



HAL
open science

Hypocomb: Bounded-degree Localized Geometric Planar Graphs for Wireless Ad Hoc Networks

Xu Li, Nathalie Mitton, Isabelle Simplot-Ryl, David Simplot-Ryl

► **To cite this version:**

Xu Li, Nathalie Mitton, Isabelle Simplot-Ryl, David Simplot-Ryl. Hypocomb: Bounded-degree Localized Geometric Planar Graphs for Wireless Ad Hoc Networks. *IEEE Transactions on Parallel and Distributed Systems*, 2013, 24 (7), pp.1341-1354. 10.1109/TPDS.2012.180 . hal-00704682

HAL Id: hal-00704682

<https://inria.hal.science/hal-00704682>

Submitted on 3 Dec 2013

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Hypocomb: Bounded-degree Localized Planar Graphs for Wireless Ad Hoc Networks

Xu Li, Nathalie Mitton, Isabelle Simplot-Ryl, and David Simplot-Ryl

Abstract—We propose a radically new family of geometric graphs, *i.e.*, Hypocomb (HC), Reduced Hypocomb (RHC) and Local Hypocomb (LHC). HC and RHC are extracted from a complete graph; LHC is extracted from a Unit Disk Graph (UDG). We analytically study their properties including connectivity, planarity and degree bound. All these graphs are connected (provided the original graph is connected) planar. Hypocomb has unbounded degree while Reduced Hypocomb and Local Hypocomb have maximum degree 6 and 8, respectively. We present a construction algorithm for these graphs and analyze its time complexity. Hypocomb family graphs are promising for wireless ad hoc networking. We report our numerical results on their average degree and their impact on FACE routing. We discuss their potential applications and pinpoint some interesting open problems for future research.

Index Terms—Geometric planar, graph planarization, Hypocomb, wireless ad hoc networks

1 INTRODUCTION

A planar graph is a sparse graph where edges intersect only at their end vertices. It has $O(|V|)$ (precisely, less than $3|V| - 6$) edges, with V being vertex set, asymptotically smaller than the maximum $O(|V|^2)$ (*e.g.*, in a complete graph where there is an edge between every pair of vertices). Planar graphs have been widely adopted in different domains to solve various problems, *e.g.*, road/street pattern design in city planning, facility layout design in operations research, image segmentation in computer vision, spanning tree construction in telecommunication, to list a few. In these applications, the position of all vertices are known, and an edge can be added between any two vertices. Planarization is thus equivalent to an edge removal process on a complete graph with connectivity preservation. In some other cases, edge addition is subject to a distance constraint, giving rise to the problem of planarization on a Unit Disk Graph (UDG). In the sequel, we always assume that a given UDG is connected, and a pair of intersecting (or crossover) edges imply two edges that intersect, but not at their end vertices.

Define the unit circle $C_\gamma(a)$ of a vertex a as the circle of radius equal to a unit distance γ and centered at a . The unit disk $D_\gamma(a)$ of a is the area enclosed by $C_\gamma(a)$. In UDG, there is an edge between two vertices a and b and they are said ‘adjacent to’ or ‘neighboring’ each other if and only if $b \in D_\gamma(a)$ (equivalently, $a \in D_\gamma(b)$). We denote by $V_{nbr}(a)$ the closed neighborhood (neighbor set) of a (including a) and by $V_{nbr}(a, b)$ the closed common neighborhood of a and b .

Wireless ad hoc networks (*e.g.*, sensor networks) where nodes have the same maximum transmission range γ (unit distance) are commonly modeled as UDG. In such networks, each node is static and assumed to know its own geographic position by attached GPS device or some other means. Two nodes are neighbors (*i.e.*, have an edge in between) if and only if they are within each other’s transmission range (*i.e.*, unit disk). Periodic ‘hello’ message is a basic ad hoc networking technique for neighborhood discovery [?]. By this technique, each node is able to gather the location information of all neighboring nodes. In the past decade, several well-known position-based ad hoc routing protocols [?] were proposed. They all rely on planar network topology for guaranteeing packet delivery. In general, UDG is not planar. A planar subgraph has to be extracted through a planarization procedure.

In wireless networks, nodes share the communication media and have limited channel capacity. The main communication cost is therefore message transmissions. To minimize the control overhead on the network, graph planarization ought to be carried out in a distributed fashion without resorting to any global knowledge and with a minimal total number of message transmissions per wireless node. Ideally, it involves no message transmission in addition to the built-in ‘hello’ message. Packets have constant size at MAC layer. Transmission of a long message requires message fragmentation and leads to increased number of transmissions. As far as energy efficiency is concerned, long message consumes more transmission power than short message and should be avoided. Thus as an additional requirement, no modification should be made to the default ‘hello’ message (normally containing constant-sized information such as sender position) during planarization. In summary, graph planarization in wireless ad hoc networks is expected to be a *strictly localized* procedure, where each node makes consistent planarization decision indepen-

- The authors are affiliated with INRIA Lille - Nord Europe, Parc scientifique de la haute borne, 50 avenue Halley, 59650 VILLENEUVE D’ASCQ, France. Phone: +33 3 62 53 15 50; fax: +33 3 20 71 70 42; email: {xu.li, nathalie.mitton, isabelle.simplot-ryl, david.simplot-ryl}@inria.fr.
- A preliminary version [?] of this article was published in the proceedings of IEEE INFOCOM 2011.

dently using 1-hop neighborhood information only.

There exist a few strictly localized planar graphs such as Gabriel Graph (GG), Relative Neighborhood Graph (RNG) [?], Partial Delaunay triangulation (PDel) [?], Local Minimum Spanning Tree (LMST) [?] and a few non-strictly localized planar graphs such as Localized Delaunay triangulation (LDel) [?], [?]. The *degree* $\Delta(G)$ of a graph G is the maximum node degree in the graph. It is often desirable that $\Delta(G)$ is small and bounded above by a constant. In wireless communications, a small node degree reduces the contention and interference and helps to mitigate the hidden and exposed terminal problems at MAC layer. In bluetooth scatternets, each node is required to have maximum degree 7. All the above local planar graphs but LMST have unbounded degree in nature. Li et al. [?] modified RNG construction such that the degree is limited to a small constant. However, the modification requires each vertex to be associated a unique identifier (ID), which does not normally exist, for example, in wireless sensor networks. Li et al. [?] showed that degree can be limited to a constant with connectivity and planarity preservation (if the original graph is connected planar) by applying Yao structure [?]. A review of the above planar graphs can be found in Section ??.

In this article, we propose a radically new family of geometric planar graphs, completely different from any known graph, and focus on their theoretical properties. We first introduce *Hypocomb* (*Hypotenuse-comb*), which is the ‘dual’ (an abused use of term duality) of a truncated mesh [?] referred to as *Besh* (*Blocked-mesh*). Given a set of vertices in the Euclidean plane, Besh is constructed by drawing rays synchronously from each vertex in four directions and allowing distance-based blocking when they meet each other. Hypocomb is obtained by linking vertices that have a ray-blocking relation in Besh. We prove that Hypocomb is connected planar with unbounded degree. Then we propose to reduce its maximum degree to 6 by applying constrained edge creation rule, without jeopardizing its connectivity and planarity: link two vertices if and only if they have a mutual ray-blocking relation. The resulting Hypocomb is called *Reduced Hypocomb*. After that, we present *Local Hypocomb* on the basis of UDG. It is constructed in a strictly localized manner, by removing any UDG edge that does not belong to the Reduced Hypocomb of the closed common neighborhood of its end vertices. We prove that Local Hypocomb remains connected planar and has slightly larger degree 8. It may serve as alternative graph in geographic routing for providing delivery guarantee in wireless ad hoc networks. We present, along with complexity analysis, a construction algorithm for Hypocomb family graphs. Through simulation we study their average degree and their impact on the well-know FACE routing protocol [?], in comparison with widely-adopted *Delaunay triangulation*, *Localized Delaunay triangulation*, and *Gabriel Graph*. Simulation results imply that Local Hypocomb, when used, leads to similar FACE

routing performance as Gabriel Graph. We indicate that Hypocomb and Reduced Hypocomb may be built in a localized way among actor nodes in emerging wireless sensor and actor networks and provide a generic solution to the challenging actor-actor coordination problem. None of the existing graphs has this potential.

