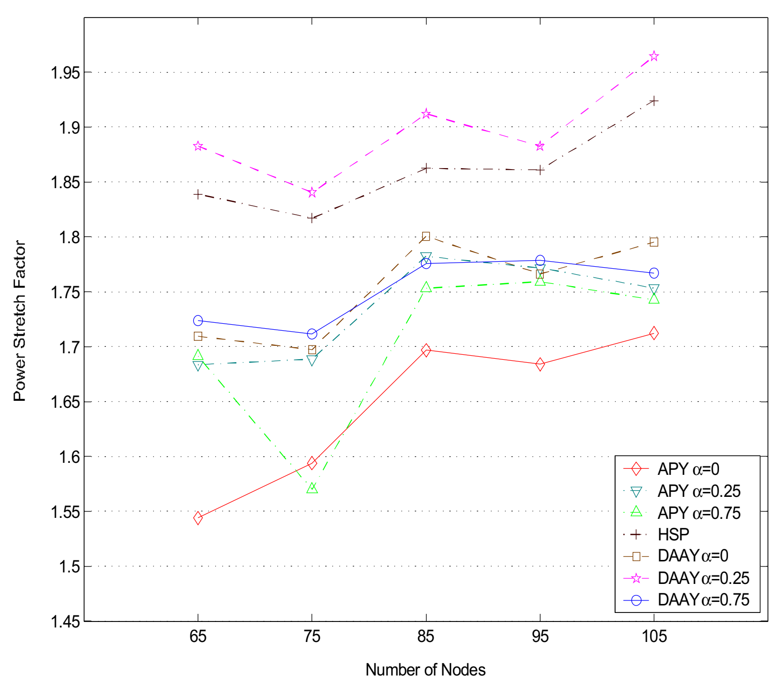


Figure 4.12: Power Stretch Factor for APY and $DAAY$ graphs

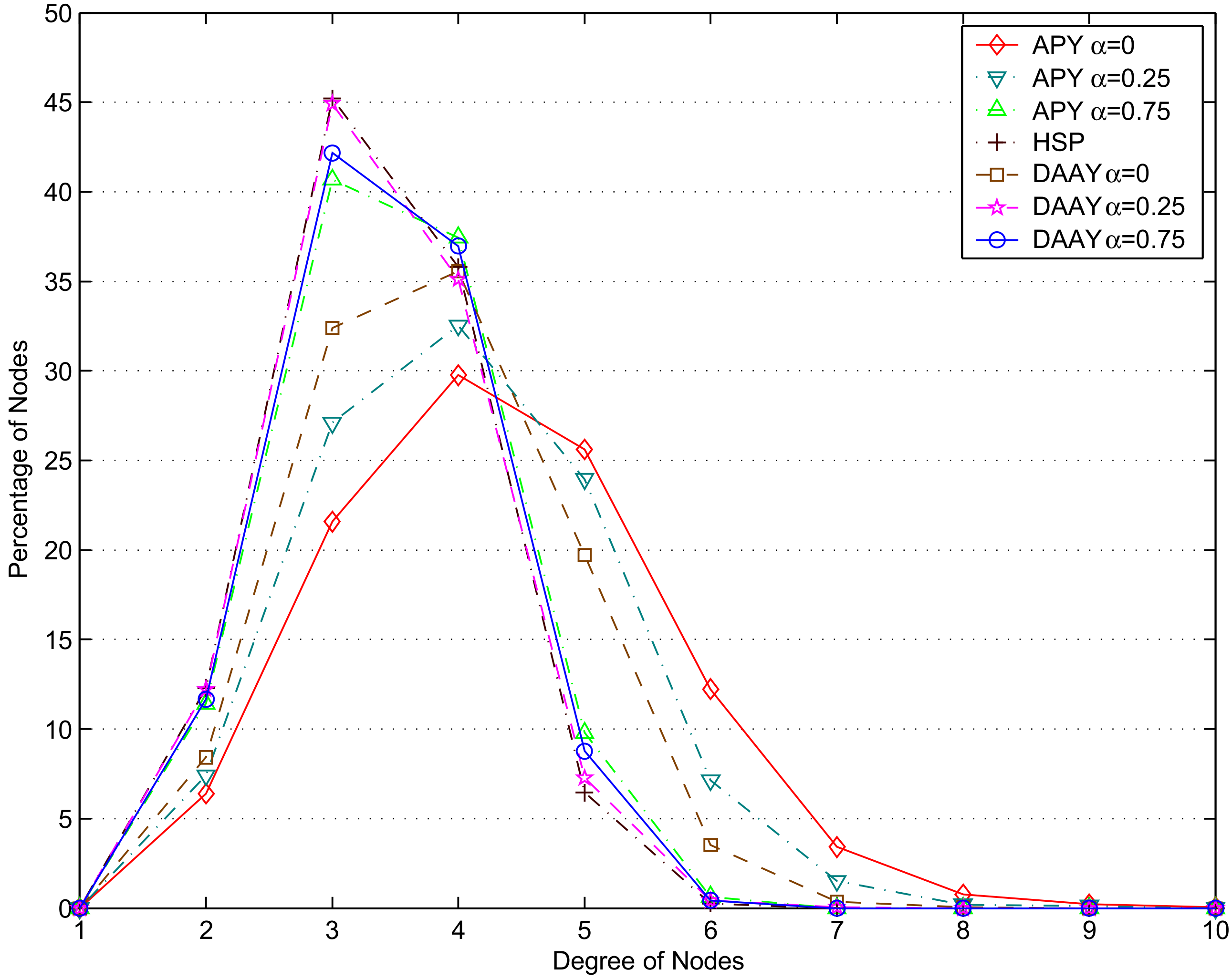


Figure 4.13: Histogram of degrees of nodes for APY and $DAAY$ graphs

4.2.2 APY and related spanning subgraphs

In the following we demonstrate a comparison of APY graph with $\alpha = 0$ and $\gamma = 0$ for the parabola with the largest width, $DAAY$ graph with $\alpha = 0$ and $p = 1$ for the cone with the largest angle, YAO graph with $k = 6$, UDG and HSP graphs and G_λ^θ graph with $\lambda = 0.51$ and $\eta = \frac{\pi}{3}$ for the largest cone angle, which we indicate it as HP.

As we can see in Figures 4.14, 4.15, 4.16 and 4.17, average node degree, maximum node degree, number of crossing edges and total weight of the APY graph are higher than $DAAY$ graph and lower than YAO graph. Specifically, node degree of the APY is less than the YAO graph which is bounded by k number of cones. All mentioned values are large for G_λ^θ graph. While APY graph has hop stretch factor and power stretch factor values that are again between the values for $DAAY$ and YAO graphs, G_λ^θ graph is lower than APY for both values. See Figures 4.18 and 4.19.

Figure 4.20 demonstrates a histogram for percentage of nodes with respect to degree of nodes. Again, APY graph has values between $DAAY$ and YAO graphs but G_λ^θ graph has higher number of nodes with higher node degree.

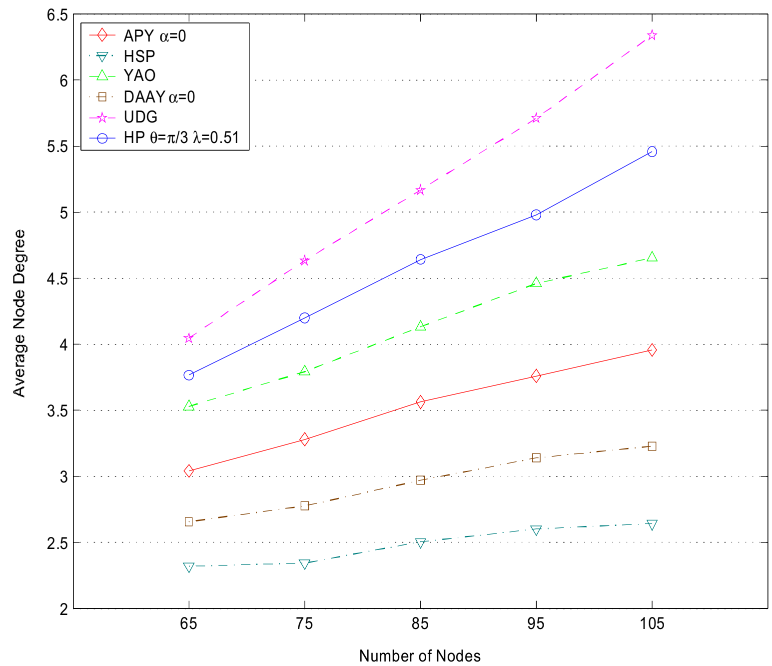


Figure 4.14: Average Node Degree for graphs with various numbers of nodes

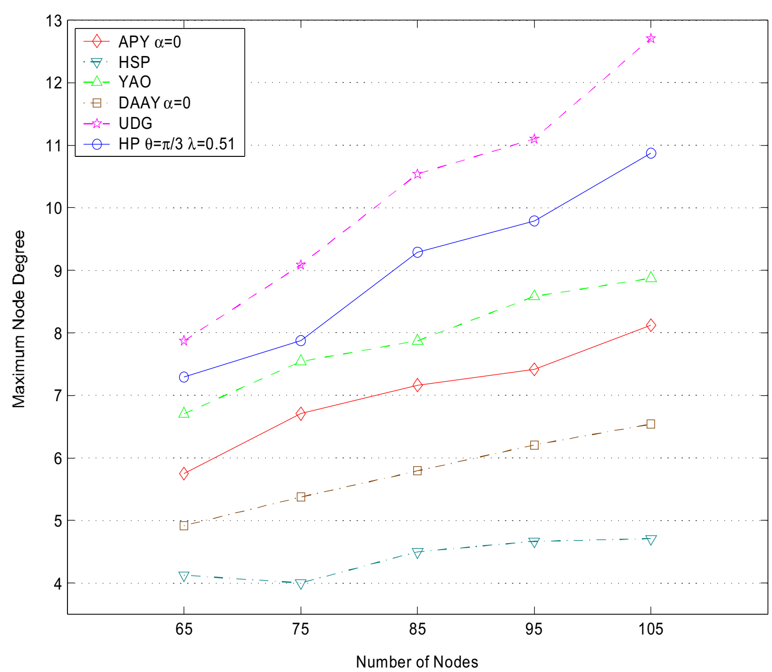


Figure 4.15: Maximum Node Degree for graphs with various numbers of nodes

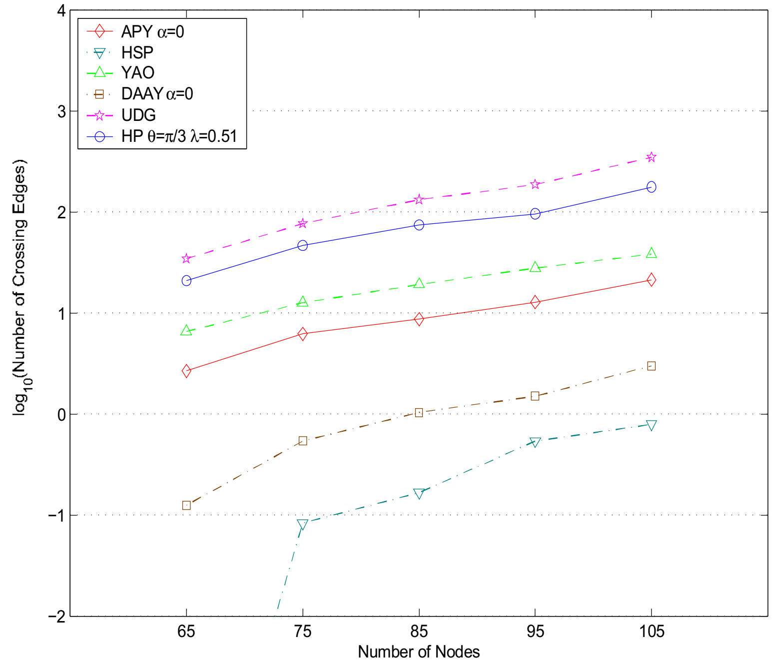


Figure 4.16: Number of Crossing Edges for graphs with various numbers of nodes

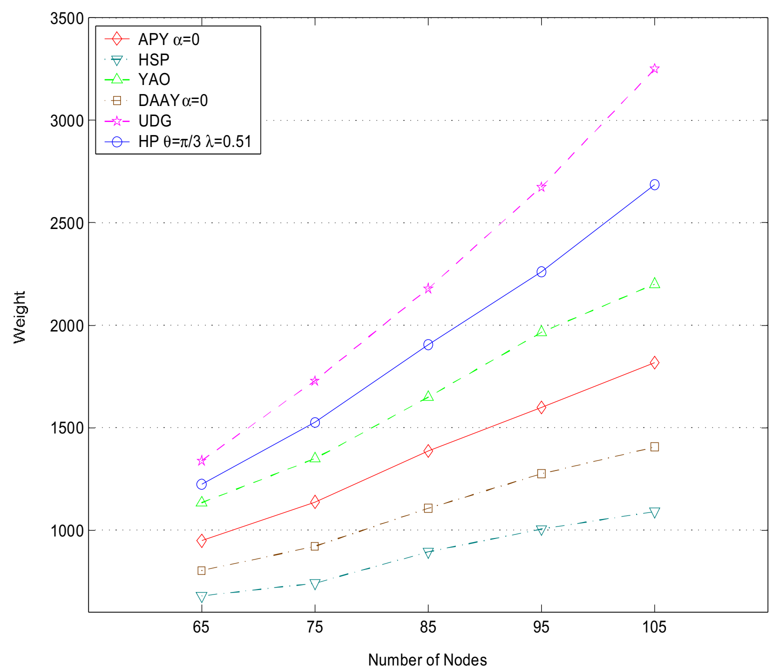


Figure 4.17: Total Euclidean Weight for graphs with various numbers of nodes

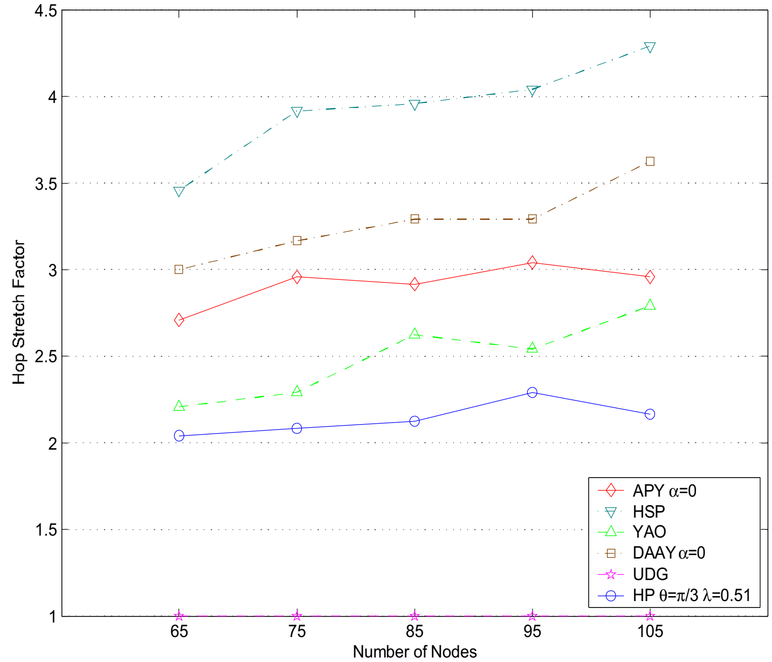


Figure 4.18: Hop stretch Factor for graphs with various numbers of nodes

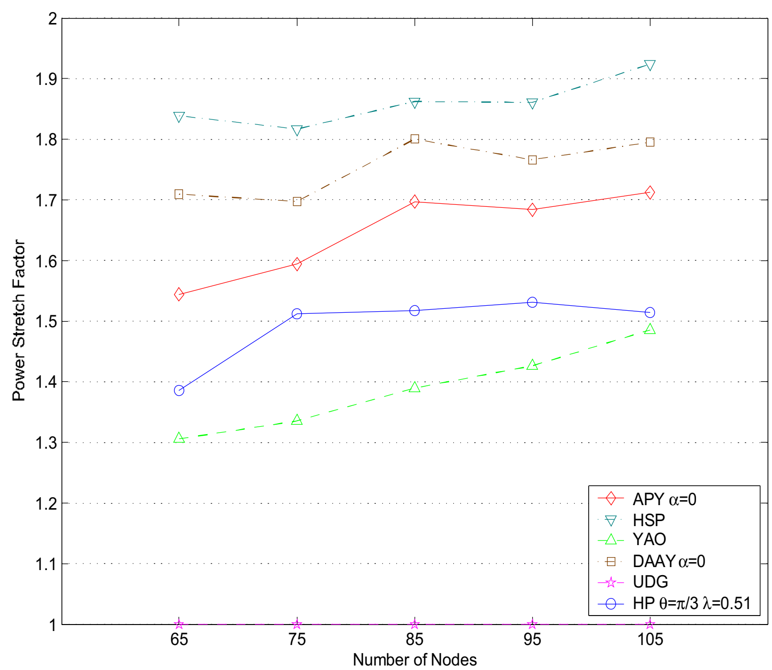


Figure 4.19: Power Stretch Factor for graphs with various numbers of nodes

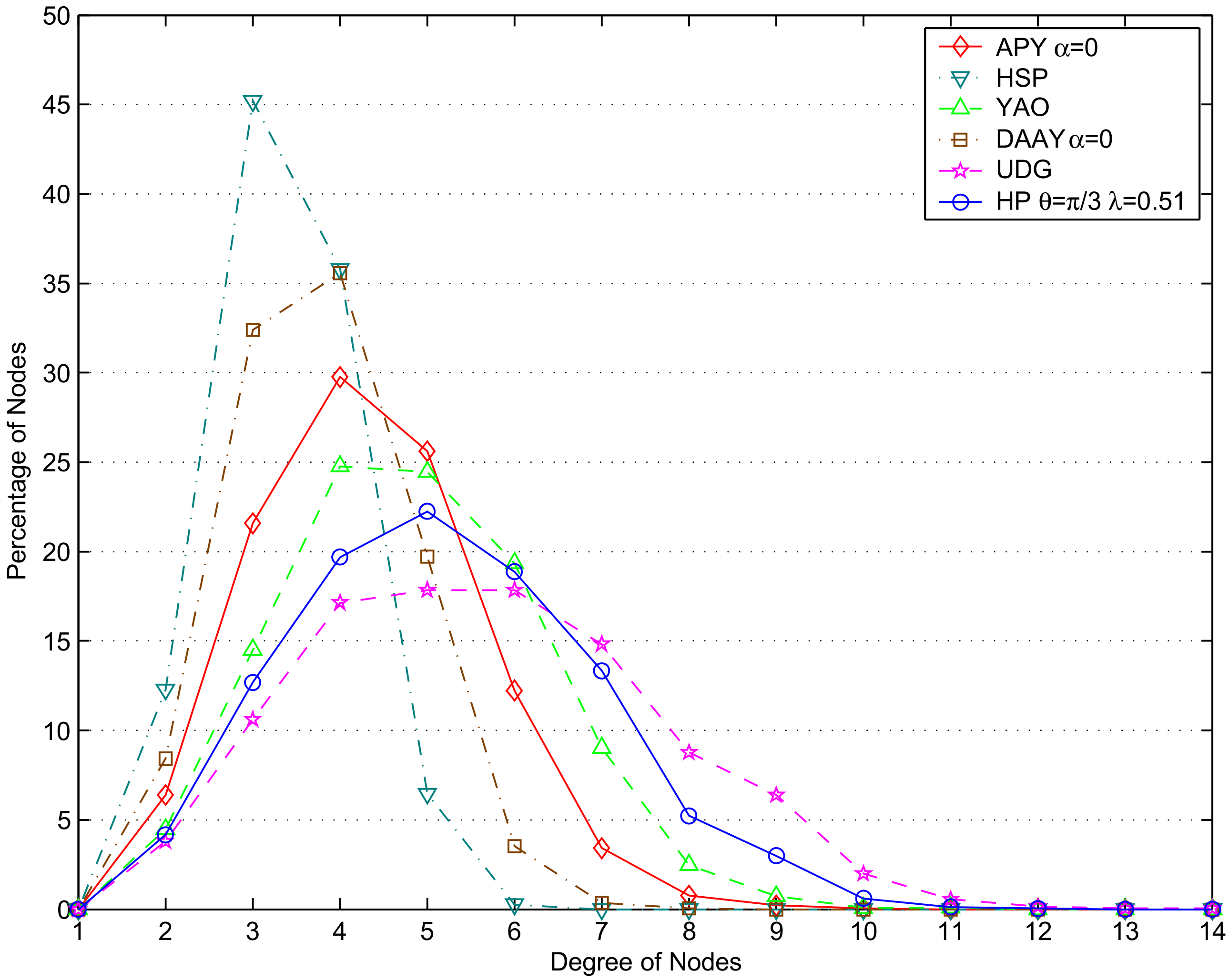