The remainder of this article is organized as follows. We review existing local planar graphs in Section ?. We propose Hypocomb, Reduced Hypocomb and Local Hypocomb and analyze their connectivity, planarity and degree bound in Section ?? - ??, along with numeric results being reported in Section ?. We conclude the article by describing some potential applications of Hypocomb family graphs and pinpointing some open problems for future research in Section ??.

2 RELATED WORK

There are only a few localized planar graphs in the literature. Given a vertex set V in the Euclidean plane, in the following we will briefly introduce how to construct these graphs. The containment relations among these graphs are given below, while TABLE ?? lists their properties in comparison with our newly proposed Hypocomb family graphs.

$$\text{MST} \subseteq \begin{matrix} \text{LMST} \\ \text{RNG}' \end{matrix} \subseteq \text{RNG} \subseteq \text{GG} \subseteq \begin{matrix} \text{PDel} \subseteq \text{Del} \\ \text{LDel} \end{matrix}$$

A *Minimum Spanning Tree* (MST) is a subgraph connecting all the vertices with weighted edges that lead to minimum total weight. If edges are weighted by Euclidean distance of their end vertices (as in our context here), it is called Euclidean MST, and it has degree bounded above by 6 according to [?]. In general, V may have many MST unless each edge has unique weight. MST cannot be computed locally, *i.e.*, each node cannot determine which edges are in MST by purely using the information of the nodes within some constant hops [?]. MST is not spanner, *i.e.*, having no constant spanning ratio. The *spanning ratio* of a graph is the maximum ratio of the Euclidean length of the shortest path connecting two arbitrary vertices in the graph and their direct Euclidean distance.

A *Local MST* (LMST) [?] is a connected subgraph of UDG, constructed locally using 1-hop neighborhood information as follows: at each vertex u , compute the MST of the sub-graph of $V_{nbr}(u)$; add incident edge uw to LMST if and only if the edge is in both $\text{MST}(V_{nbr}(u))$ and $\text{MST}(V_{nbr}(w))$. LMST contains MST as subgraph and has the same degree bound 6. In [?], it is proved that LMST is also planar, and the notion is extended to k -Local MST (LMST $_k$) with k -hop neighborhood information being used. LMST is not spanner either.

Gabriel Graph (GG) is built by connecting any two vertices u and w if and only if the closed disk $\text{disk}(u, w)$ having uw as diameter contains no other vertex from V , while *Relative Neighborhood Graph* (RNG) is built by connecting u and w if and only if the interior of the

TABLE 1
A summary view of common planar graphs

| Graphs | ID | Locality | Degree Bound | Spanner |
|--------|-----|----------|--------------|---------|
| LMST | no | 1-hop | 6 | no |
| RNG' | yes | 1-hop | 6 | no |
| RNG | no | 1-hop | no | no |
| GG | no | 1-hop | no | no |
| PDel | no | 1-hop | no | no |
| LDel | no | 2-hop | no | yes |
| LHC | no | 1-hop | 8 | ? |
| RHC | no | global | 6 | ? |
| HC | no | global | no | ? |
| MST | no | global | 6 | no |
| Del | no | global | no | yes |

'?' implies that the property is unknown at the time of this writing.

lune $lune(u, w)$ of u and w (i.e., the intersection of the two circles of radius $|uw|$ centered at u and w) contains no other vertex. It is proved that both GG and RNG are connected planar if the original graph is UDG and that each of them can be constructed strictly locally by each vertex checking the construction condition for its neighbors only. GG and RNG are so-called proximity graphs [?]. They both belong to the general class of β -skeletons [?]. None of them is a spanner, and none of them has bounded degree. A study of their spanning ratio in relation with $|V|$ was presented in [?].

Assuming each vertex is associated with a unique ID, a modified RNG, called RNG', was proposed in [?]. RNG' contains all edges uw such that the interior of $lune(u, w)$ contains no vertex, and (2) there is no vertex v on the boundary of $lune(u, w)$ such that $ID(v) < ID(w)$ and $|vw| < |uw|$, and (3) there is no vertex v on the boundary of $lune(u, w)$ such that $ID(v) < ID(u)$ and $|vu| < |uw|$, and (4) there is no vertex v on the boundary of $lune(u, w)$ such that $ID(v) < ID(u)$, $ID(v) < ID(w)$, and $|vu| = |uw|$. RNG' is a subgraph of RNG. It is proved that RNG' has maximum degree 6 and contains MST as subgraph.

A *Delaunay triangulation* (Del) is built by connecting any two vertices $u, w \in V$ if and only if the circumscribe circle of the triangle defined by u, w and any other vertex $v \in V$ is empty. Given V , there may be more than one Delaunay triangulation, but only if V contains four or more co-circular vertices. Del has constant spanning ratio [?]. Del cannot be constructed locally, because it may contain arbitrary long edges.

A connected planar was proposed for UDG on the basis of Del and under the assumption of no four co-circular vertices in [?]. The graph is a subset of Del and thus named *Partial Delaunay triangulation* (PDel). It contains only a few more edges than GG. To construct PDel, each node u for each $w \in V_{nbr}(u)$ checks the following conditions: (1) $disk(u, w)$ is empty (i.e., uw belongs to

TABLE 2
A list of important denotations

| | |
|-----------------------|--|
| V | a given vertex set (creating points) |
| $\Delta(G)$ | degree of graph G |
| $C_\gamma(a)$ | unit circle of a ; γ is unit distance |
| $D_\gamma(a)$ | unit disk of a ; γ is unit distance |
| $V_{nbr}(a)$ | closed neighbor set of a (in UDG) |
| $V_{nbr}(a, b)$ | closed common neighbor set of a, b |
| R_a^{dir} | ray from a in direction dir |
| \overline{dir} | opposite direction to dir |
| \hat{dir} | set of directions perpendicular to dir |
| $Q_a(dir, dir')$ | quadrant of a defined by $R_a^{dir}, R_a^{dir'}, dir' \in \hat{dir}$ |
| $a \xrightarrow{u} b$ | a collinearly blocks b at u |
| $a \perp^u b$ | a orthogonally blocks b at u |
| $\square ab$ | axis-parallel rectangular area with ab as diagonal line |
| $mlen(\square ab)$ | length of the longer side of $\square ab$ |
| $disk(u, v)$ | disk of diameter uv |

GG); (2) $disk(u, w)$ contains vertices only on one side of uw , with x being one of those vertices that maximizes $\angle uxw$ in triangle Δuxw such that $\angle uxw + \angle uyw < \pi$, where $\angle uyw$ is in triangle Δuyw and maximum with y being from a subset of vertices (referred to as search set) on the other side of uw . The search set can be defined either as the set of common neighbors of u and w (1-hop knowledge suffices for planarization in this case) or as the 2-hop neighbor set of u . If any of these two conditions holds, edge uw is added to PDel. PDel has unbounded degree. Its degree is limited to 7 after Yao structure [?] is applied. PDel is not a spanner.

Another Del-based connected planar graph, called *Localized Delaunay triangulation* (LDel), was proposed for UDG independently in [?] and [?]. As PDel, it contains GG as subgraph; unlike PDel, it has a good spanning ratio. The planarization process works as follows: $\forall u \in V$, compute $Del(V_{nbr}(u))$; $\forall w \in V_{nbr}(u)$, uw is added to LDel if $uw \in Del(V_{nbr}(u))$ and $\nexists v \in V_{nbr}(u)$ such that $u, w \in V_{nbr}(v)$ and $uw \notin Del(V_{nbr}(v))$. Construction of LDel requires 2-hop neighborhood information. LDel has unbounded degree. In [?], the degree of LDel is limited to $19 + 2\pi/\alpha$, where $0 < \alpha \leq \pi/3$, by applying Yao [?] structure, without scarifying its spannerity. Note that Yao graph itself does not guarantee planarity.

3 HYPOCOMB

Given as creating points a vertex set V in the Euclidean plane, we first show how to build a connected planar graph, named *Hypocomb*, by adding edges between them. This is equivalent to removing edges from a complete graph of V . For ease of understanding, we divide our graph planarization process into two steps and present

Fig. 1. Besh and Hypocomb. The solid small dots are Besh points; the thin dashed lines are Besh edges. Hypocomb edges are drawn in green thick links (solid and dashed). Solid thick green lines belong to Reduced Hypocomb

them separately. TABLE ?? lists the important denotations to be used throughout the rest of the article.