Figure 4.20: Histogram of degrees of nodes for different spanning subgraphs

4.2.3 More on APY , $DAAY$ and G_λ^θ graphs

The following is a comparison of the Adaptive Parabolic Yao graph, Displaced Apex Adaptive Yao graph and G_λ^θ graph. For all three graphs we have considered their respective parameter values such that the selected area to discard neighboring nodes is the widest, for those parameters we have placed the apex of the parabola at the two boundary values of distance from the nearest neighbor. Also a case of a narrow region for the same positions of apex is considered. The results are presented in tables 4.1, 4.2, 4.3, 4.4, 4.5 and 4.6.

As we can see, APY graph is the most dependent on its parameters α and γ while the least dependent is the G_λ^θ graph. It is shown in [4] that G_λ^θ graphs are strong t-spanners, it is reflected in its hop stretch factor and power stretch factor values in Tables 4.5 and 4.6. While with varying parameters α and γ, APY graph is able to produce spanning subgraphs of the UDG graph that have different properties suitable for different purposes and different routing algorithms.

Table 4.1: Average Node Degree

APY	$\gamma=0$		$\gamma=16$	$\gamma=64$
	$\alpha=0$	$\alpha=1$	$\alpha=0$	$\alpha=1$
	3.3	2.5	4.3	4.1
$DAAY$	$s=1$		$s=0.2$	
	$\alpha=0$	$\alpha=1$	$\alpha=0$	$\alpha=1$
	2.8	2.5	4.3	4.2
G_λ^θ	$\theta=89^\circ$		$\theta=10^\circ$	
	$\lambda=0.51$	$\lambda=0.99$	$\lambda=0.51$	$\lambda=0.99$
	4.1	4.1	4.6	4.6

Table 4.2: Maximum Node Degree

APY	$\gamma=0$		$\gamma=16$	$\gamma=64$
	$\alpha=0$	$\alpha=1$	$\alpha=0$	$\alpha=1$
	6.7	4.5	8.4	8.1
$DAAY$	$s=1$		$s=0.2$	
	$\alpha=0$	$\alpha=1$	$\alpha=0$	$\alpha=1$
	5.4	4.4	8.3	8.3
G_λ^θ	$\theta=89^\circ$		$\theta=10^\circ$	
	$\lambda=0.51$	$\lambda=0.99$	$\lambda=0.51$	$\lambda=0.99$
	7.5	7.8	8.9	8.9

Table 4.3: Number of Crossing Edges

APY	$\gamma = 0$		$\gamma = 16$	$\gamma = 64$
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	6.2	0.0	46.9	31.4
$DAAY$	$s = 1$		$s = 0.2$	
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	0.5	0.0	43.3	42.8
G_λ^θ	$\theta = 89^\circ$		$\theta = 10^\circ$	
	$\lambda = 0.51$	$\lambda = 0.99$	$\lambda = 0.51$	$\lambda = 0.99$
	42.8	43.6	70.4	71.4

Table 4.4: Total Euclidean Weight

APY	$\gamma = 0$		$\gamma = 16$	$\gamma = 64$
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	1137.5	804.55	1580.8	1489.2
$DAAY$	$s = 1$		$s = 0.2$	
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	921.78	779.68	1556.7	1547.8
G_λ^θ	$\theta = 89^\circ$		$\theta = 10^\circ$	
	$\lambda = 0.51$	$\lambda = 0.99$	$\lambda = 0.51$	$\lambda = 0.99$
	1470.5	1479.1	1700.4	1702.6

Table 4.5: Hop stretch Factor

APY	$\gamma = 0$		$\gamma = 16$	$\gamma = 64$
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	2.9583	3.5	2.0	2.2714
$DAAY$	$s = 1$		$s = 0.2$	
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	3.1667	3.9167	2.0417	2.2917
G_λ^θ	$\theta = 89^\circ$		$\theta = 10^\circ$	
	$\lambda = 0.51$	$\lambda = 0.99$	$\lambda = 0.51$	$\lambda = 0.99$
	2.3333	2.1667	1.9167	1.9167

Table 4.6: Power Stretch Factor

APY	$\gamma = 0$		$\gamma = 16$	$\gamma = 64$
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	1.594	1.6201	1.1922	1.2777
$DAAY$	$s = 1$		$s = 0.2$	
	$\alpha = 0$	$\alpha = 1$	$\alpha = 0$	$\alpha = 1$
	1.6974	1.78	1.1931	1.3833
G_λ^θ	$\theta = 89^\circ$		$\theta = 10^\circ$	
	$\lambda = 0.51$	$\lambda = 0.99$	$\lambda = 0.51$	$\lambda = 0.99$
	1.6628	1.6938	1.0948	1.0948

Chapter 5

Conclusions and future work

In this thesis we presented an algorithm to construct a spanning subgraph of the unit disk graph. We have shown the results of numerical simulations and also, comparisons to some known spanning subgraphs have been discussed. The results have demonstrated that the APY subgraphs have some interesting properties which we discuss in the following. In section 4.2 we will suggest some future research directions that can be extended from the work of this thesis.

5.1 Discussions

The APY algorithm constructs a family of subgraphs by varying the values of α and γ. The behavior of subgraphs constructed with $0 \leq \alpha < 0.5$ differs from the subgraphs with $0.5 < \alpha \leq 1$. For the values of $0.5 < \alpha \leq 1$ when $\gamma = 0$, APY is behaving more like HSP.

Although the APY graph has a conditional maximum out-degree for each node, since it depends on d_0 which is the distance between the closest nodes in the network, it is possible to control the node degree through the parameter γ which changes the width of the parabola. This shows that high out-degree happens for the nodes with closest neighbors and the algorithm keeps the shortest edges. Also, the node degree of the APY is still less than the YAO graph which is bounded by fixed k number of cones.

The APY graphs with $\alpha > 0.5$ have low number of crossing edges. These graphs are more suitable for routing algorithms that use planar graphs. For all γ, graphs with $\alpha > 0.5$ have higher hop stretch factor and lower power stretch factor compared to graphs with $\alpha < 0.5$. For both of these factors the values of the HSP graph is a critical point. Although the APY graph has relatively higher node degree and number of crossing edges than $DAAY$ graph, as the number of nodes in the network increases, the APY graph has the ability to greatly reduce the degree of nodes and number of crossing edges to a lower limit than the $DAAY$ graph. This is made possible by varying the value of α. On the other hand, hop stretch factor and power stretch factor of APY graph are clearly less than the values for $DAAY$ graph, for any number of nodes in the network. Although large number of crossing edges could be a restriction for some routing algorithms, APY graph enables to model a large network with low value for both hop stretch factor and power stretch factor.