For all $a, b \in V^2$, $a \neq b$, rays issued from a and b are *collinear* if they have the same X or Y coordinate. Define *north* as the positive direction of the Y axis, and *east* as the positive direction of the X axis. Let $T = \{\text{north}, \text{west}, \text{south}, \text{east}\}$. For each $\text{dir} \in T$, $\overline{\text{dir}}$ is the opposite direction and $\hat{\text{dir}}$ the set of perpendicular directions, e.g., if $\text{dir} = \text{north}$, then $\overline{\text{dir}} = \text{south}$, $\hat{\text{dir}} = \{\text{west}, \text{east}\}$. The border of V is the smallest rectangle containing V and parallel to both axes.

3.1 The first step: Besh

At the first step, we build an auxiliary structure, referred to as *Besh* [?]. We synchronously grow from all $v \in V$ four rays R_v^{north} , R_v^{west} , R_v^{south} and R_v^{east} with mutual angle of $\frac{\pi}{2}$, respectively in the north, west, south and east direction. The growth of these rays is limited by the border of V . If there was no further constraint, we would obtain a mesh. However we indeed apply a blocking rule [?] to control ray growth.

Definition 1 (Blocking rule). $\forall a, b \in V$, $a \neq b$ and $\forall \text{dir}, \text{dir}' \in T$, $\text{dir} \neq \text{dir}'$, if R_a^{dir} and $R_b^{\text{dir}'}$ meet at point u , R_a^{dir} will stop growing only in any of the following cases:

- 1) $|au| > |bu|$;
- 2) $|au| = |bu|$, $\text{dir}' \in \hat{\text{dir}}$ and $\text{dir} = \text{north}$ or south ;
- 3) $|au| = |bu|$ and $\text{dir}' = \overline{\text{dir}}$.

When this happens, we say ' b blocks a at u . In the first two cases (orthogonal blocking), it is expressed as $b \stackrel{u}{\not\leftrightarrow} a$ (or, $R_b^{\text{dir}'} \not\leftrightarrow R_a^{\text{dir}}$); in the last case (collinear blocking), it is expressed as $a \stackrel{u}{\not\leftrightarrow} b$ (or, $R_a^{\text{dir}} \not\leftrightarrow R_b^{\text{dir}'}$).

Use of the blocking rule causes some rays to stop growing early, before hitting the border of V , and yields a truncated mesh, which is our so-called *Besh* (standing for *blocked mesh*). The *Besh*, denoted by $BS(V)$, is defined by a vertex set and an edge set. The former contains the creating points V and added *Besh points*, where the blocking rule is engaged; the latter contains the edges between the vertices. In $BS(V)$, each cell is either a triangle or a quadrilateral. The creating points (i.e., vertices in V) whose rays define the perimeter of a cell is called the *defining points* of the cell. Each cell obviously has at least two, and at most four, defining points. For a Besh cell, with respect to a given corner vertex (which is either a defining point or a Besh point), the *diagonal defining points* are the defining points that are not collinear with the vertex.

Figure ?? shows a Besh structure created by using 8 points a, b, \dots, h , whose border is marked by a thick rectangle. The solid small dots are Besh points; the thin dashed lines are Besh edges. Besh cell $bswt$ is defined by a, b and h . For this cell, the diagonal defining points with respect to b is a and h , and that with respect to s is a . Examples of blocking case 2 are $b \stackrel{u}{\not\leftrightarrow} a$, $a \stackrel{t}{\not\leftrightarrow} b$ and $h \stackrel{w}{\not\leftrightarrow} a$. An example of blocking case 3 is $b \stackrel{s}{\not\leftrightarrow} c$. Notice $|bs| = |cs| < |ds| = |hs|$. By the blocking rule, d (similarly, h) can be blocked by all the other three vertices at s . To reduce this ambiguity, we define the following important prioritized blocking policy, by which only the blocking from b and c is recognized at s in Fig. ??.

Definition 2 (Prioritized blocking). $\forall a, b, c \in V$ and $a \neq b \neq c$, when $a \stackrel{u}{\not\leftrightarrow} b$ and $c \stackrel{u}{\not\leftrightarrow} b$ are both possible for the same u , b is considered being blocked by c rather than a .

Definition 3 (Quadrant). Given a point a , $\forall \text{dir} \in T$ and $\forall \text{dir}' \in \hat{\text{dir}}$, R_a^{dir} and $R_a^{\text{dir}'}$ defines a quadrant $Q_a(\text{dir}, \text{dir}')$. As such, a has four different quadrants.

Lemma 1. $\forall a, b \in V$, $a \neq b$, $\text{dir} \in T$, $\text{dir}' \in \hat{\text{dir}}$, $b \in Q_a(\text{dir}, \text{dir}')$ and $|bb'| \leq |ab'|$ with b' being the projection of b on $R_a^{\text{dir}'}$, if $\exists c \in V$, $c \in Q_a(\text{dir}, \text{dir}')$ and $c \neq b$ such that c blocks $R_b^{\text{dir}'}$ at u , then $|cc'| \leq |ac'|$ where c' is the projection of c on R_a^{dir} .

Proof: We prove this lemma by case study with illustrations being given in Fig. ?. According to the way that c blocks $R_b^{\text{dir}'}$, we have three cases to consider.

- 1) $R_c^{\text{dir}} \not\leftrightarrow R_b^{\text{dir}'}$: This is the case of $c = c_1$, $c' = c'_1$ and $u = u_1$. We know $|bb'| \leq |ab'|$ and $|cu| \leq |bu|$. Then $|cc'| = |bb'| - |bu| \leq |ab'| - |cu| = |ac'|$.
- 2) $R_c^{\text{dir}'} \not\leftrightarrow R_b^{\text{dir}'}$: This is the case of $c = c_2$, $c' = c'_2$ and $u = u_2$. We have $|cc'| = |bb'| - |bu| \leq |bb'| \leq |ab'| \leq |ab'| + |b'c'| = |ac'|$.
- 3) $R_c^{\text{dir}'} \not\leftrightarrow R_b^{\text{dir}'}$: This is the case of $c = c_3$, $c' = c'_3 = b'$ and $u = u_3$. $|cc'| < |bc'| < |ac'|$.

Note that c' is within distance $|bb'|$ from b' . \square

Lemma ?? tells us an important property of the blocking rule: if b blocks a orthogonally at u in the case that a and b are the only vertices in V , then a must be blocked

(a) Lemma ?? (b) Lemma ??

Fig. 2. Illustrations for Besh analysis

Algorithm 1 Blocking-Detection(V, a, dir)

Require: $a \in V$ and $dir \in T$

- 1: $dist := \infty$
- 2: Let dir' and dir'' be the two elements in \hat{dir}
- 3: $c := \arg(\min_{b \in V, b \neq a, b \in R_a^{dir}} |ab|)$
- 4: **if** $c \neq null$ **then**
- 5: $dist := \frac{1}{2}|ac|$
- 6: $d := \text{mid point of } a \text{ and } c$
- 7: $S_1 := \{(b, p) | b \in V \text{ such that } R_b^{dir'} \not\perp R_c^{dir} \text{ in case of } V = \{b, c\}\}$
- 8: $S_2 := \{(b, p) | b \in V \text{ such that } R_b^{dir''} \not\perp R_c^{dir} \text{ in case of } V = \{b, c\}\}$
- 9: $S'_1 := \{m | m \in S_1 \text{ and } |\text{Second}(m)c| \leq dist \text{ such that } \nexists t \in S_1, \text{ First}(t) \text{ blocks } R_{\text{First}(m)}^{dir'} \text{ in case of } V = \{\text{First}(m), \text{First}(t)\}\}$
- 10: $S'_2 := \{m | m \in S_2 \text{ and } |\text{Second}(m)c| \leq dist \text{ such that } \nexists t \in S_2, \text{ First}(t) \text{ blocks } R_{\text{First}(m)}^{dir''} \text{ in case of } V = \{\text{First}(m), \text{First}(t)\}\}$
- 11: **if** $S'_1 \cup S'_2 \neq \emptyset$ **then**
- 12: $dist := |\text{Second}(\arg(\min_{m \in S'_1 \cup S'_2} |\text{Second}(m)c|))a|$
- 13: **end if**
- 14: **end if**
- 15: $W_1 := \{(b, p) | b \in V \text{ such that } R_b^{dir'} \not\perp R_a^{dir} \text{ in case of } V = \{a, b\}\}$
- 16: $W_2 := \{(b, p) | b \in V \text{ such that } R_b^{dir''} \not\perp R_a^{dir} \text{ in case of } V = \{a, b\}\}$
- 17: $W'_1 := \{m | m \in W_1 \text{ and } |\text{Second}(m)a| \leq dist \text{ such that } \nexists t \in W_1, \text{ First}(t) \text{ blocks } R_{\text{First}(m)}^{dir'} \text{ in case of } V = \{\text{First}(m), \text{First}(t)\}\}$
- 18: $W'_2 := \{m | m \in W_2 \text{ and } |\text{Second}(m)a| \leq dist \text{ such that } \nexists t \in W_2, \text{ First}(t) \text{ blocks } R_{\text{First}(m)}^{dir''} \text{ in case of } V = \{\text{First}(m), \text{First}(t)\}\}$
- 19: **if** $W'_1 \cup W'_2 \neq \emptyset$ **then**
- 20: $dist' := \min_{m \in W'_1 \cup W'_2} |\text{Second}(m)a|$
- 21: $ret := \{m | m \in W'_1 \cup W'_2 \text{ such that } |\text{Second}(m)a| = dist'\}$
- 22: **else if** $c \neq null$ **then**
- 23: $ret := \{(c, d)\}$
- 24: **else**
- 25: $ret := \emptyset$
- 26: **end if**
- 27: **return** ret

by some c (possibly identical to b) orthogonally at u' within distance $|bu|$ from u when V contains also other vertices. Based on this result, we develop a computer algorithm named *Blocking-Detection* to support Besh construction. Given $a \in V$ and $dir \in T$, this algorithm returns the set of vertices (at most 2 by the prioritized blocking policy) that block R_a^{dir} and the associated Besh point (a single point). If no vertex blocks R_a^{dir} , it returns an empty set. The pseudo codes are given in Algorithm ???. Functions $First()$ and $Second()$ return respectively the first and the second elements of an input pair. Function $arg()$ returns the argument of an input function.

Examine Algorithm ???. In Line 3, we find the vertex c that is located on R_a^{dir} and nearest to a in $O(|V|)$ time. Assume that R_a^{dir} is not blocked by anybody else. This vertex c will collinearly block R_a^{dir} at the mid

point d of a and c if R_c^{dir} is not orthogonally blocked before reaching d . Thus we perform further check on this blocking potential. In Lines 7 and 8 we compute the sets S_1 and S_2 of vertices (together with the corresponding blocking points) that have the potential to orthogonally block R_c^{dir} . The computation can be finished in $O(|V|)$ time. In the light of Lemma ??, we in Line 9 reduce S_1 to S'_1 by removing the vertices that are not able to block R_c^{dir} before d , due to being blocked by other vertices in S_1 . The computation time is at most $O(|V|^2)$. Lines 10 reduces S_2 to S'_2 in a similar way in $O(|V|^2)$ time. If the union of S'_1 and S'_2 is empty (namely, no vertex blocks R_c^{dir} orthogonally), then we can conclude that R_c^{dir} is able to block R_a^{dir} . Otherwise, R_c^{dir} will not block R_a^{dir} , and R_a^{dir} will reach the point where R_c^{dir} is orthogonally blocked by a vertex and be blocked by that same vertex. Hence, the result from Lines 4-13 is a coarse upper bound of the length of R_a^{dir} in Besh, stored in variable $dist$. It is infinity in the case that c does not exist (without considering the constraint from the border of V).

The upper bound $dist$ is derived under the assumption that R_a^{dir} is not blocked by anybody else. In the latter half of the algorithm, we remove this assumption. Lines 15 and 16 compute the sets W_1 and W_2 of vertices that have the potential to block R_a^{dir} in $O(|V|)$ time; Lines 17 and 18 reduce W_1 and W_2 to W'_1 and W'_2 respectively, by removing the vertices that are not able to orthogonally block R_a^{dir} in $O(|V|^2)$ time. The computation in these four lines is similar to that in Lines 7-10. In the case that the union of W'_1 and W'_2 is not empty, the associated blocking point nearest to a in these two sets is identified in $O(|V|)$ time (Line 20). It is the true blocking point, *i.e.*, Besh point. The rationale is that R_a^{dir} can be blocked only at a single point, and after that no blocking is possible at any point further away from a . Thus the set of blocking vertices associated with this point are found and returned (Line 21). This final step takes another $O(|V|)$ time. In the case that the union of W'_1 and W'_2 is empty, the return value is single-element set $\{(c, d)\}$ if c exists (Line 23), and \emptyset otherwise (Line 25). The computation time is constant $O(1)$.

All the other lines in the algorithm take $O(1)$ time. In total $O(|V|^2)$ is the complexity of Algorithm ???. More efficient algorithms may be developed, but beyond the scope of this article. The correctness of Algorithm ?? simply follows from the above analysis. Then we may construct Besh within $O(|V|^3)$ time, by running this algorithm for every vertex in V four times, each time for a different direction in T . Although Besh is a transit product of our graph planarization process, it has its own importance in real life applications. In [?], we derived that Besh has good proximity property like Voronoi diagram through analytical study and simulation experiments, showed how to accomplish Besh in a localized way, without knowing V , and proposed a Besh-based localized distance-sensitive service discovery algorithm for wireless sensor and actor networks. Now we shall

move to the second drawing step. Before proceeding further, we would like to introduce a few important definitions and lemmas to be used in the sequel.

Definition 4 (Emptiness and Cleanness). *A region is empty if and only if there are no vertices located in it; a region is clean (with respect to Besh) if and only if it does not contain any Besh edge. A clean region must be empty, while the converse is obviously not necessarily true.*

Lemma 2. $\forall a, b \in V, a \neq b, dir \in T$ and $dir' \in \hat{dir}$, if $R_a^{dir} \not\leftrightarrow R_b^{dir'}$ then the region defined by triangle Δ_{aub} (including its perimeter) is empty.

Proof: Assume for the sake of contradiction that there are vertices inside Δ_{aub} . Among those vertices take the one, denoted as c , that is closest to $R_b^{dir'}$. There is must be a vertex d that blocks R_c^{dir} at v (i.e., $|dv| \leq |cv|$) before R_c^{dir} hits $R_b^{dir'}$ because otherwise, R_c^{dir} would block $R_b^{dir'}$ there and make $R_a^{dir} \not\leftrightarrow R_b^{dir'}$ impossible. This vertex d must not be in $Q_a(dir, dir')$ because otherwise, it would block R_a^{dir} and render $R_a^{dir} \not\leftrightarrow R_b^{dir'}$ impossible. Thus, d must be located outside Δ_{aub} and in $Q_c(dir, \overline{dir'})$. In this case, we however have $|dv| > |cv|$, contradicting that d blocks c at v . Hence, no vertices are located inside Δ_{aub} . By the blocking rule and under the constraint of $R_a^{dir} \not\leftrightarrow R_b^{dir'}$, it is obvious that no vertices can be located on the perimeter of Δ_{aub} either. \square

Lemma 3. $\forall a, b \in V$ and $a \neq b$, if they have a mutual blocking relation, then $\square ab$ is clean in $BS(V)$. Here $\square ab$ is the region defined by the rectangle (including its perimeter) parallel to the X and Y axes and with ab being diagonal line.

Proof: It follows immediately from Lemma ?? . \square

Lemma 4. $\forall a, b \in V$ and $R_a^{dir} \not\leftrightarrow R_b^{dir'}$ in $BS(V)$, if $R_b^{dir} \leftrightarrow R_a^{dir'}$ is not present in $BS(V)$ (namely, there is no mutual blocking between a and b), then

- 1) for the Besh cell $BC_w(\overline{dir}, \overline{dir'})$ cornered at w in $Q_w(\overline{dir}, \overline{dir'})$, there is exactly one diagonal definition point c with respect to w , and
- 2) c has a blocking relation with both a and b , and
- 3) $\max(mlen(\square ac), mlen(\square bc)) < mlen(\square ab)$, where $mlen(\cdot)$ is the length of the longest side of the box.