Although we have mentioned in Section 3.2.3 that the maximum angle ϕ should be bounded from above by $\pi/3$, the experimental results show no drastic change in stretch factors when this value is exceeded. So, in practice it would appear that such a restriction on a is not necessary.

In conclusion, APY graph has similar properties to $DAAY$ graph in bounding the node out-degree, containing $EMST$ as its subgraph and being a spanner subgraph of UDG with bounded stretch factor. Moreover, it has the flexibility of producing spanning subgraph with a combination of properties which can be useful for different routing algorithms to apply on the graphs.

5.2 Future work

The work done in this thesis can be extended in the following directions.

- In this thesis we assumed that all nodes are homogeneous, however transmission range of nodes in $MANET$s vary due to natural or man-made obstacles. This is why constructing APY subgraph for a quasi-unit disk graph model of a network would be an interesting future research. Another approach would be to investigate APY subgraph of a mutual inclusion graph as a model for $MANET$.

- Since we mentioned that the restriction $\phi < \frac{\pi}{3}$ was not reflected in our experimental results, next step would be to find a better upper bound for ϕ and therefore to update the maximum out-degree and stretch factors.

- We have assumed that the nodes in our network model are static or quasi-static in a time period. Next step would be to include node mobility and update the APY graph regularly. This will not be difficult since deciding whether an edge is in the subgraph is done locally using only one-hop neighbors of a node.

- It is possible to extend the APY to the $3D$ space. In this case the parabola will be rotating about its axis of symmetry creating a $3D$ parabolic surface. Since APY uses one hop distance information, the calculations and implementation to the $3D$ space will be possible.The difficult part would be to prove the similar properties for the $2D$ subgraph.

References

[1] A.E. Abdallah. Position based routing algorithms for three-dimensional ad hoc networks. *Ph.D thesis, Concordia University*, 2009.

[2] L. Barrière, P. Fraigniaud, L. Narayanan, and J. Opatrny. Robust position-based routing in wireless ad hoc networks with irregular transmission ranges. *Wireless Communications and Mobile Computing Journal*, 3(2):141–153, 2003.

[3] S. Basagni, I. Chlamtac, V. Syrotiuk, and B. Woodward. A distance routing effect algorithm for mobility (DREAM). In *Proc. of 4th ACM/IEEE Conference on Mobile Computing and Networking (Mobicom '98)*, pages 76–84, 1998.

[4] P. Bose, P. Carmi, M. Couture, M. Smid, and D. Xu. On a family of strong geometric spanners that admit local routing strategies. Technical Report TR-07-08, SCS, Carleton University, Ottawa, Canada, 2007.

[5] P. Bose, L. Devroye, W. Evans, and D. Kirkpatrick. On the spanning ratio of Gabriel graphs and beta-skeletons. *In Proceedings of the Latin American Theoretical Infocomatics (LATIN)*, pages 479–493, Mexico, 2002.

[6] P. Bose, J. Gudmundsson, and P. Morin. Ordered theta graphs. *Computational Geometry: Theory and Applications*, 28(1):11–18, 2004.

[7] P. Bose, J. Gudmundsson, and M. Smid. Constructing plane spanners of bounded degree and low weight. *Algorithmica*, 42(3–4):249-264, 2005.

[8] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia. Routing with guaranteed delivery in ad hoc wireless networks. *Wireless Networks, volume 7*, Kluwer Academic Publishers, pages 609-616, 2001.

[9] S. Capkum, M. Hamdi, and J.P. Hubaux. GPS-free positioning in mobile ad-hoc networks. In *In Proceedings of Hawaii International Conference On System Sciences*, 2001.

[10] E. Chavez, S. Dobrev, E. Kranakis, J. Opatrny, L. Stacho, H. Tejeda, and J. Urrutia. Half-space proximal: A new local test for extracting a bounded dilation spanner. In *Proceedings of the International Conference On Principles of Distributed Systems (OPODIS 2005), volume 3974 of LNCS*, pages 235–245, Pisa, Italy, 2006.

[11] D. Dobkin, S. Friedman, and K. Supowit. Delaunay graphs are almost as good as complete graphs. *Discrete Computational Geometry*, 5(4):399-407, 1990.

[12] T. Fevens, A.E. Abdallah, T. El Salti and L. Harutyunyan. A class of orientation-invariant Yao-type subgraphs of a unit disk graph. *Proc. of the DIALM-POMC International Workshop on Foundations of Mobile Computing*, Portland, USA, 2007.