Proof: Among the four sides of $BC_w(\overline{dir}, \overline{dir'})$, the two that are joint at w are defined by R_a^{dir} and $R_b^{dir'}$. The definition of the other two sides cannot involve more than two additional creating points due to blocking. It cannot involve two additional creating points either, because otherwise, the two defining rays from these two points would cross over at the corner of $BC_w(\overline{dir}, \overline{dir'})$ diagonal to w , which we know is not possible because of blocking. Because R_b^{dir} does not block $R_a^{dir'}$, w then have exactly one diagonal definition point. We enumerate all the construction possibilities of $BC_w(\overline{dir}, \overline{dir'})$ by varying the relative location of c to a and b in Fig. ?? . Observe that c has a blocking relation with both a and b in any case. The blocking relations with a and b , together with the blocking relation between a and

b at w , constrain c to be located in the three shaded areas in Fig. ?? where $\angle bpa = \angle pbq = \angle sbp = \pi/4$, respectively called side zone, bottom zone and inner zone. We know c must not be located at p , and not at q either if $\square ab$ is a square, because otherwise, we would have $R_b^{dir} \not\leftrightarrow R_a^{dir'}$. Note that, depending on which of dir and dir' is a horizontal direction, c may not be located on line segment pr , br or bq in order to ensure the blocking relations. Recall $|bt| \leq |at|$. According to the blocking rule and by trivial comparison, we conclude $\max(mlen(\square ac), mlen(\square bc)) < mlen(\square ab)$. \square

3.2 The second step: 'Dual' of Besh

Having obtained $BS(V)$, we start the second step. At this step, we create the 'dual' of $BS(V)$ by adding edges between the creating points that have a blocking relation. Here term 'dual' is from an abused use of duality. It is of importance to remember that inter-vertex blocking relation is subject to the prioritized blocking policy. Formally, we define:

Definition 5 (HC edge creation rule). $\forall a, b \in V$ and $a \neq b$, create edge ab if and only if a and b have a blocking relation.

The dual of Besh $BS(V)$ is composed of the given vertex set V and the added edge set. We name it *Hypocomb* (standing for *Hypotenuse-comb*) and denote it by $HC(V)$. The name 'Hypocomb' owns its inspiration to the fact that each edge ab due to $a \not\leftrightarrow b$ is the hypotenuse of the right triangle Δ_{aub} . In Fig. ??, Hypocomb edges are drawn in thick links. By the HC edge creation rule and using Algorithm ??, we can trivially build Hypocomb in $O(|V|^3)$ time. Below we analyze the connectivity, planarity and degree bound of Hypocomb.

Theorem 1. $HC(V)$ is connected.

Proof: For all $a, b \in V$ and $dir, dir' \in T$, if $R_b^{dir'} \leftrightarrow R_a^{dir}$ or $R_b^{dir} \leftrightarrow R_a^{dir'}$, then we say R_b^{dir} is an extension of R_a^{dir} . A ray has at most 2 extensions. In Fig. ??, R_d^{south} is extended by both R_c^{south} and R_b^{south} , for example. Ray extension occurs from a toward dir in a cascaded fashion until a vertex, called *terminal node*, whose ray growing in direction dir is not blocked (by any other vertex) is reached. Cascaded ray extension defines a directed acyclic graph $DAG(a, dir)$, where nodes are the vertices involved and edges imply direct ray extension relation. Define $DAG(a, dir, \overline{dir}) = DAG(a, dir) \cup DAG(a, \overline{dir})$. It spans the space enclosed by the border of V . Because direct ray extension implies blocking relation, each edge in $DAG(a, dir, \overline{dir})$ corresponds to an edge with the same end nodes in $HC(V)$. As such, this DAG is mapped to a subgraph of $HC(V)$, denoted by $M_{DAG}(a, dir, \overline{dir})$, which is connected due to the reachability from a to every other node in $DAG(a, dir, \overline{dir})$. For all $a' \in V$, $a' \neq a$ and $dir' \in \hat{dir}$, $DAG(a', dir', \overline{dir'})$ must have some node(s) in common with $DAG(a, dir, \overline{dir})$. It is due to the spanning property and perpendicularity of the two DAGs. As a consequence, $M_{DAG}(a, dir, \overline{dir})$

Fig. 3. Theorem ??

and $M_{DAG}(a', \overline{dir'}, \overline{dir'})$ are connected. By definition, $HC(V) = \bigcup_{a \in V} (M_{DAG}(a, \overline{dir}, \overline{dir}) \cup M_{DAG}(a, \overline{dir'}, \overline{dir'}))$. The connectivity of $HC(V)$ therefore follows. \square

Theorem 2. $HC(V)$ is planar.

Proof: Assume for the sake of contradiction $ab, cd \in HC(V)$ and they intersect. Let u be a blocking point of a and b , and let v be a blocking point of c and d . Consider the two triangles Δ_{aub} and Δ_{cud} . By Lemma ??, they are both empty, that is to say, $a, b \notin \Delta_{cud}$ and $c, d \notin \Delta_{aub}$. Then the two triangles must intersect, with their hypotenuses being across. In this case, one of the catheti of Δ_{aub} , say au , intersects with one of the catheti, say cv , of Δ_{cud} . Let the crossover point be w . A blocking relation between a and c occurs at w . This renders either the blocking of a and b at u or that of c and d at v invalid. A contradiction is reached. \square

Theorem 3. $\Delta(HC(V)) \leq |V| - 1$.

Proof: It is obvious that $\Delta(HC(V))$ cannot be larger than $|V| - 1$ which is the degree of the complete graph of V . We just need to show that it is possible to have $\Delta(HC(V)) = |V| - 1$. Examine a particular vertex arrangement given in Fig. ??, where $|au| = |cu|$. Any vertex on the line segment bc will be blocked by a , and thus has an incidental edge with a in the corresponding Hypocomb. If all the other vertices in V are located on bc , vertex a will have degree exactly equal to $n - 1$. \square

4 REDUCED HYPOCOMB

In the previous section we presented a novel planar graph, Hypocomb, which is extracted from a complete graph and has unbounded degree. In this section we simplify Hypocomb, reducing the number of edges, by applying a constrained edge creation rule (see Definition ??) at the second planarization step. We refer to the resulting simplified Hypocomb as *Reduced Hypocomb* and denote it by $RHC(V)$.

Definition 6 (RHC edge creation rule). $\forall a, b \in V$ and $a \neq b$, create edge ab if and only if a and b have a mutual blocking relation.

Corollary 1. $RHC(V) \subseteq HC(V)$.

In Fig. ??, only solid thick lines belong to Reduced Hypocomb. Corollary ?? is derived immediately from Definition ?. With Algorithm ??, Reduced Hypocomb construction is straightforward and has the same complexity $O(|V|^3)$ as Hypocomb construction. Below, we show that Reduced Hypocomb not only remains connected planar but also possesses the desired bounded-degree property.

Theorem 4. $RHC(V)$ is connected.

Proof: Since $RHC(V)$ is a subgraph of $HC(V)$, the construction of $RHC(V)$ can be viewed an edge removal process in $HC(V)$, where we remove non-RHC edges one by one. Consider an arbitrary non-RHC edge $ab \in HC(V)$. By definition, a and b have no mutual blocking relation. Without loss of generality, let the inclusion of ab in $HC(V)$ be due to $R_a^{dir} \stackrel{w}{\not\leftrightarrow} R_b^{dir'}$ with $dir \in T$ and $dir' \in \overline{dir}$. By Lemma ??, we have $ac, bc \in HC(V)$ where c is the unique diagonal definition point of the Besh cell cornered at w and located in $Q_w(\overline{dir}, \overline{dir'})$ with respect to w . If we remove ab and only ab from $HC(V)$, a and b remain connected via c . We call such an edge removal action ‘connectivity division’ and call ac and bc the results of division of ab by c . Because ac and bc can also be removed, connectivity division does not preserve connectivity unless no division loop is induced. Below we prove that indeed no division loop occurs.

Assume for the sake of contradiction that there are division loops. Take a smallest loop where each edge appears only once. We express this loop by $u_0v_0 \xrightarrow{w_1} u_1v_1 \xrightarrow{w_2} u_2v_2 \xrightarrow{w_3} \dots \xrightarrow{w_n} u_0v_0$. Let $u_nv_n = u_0v_0$. For $i = 1, \dots, n$, $u_{i-1}v_{i-1} \xrightarrow{w_i} u_iv_i$ indicates that u_iv_i is a result of the connectivity division of $u_{i-1}v_{i-1}$ by w_i , where $u_i \in \{u_{i-1}, v_{i-1}\}$ and $v_i = w_i$. We know u_{i-1} and v_{i-1} have no mutual blocking relation and $R_{u_{i-1}}^{dir_{i-1}} \stackrel{w_i}{\not\leftrightarrow} R_{v_{i-1}}^{dir'_{i-1}}$ (or $R_{v_{i-1}}^{dir'_{i-1}} \stackrel{w_i}{\not\leftrightarrow} R_{u_{i-1}}^{dir_{i-1}}$) for $dir_{i-1} \in T$ and $dir'_{i-1} \in \overline{dir}_{i-1}$. Recall that $mLen(\square_{u_{i-1}v_{i-1}})$ is the maximum side length of $\square_{u_{i-1}v_{i-1}}$. In this case, applying lemma ?? along the loop, we have $mLen(\square_{u_0v_0}) > mLen(\square_{u_1v_1}) > \dots > mLen(\square_{u_{n-1}v_{n-1}}) > mLen(\square_{u_0v_0})$, which is impossible. This completes the proof. \square

Theorem 5. $RHC(V)$ is planar.

Proof: It follows from Theorem ?? and Corollary ?. \square

Lemma 5. $\forall ab, ac \in RHC(V)$, creation of ab is due to $R_a^{dir} \stackrel{u}{\not\leftrightarrow} R_b^{dir}$ and ac is due to $R_a^{dir} \stackrel{w}{\not\leftrightarrow} R_c^{dir'}$ with $dir \in T$ and $dir' \in \overline{dir}$, $\nexists ad \in RHC(V)$ such that $ad \in Q_a(dir, dir')$.

Proof: Assume for the sake of contradiction that such ad exists. By Definition ??, a and d have a mutual blocking relation. Without loss of generality, let $R_a^{dir} \stackrel{u}{\not\leftrightarrow} R_d^{dir'}$ and symmetrically $R_d^{dir} \stackrel{w}{\not\leftrightarrow} R_a^{dir'}$ for some u and w . However $R_d^{dir} \stackrel{w}{\not\leftrightarrow} R_a^{dir'}$ and $R_a^{dir} \stackrel{t}{\not\leftrightarrow} R_c^{dir'}$ obviously

cannot hold at the same time (even if $w = t$, by Definition ??). \square

Theorem 6. $\Delta(RHC(V)) \leq 6$.

Proof: $\forall a \in V$, there are at most 4 clean $\square ab$ in $BS(V)$ in the four quadrants of a (one in each quadrant), and at most 4 clean $\square ab$ (which reduces to ab) along the X and the Y axis respectively in the four directions. Hence a has at most 8 incidental edges in $RHC(V)$, 4 quadrant edges and 4 axis edges. By Lemma ??, two axis edges must be either separated by more than one quadrant edge or adjacent to each other. This constraint then lowers the upper bound to 6. The scenario of degree 6 is that a has 4 quadrant edges and 2 collinear axis edges either along the X axis or the Y axis. \square

5 LOCAL HYPOCOMB

Till now we have bounded the degree of Hypocomb successfully above by a small constant 6, by applying constrained edge creation rule and yet without jeopardizing its connectivity and planarity properties. Hypocomb and Reduced Hypocomb are built with complete knowledge of V and with no constraint on edge length, *i.e.*, extracted from a complete graph of V . In this section we investigate how to build Reduced Hypocomb on UDG with limited local knowledge. UDG has the following important property (proof is given in Appendix ??).

Lemma 6. *In UDG, if two edges intersect, then one end vertex of one edge neighbors the two end vertices of the other.*

Specifically, $\forall a \in V$, when we draw incidental edges for it, we merely have the position information of vertices b located in the unit disk $D_\gamma(a)$ of a . In this case, we propose a local edge creation rule (see Definition ??), which adds ab according to its inclusion in the Reduced Hypocomb graph of the *closed common neighbor set* of a and b . And obviously, the creation decision on edge ab is symmetric for a and b .

Definition 7 (LHC edge creation rule). $\forall a \in V$ and $b \in V_{nbr}(a)$ and $a \neq b$, create edge ab if and only if $ab \in RHC(V_{nbr}(a, b))$.

This local edge creation rule is dependent on 1-hop neighbors position information only. It gives the graph construction process strictly localized feature. The resulting Hypocomb variant is therefore called *Local Hypocomb*, and denoted by $LHC(V)$. We know $|V_{nbr}(a, b)| \leq d(a) + 1$, where $d(a)$ is the degree of a in UDG. Using Algorithm ?? each node a is able to build $RHC(V_{nbr}(a, b))$ for each neighbor b in $O(d(a)^3)$ time, and the total cost of determining LHC edges is therefore $O(d(a)^4)$. Because $d(a) \leq \Delta(UDG(V))$, an upper bound of the computation cost on each node for Local Hypocomb construction is $O(\Delta(UDG(V))^4)$. Below we show that Local Hypocomb surprisingly remains connected planar and has degree bounded above by 8 (just slightly larger than the degree bound 6 of Reduced Hypocomb).

(a) Case 3.1

(b) Case 3.2

Fig. 4. Lemma ??

Theorem 7. $LHC(V)$ is connected.

Proof: We view $LHC(V)$ construction as an edge removal process in $UDG(V)$. For every $ab \in UDG(V)$ and $ab \notin LHC(V)$ (*i.e.*, a removed edge), by definition we have $ab \notin RHC(V_{nbr}(a, b))$. This implies either $ab \notin HC(V_{nbr}(a, b))$ or, $ab \in HC(V_{nbr}(a, b))$ and there exists a unique $c \in V_{nbr}(a, b)$ such that it divides ab into $ac, bc \in HC(V_{nbr}(a, b))$ (Lemma ??). In the former case, the removal of ab does not affect the connectivity between a and b since we know $HC(V_{nbr}(a, b))$ is connected (Theorem ??). In the latter case, a and b remain connected (through c) from the local view of a and b after removing ab . To prove the connectivity of $LHC(V)$, it is sufficient to prove that local connectivity division (edge removal) actions do not generate division loop in a global sense. The loop-free property can be proved similarly as in Theorem ??). The key is to explore the stability of $mLen(\square ab)$ (*i.e.*, it is the same in any vertex's local view) and the monotonically decreasing nature of $mLen(\square ac)$ and $mLen(\square bc)$ relative to $mLen(\square ab)$. \square

Lemma 7. *Any two crossover edges $ab, cd \in UDG(V)$ do not belong to $LHC(V)$ at the same time if $ac, ad \in UDG(V)$ and one of bc and bd appears in $UDG(V)$.*

Proof: Assume for the sake of contradiction $ab, cd \in LHC(V)$. Without loss of generality, let $bc \in UDG(V)$. By Definition ??, $ab \in RHC(V_{nbr}(a, b))$. From Lemma ??, $\square ab$ is clean in $BS(V_{nbr}(a, b))$ and thus $c \notin \square ab$; likewise, $\square cd$ is clean in $BS(V_{nbr}(c, d))$ and $a \notin \square cd$. Under these constraints, we vary the relative position of b and d to $\square cd$ and $\square ab$ and obtain the following cases: (1) $b \in \square cd$ and $d \notin \square ab$; (2) $d \in \square ab$ and $b \notin \square cd$; (3) $b \notin \square cd$ and $d \notin \square ab$. Note that $b \in \square cd$ and $d \in \square ab$ cannot hold at the same time and thus . In case (1), $|bd| \leq |cd|$ because $\angle cbd$ is not acute in $\triangle cbd$. In case (2), $|bd| \leq |ab|$ because $\angle adb$ is not acute in $\triangle adb$. In the two cases, $d \in V_{nbr}(a, b)$ and $b \in V_{nbr}(c, d)$, and obviously ab and cd do not appear in $LHC(V)$ at the same time, contradicting our assumption. Case (3) has two sub-cases. Below, we derive a contradiction from both of them.

Case 3.1 (Fig. ??) : We first put ourselves under the condition: a would block d at u if V contained only a, d .