[13] K. Gabriel and R. Sokal. A new statistical approach to geographic variation analysis. *Systematic Zoology*, 18:259–278, 1969.

[14] S. Giordano, I. Stojmenovic, and L. Blazevic. Position based routing algorithms for ad hoc networks: a taxonomy. In X. Cheng, X. Huang, and D.Z. Du, editors, *Ad hoc wireless networking*, Kulwer Academic Publishers, pages 103–136, 2003.

[15] Z. Haas, and M. Pearlman. The performance of query control scheme for the zone routing protocol. *ACM/IEEE Transactions on Networking*, 9(4) pages 427-438, August 2001.

[16] D. Johnson, and D. Maltz. Dynamic Source Routing. *Mobile Computing*, Kulwer Academic Publishers, 1996.

[17] I. Kanj, and L. Perkovic. Improved stretch factor for bounded-degree planar power spanners of wireless ad-hoc networks. In S. Nikoletseas and J. D. P. Rolim, editors, *ALGOSENSORS*, volume 4240 of *LNCS*, pages 95–106. Springer, 2006.

[18] I. Kanj, and L. Perkovic. On Geometric Spanners of Euclidean Graphs and their Applications in Wireless Networks. Technical Report, DePaul University, Chicago, USA, 2007.

[19] B. Karp, and H.T. Kung. GPSR: Greedy perimeter stateless routing for wireless networks. *In ACM/IEEE International Conference on Mobile Computing and Networking*, 2000.

[20] J. Keil, and C. Gutwin. Classes of graphs which approximate the complete euclidean graph. *Discrete Computational Geometry*, 7(1):13–28, 1992.

[21] Xiang-yang Li. Applications of Computational Geometry in Wireless Networks. *Ad Hoc Wireless Networking*, Kluwer Academic Publishers, 2003.

[22] X. Li. Topology Control in Wireless Ad Hoc Networks. *Ad Hoc Networking*, IEEE Press, 2003.

[23] X.-Y. Li, G. Calinescu, P.-J. Wan, and Y. Wang. Localized delaunay triangulation with application in Ad Hoc wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 14(10):1035-1047, 2003.

[24] X.-Y. Li, P.-J. Wan, and Y. Wang. Power efficient and sparse spanner for wireless ad hoc networks. In *Proceedings of IEEE International Conference on Computer Communications and Networks (ICCCN01)*, pages 564–567, 2002.

[25] X.-Y. Li, Y. Wang, P.-J. Wan, and O. Frieder. Sparse power efficient topology for wireless networks. In *In IEEE Hawaii International Conference On System Sciences*, 2002.

[26] X.-Y. Li, Y. Wang, P.-J. Wan, W.-Z. Song, and O. Frieder. Localized low-weight graph and its applications in wireless ad hoc networks. In *In Proceedings of Twenty-Third IEEE INFOCOM 2004*, Hong Kong, China, March 2004.

[27] M. Mauve, J. Widmer, and H. Hartenstein. A survey of position-based routing in mobile ad-hoc networks. *IEEE Network Magazine*, 15(6):30–39, 2001.

[28] S. Murthy, and J. Garcia-Luna-Aceves. A routing protocol for packet radio networks. In *In Proceedings of the ACM International Conference on Mobile Computing and Networking*, pages 86–95, 1995.

[29] G. Narasimhan, and M. Smid. Geometric Spanner Networks. *Cambridge University Press*, 2007.

[30] C. Perkins and P. Bhagwat. Highly dynamic destination sequenced distance-vector routing for mobile computers. *Computer Communication Review*, pages 234-244, October 94.

[31] C. Perkins and E. Royer. Ad-hoc on-demand distance vector routing. *In Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pages 90-100, Feb 1999.

[32] R. Rajaraman. Topology control and routing in ad hoc networks: A survey. *SICACT News*, 33:60–73, 2002.

[33] W.-Z. Song, Y. Wang, X.-Y. Li, and O. Frieder. Localized algorithms for energy efficient topology in wireless ad hoc networks. *MONET*, 10(6):911–923, 2005.

[34] G. Toussaint. The relative neighbourhood graph of a finite planar set. *Pattern Recognition*, 12(4):261–268, 1980.

[35] Y. Wang and X.-Y. Li. Localized construction of bounded degree and planar spanner for wireless ad hoc networks. *MONET*, 11(2):161-175, 2006.

[36] Y. Wang, X.-Y. Li, and O. Frieder. Distributed spannar with bounded degree for wireless ad hoc networks. *International Journal on Foundations of Computer Science*, 14(2):183–200, 2003.

[37] A.-C. Yao. On constructing minimum spanning trees in k-dimensional spaces and related problems. *SIAM Jurnal of Computing*, 11:721–736, 1982.