We have $|au| \leq |du|$ by the blocking rule. Because $a \in V_{nbr}(c, d)$ and $cd \in RHC(V_{nbr}(c, d))$, R_a^{dir} must be blocked by a vertex e_0 in $BS(V_{nbr}(c, d))$ before reaching segment dp . If e_0 has a projection e'_0 on dp , then by Lemma ?? we have $|e_0e'_0| \leq |de'_0|$ and are facing the same situation as with a , and therefore the same argument can be made for e_0 . By these means, we are presented a blocking chain in $BS(V_{nbr}(c, d))$ that ends at a vertex e_n ($n \geq 0$) that has no projection on dp . Let e''_i be the projection of e_i on R_a^{dir} . By the blocking rule and Lemma ??, we easily have $|e_i e''_i| \leq |ae''_i| < |au| < |du|$ for $i = 0, \dots, n$. Since $|du| > |au|$, e_n cannot be around vertex d but point p . For ease of presentation, let $e = e_n$ and $e'' = e''_n$, as shown in the figure. Notice $|cr| = |pu| \leq |ee''| \leq |ae''| < |au| < |ar|$. This implies that c is in the same situation with respect to a as a with respect to d . By the same argument, we conclude that there exists such a vertex f around w for c (like e for a).

By simple geometry, the four vertices a, b, c and d are all neighboring e if $|su| \leq |sv|$, and f otherwise. Without loss of generality, we consider $|su| \leq |sv|$ since the other case is symmetric. In $BS(V_{nbr}(a, b))$, R_e^{dir} must be blocked by a vertex g_0 at a point x on segment ee'' . According to the blocking rule, g_0 must be located in a square area (shaded in the figure) with e as corner and with the diagonal line defined by two other corners lying on segment aw . Trivially, all points in the square are common neighbor of a, b, c and d . Since in $BS(V_{nbr}(c, d))$, g_0 does not block R_e^{dir} , there must be a vertex g_1 that blocks g_0 at a point y on segment g_0x . This vertex g_1 is again located in the square area and thus neighboring a, b, c and d . The argument can be made iteratively, alternate between $BS(V_{nbr}(a, b))$ and $BS(V_{nbr}(c, d))$, giving us a set of vertices g_0, \dots, g_m all located in the square area and neighboring a, b, c and d . Hence the blocking relations among them appear in both $BS(V_{nbr}(a, b))$ and $BS(V_{nbr}(c, d))$, and ab and cd cannot appear in $LHC(V)$ at the same time, contradicting our assumption.

We now consider the opposite condition: d would block a if a and b were the only vertices in V . We have $|au| \geq |du|$ ($\geq |ds|$) and $|bd| < |ds| + |bs| \leq |au| + |bs| = |bt| \leq |ab|$, i.e., $d \in V_{nbr}(a, b)$. Then the only situation worth investigation includes the combination of the following conditions: b would block d at s if V contained only b, d ($|ds| \geq |bs| \geq |bv|$), c would block b at v if V contained only b, c ($|bv| \geq |cv| \geq |cr|$), and a would block c at r if V contained only a, c ($|cr| \geq |ar| \geq |au|$). It is because any other situation is equivalent to the previous one after vertex renaming and therefore leads to a similar contradiction. Under this circumstance, we have $|au| = |ds| = |bv| = |cr| = |au|$ and $|su| = |sv| = 0$. Then either a and b or, c and d , do not have blocking relation according to the prioritized blocking policy, a contradiction.

Case 3.2 (Fig. ??): As in case (3.1), we first investigate under the condition that a would block d at u if V contained only a, d . Then with respect to a , we may conclude a

(a) Lemma ?? and ?? (b) Lemmas ?? and ??

Fig. 5. Illustrations for Local Hypocomb analysis

similar set of vertices $e_0, \dots, e_n \in V_{nbr}(c, d)$ in order to enable $cd \in RHC(V_{nbr}(c, d))$. Among them, e_n is around p . Let e be the one closest to segment bt and on the same side as p and e'' the projection of e on R_a^{dir} . In right triangle $\Delta ee''a$, $|ae|^2 = |ae''|^2 + |ee''|^2 < |au|^2 + |du|^2 = |ad|^2$. Let j be the intersection point of ae and bt and k the intersection point of ee'' and bt . Trivially, $|ek| \leq |kj|$. In right triangle Δekb , $|be|^2 = |ek|^2 + |kb|^2 \leq |kj|^2 + |kb|^2 \leq (|tj| + |kj| + |kb|)^2 + |at|^2 = |ab|^2$. Hence $e \in V_{nbr}(a, b)$. By the same technique we may derive a similar contradiction as in case (3.1).

Likewise, we can derive a contradiction under the condition that c would block b at v if V contained only b, c . We only remain to consider the combination of the opposites of the two conditions, where $|au| \geq |du|$ and $|cv| \geq |bv|$. Observe $|bt| = |au| + |sv| + |bv|$, $|dp| = |cv| + |at| + |du|$, $|ds| = |at| + |du|$ and $|cp| = |sv|$. Let $\delta_1 = |bd|^2 - |ab|^2$ and $\delta_2 = |bd|^2 - |cd|^2$. In right triangle Δcpd , $|cd|^2 = |dp|^2 + |cp|^2 = (|cv| + |at| + |du|)^2 + |sv|^2$. In right triangle Δbsd , $|bd|^2 = |bs|^2 + |ds|^2 = (|bv| + |sv|)^2 + (|at| + |du|)^2$. In right triangle Δatb , $|ab|^2 = |at|^2 + |bt|^2 = |at|^2 + (|au| + |sv| + |bv|)^2$. Then $\delta_1 = |du|^2 - |au|^2 + 2|du||at| - 2|au|(|sv| + |bv|)$ and $\delta_2 = |bv|^2 - |cv|^2 + 2|bv||sv| - 2|cv|(|at| + |du|)$. Recall $|du| \leq |au|$ and $|bv| \leq |cv|$. If $|at| \leq |sv| + |bv|$, then $\delta_1 \leq 0$ (i.e., $|bd| \leq |ab|$); otherwise, $|sv| < |at| - |bv| < |at| + |du|$ and thus $\delta_2 \leq 0$ (i.e., $|bd| \leq |cd|$). This implies $bd \in UDG(V)$. Thus current situation is equivalent to the first situation examined (after switching the name of a and d and other vertex remaining), and we may derive a contradiction similarly. \square

Lemma 8. Any two crossover edges $ab, cd \in UDG(V)$ do not belong to $LHC(V)$ at the same time if $ac, ad \in UDG(V)$ and $bc, bd \notin UDG(V)$.

Proof: Clearly, cd must intersect the unit circle $C_\gamma(b)$ of b as, otherwise, $ab \notin UDG(V)$. Let c' and d' be intersection points of cd and $C_\gamma(b)$. $|cd| > |c'd'|$. Then c' and d' must be on arc st of $\pi/3$ of $C_\gamma(b)$, with chord st parallel to cd , as shown in Fig. ???. It is because, otherwise, $|cd| > \gamma$ (given cd intersects ab) cannot belong to $UDG(V)$. In this case and being with the constraint $ab \in UDG(V)$, a must be located in the arc segment area

defined by c' and d' . And, it must be located outside $\square cd$ so that $cd \in RHC(V_{nbr}(c, d))$. This additional restriction limits the location of a to be within the arc segment defined by the intersection points p and q of $\square cd$ and arc $c'd'$. In Fig. ??, $\square cd$ is shown by a dotted rectangle. The tangent of $C_\gamma(b)$ at s has a $\pi/6$ angle with st . The angle of the tangent at p therefore has an angle less than $\pi/6$ with pq . We have $\angle acq < \angle apq < \pi/6 < \pi/4$. Recall $a \in V_{nbr}(c, d)$. In right triangle $\Delta ca'a$, where a' is the projection of a on cq , $|aa'| < |ca'|$. This implies $R_a^{dir} \not\leftrightarrow R_c^{dir'}$ if no other vertex blocks R_a^{dir} before it reaches $R_c^{dir'}$. It is possible that some vertex $m \in V_{nbr}(c, d)$ blocks a such that this blocking relation does not exist. However, in this case, m will block $R_c^{dir'}$ if no other vertex blocks m by Lemma ?. The same argument can be made iteratively. Since we have a finite number of vertices in $V_{nbr}(c, d)$, finally a vertex will block $R_c^{dir'}$. And obviously this vertex must be located in either of the two squares with aa' as common edge. These two squares are between p and q due to the fact that $\angle apq < \pi/4$ and $\angle pqa < \pi/4$. Thus the mutual blocking relation of c and d is broken. This finally contradicts $cd \in RHC(V_{nbr}(c, d))$. \square

Theorem 8. $LHC(V)$ is planar.

Proof: Any edge in $LHC(V)$ is also in $UDG(V)$. For any pair of crossover edges ab and cd in $UDG(V)$, without loss of generality, let $ac, ad \in UDG(V)$ by Lemma ?. Then regardless the containment relations of bc and bd in $UDG(V)$, ab and cd do not appear in $LHC(V)$ at the same time according to Lemma ? and ?. Thus the theorem holds. \square

Lemma 9. $\forall dir \in T, dir' \in \hat{dir}, ab, ac \in LHC(V), ab, ac \in Q_a(dir, dir')$ and $ab \neq ac, \nexists ad \in LHC(V)$ and $ad \neq ab, ac$ such that $ad \in Q_a(dir, dir')$.

Proof: We first derive $|bc| > \gamma$, where γ is unit distance. It is because, otherwise, b and c would be in each other's closed common neighborhood with a , and in this case, by Lemma ?? at most one of ab and ac would belong to $LHC(V)$. Then c must be located in the differential area of $D_\gamma(a)$ and the $D_\gamma(b)$ in $Q(dir, dir')$. Let s and t respectively be the intersection point of R_a^{dir} and $R_a^{dir'}$ with $C_\gamma(a)$, as shown in Fig. ?. For such a residence area of c exists, b must be in one of the shaded areas, which are defined by $C_\gamma(s)$ and $C_\gamma(t)$. Symmetrically, c must be in the other shaded area. For the sake of contradiction, assume $\exists ad \in LHC(V)$, $ad \neq ab, ac$, and $ad \in Q_a(dir, dir')$. Then $b \notin D_\gamma(d)$ and $c \notin D_\gamma(d)$. That is, vertex d must be located in the intersection area of the two shaded areas, which however does not exist. \square

Lemma 10. $\forall ad \in LHC(V)$, if the creation of ad is due to $R_a^{dir} \xrightarrow{u} R_d^{dir'}$ with $dir \in T$, then $\nexists ab, ac \in LHC(V)$ and $ab \neq ac$ such that $\exists dir' \in \hat{dir}, ab, ac \in Q_a(dir, dir')$.

Proof: Assume for the sake of contradiction that

(a)(b)
Ma-
Av-
i-
er-
mu-
age
de-
de-
gree

Fig. 6. Numerical results

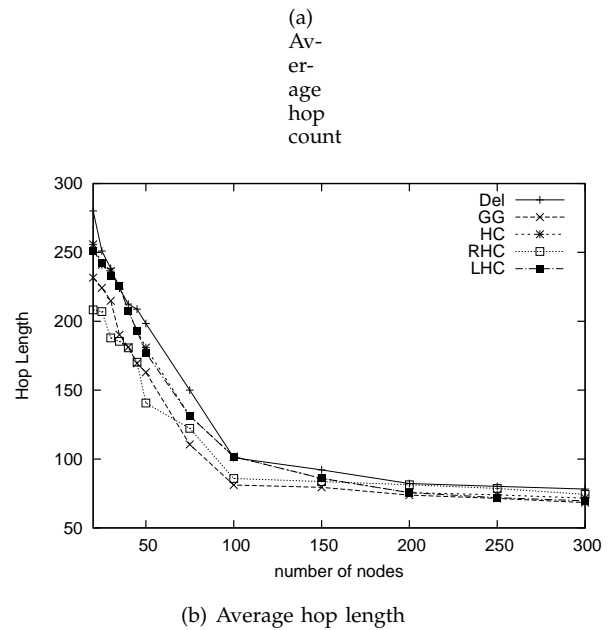


Fig. 7. Simulation results

such ab and ac exist. Observe Fig. ??, which depicts $Q_a(dir, dir')$ in a generic way. Vertices b and c must be located separately in the two shaded areas, as we analyzed in the proof of Lemma ?. Without loss of generality, let b be in zone 1 and c in zone 2. Obviously, $\angle uab < \pi/4$. To ensure the blocking relation $R_a^{dir} \xrightarrow{u} R_d^{dir'}$, we must have $R_a^{dir} \xrightarrow{u'} R_b^{dir'}$ with $u' \in au$. This implies $|bu'| \geq |au'|$ and thus $\angle uab \geq \pi/4$. A contradiction is reached. Hence the lemma holds. \square

Theorem 9. $\Delta(LHC(V)) \leq 8$

Proof: It follows immediately from Lemmas ?? and ?. Lemma ?? indicates that in LHC each node has at most 2 edges in each quadrant; Lemma ?? indicates that in LHC, if a node has an axis edge, then it has at most 1 edge in each of the 2 quadrants adjacent to that edge. Thus the scenario of degree 8 is that a vertex has 2 edges in each quadrant or that it has 4 axis edges and 4 quadrant edges, one in each quadrant. \square

6 NUMERICAL RESULTS

We now study the average degree of Hypocomb family graphs and their impact on FACE routing [?], in comparison with Del, LDel and GG through extensive simulation. We run simulation experiments using a custom C simulator to build these graphs over the same

random node (*i.e.*, vertices) distribution. To do so, we compute a virtual square grid and place n nodes at n randomly selected distinct grid points. For GG, LDel and LHC computation, a UDG is generated with a properly selected unit distance to ensure connectivity. An example construction of these graphs when $n = 40$ can be found in Appendix ???. We run FACE over each graph for a randomly picked pair of source and destination. Indeed, FACE has to be run on a planar graph only, and it was supported by GG in [?]. Below we report our simulation results, which are obtained from 1000 simulation runs with n being varied from 20 to 300.

Figure ?? verifies our theoretical findings about degree bound: HC has unbounded degree while the degree of RHC and LHC is bounded above by 6 and 8, respectively. Figure ?? shows the average degree (reflecting how sparse or dense a graph is), which as expected slowly increases with the overall number n of nodes. For RHC and LHC, it never exceeds the corresponding degree bound. We observe that their curves become flat after a turnover point of $n = 200, 250$ respectively. Del, HC and RHC are extracted from complete graph and therefore comparable to each other. Among them Del and RHC are respectively the densest and the sparsest. LDel, GG and LHC are both local graphs and thus competitors. LHC is a bit denser than GG for before $n = 250$ and becomes increasingly sparser afterwards as GG has no degree bound. On average, LDel is the densest before $n = 200$ and then the sparsest on average.

Generally speaking, the denser a graph, the higher the degree of the graph, and the smaller the average face size in the graph. Hence, as n goes up, FACE is expected to produce increasingly long paths, in terms of hop count, which are composed of shorter and shorter hops. This expectation is confirmed by the ascending trend of the curves in Fig. ?? and the descending trend of the curves in Fig. ??.

Observe Fig. ?. HC and RHC both have larger hop count than Del in all tested graphs, and RHC always has larger hop count than HC. When $n < 200$, LHC has smaller hop count than GG and larger hop count than LDel; when $n > 200$, it shows larger hop count than GG and smaller hop count than LDel (because the average degree of LDel is smaller than that of LHC). Examine Fig. ?. When $n < 200$, HC is comparable to Del (with slight difference), and RHC has shorter hop length than Del. When $n \geq 200$, HC and RHC both show slightly longer hop length than Del. RHC always has shorter hop length than HC. Compared with GG, LHC has longer hop length for $n < 200$ and shorter hop length for $n \geq 200$. It exhibits shorter hop length than LDel almost all the time, with noticeable difference.

7 CONCLUSIONS

In this article we have proposed Hypocomb family graphs. We proved that they are connected planar. We also investigated their degree bound. These graphs are

radically novel. They have no similarity with any existing geometric planar graph. The work opened a new direction of research. Many follow-up works are possible. In Appendix ??, we show through a counter example that Hypocomb family graphs may not contain MST as subgraph. It is therefore interesting to study whether or not Hypocomb family graphs are *low-weight* graphs. A structure is called low-weight if its total edge length is within a constant factor of the total edge length of the MST [?]. In Appendix ??, we show the numerical results of the spanning ratio of Hypocomb family graphs in comparison with other planar graphs. It is an open problem whether Hypocomb family graphs are spanner. Another research topic is to develop graph construction algorithms more efficient than Algorithm ??.

There are also open practical problems brought by Hypocomb family graphs. In [?], we presented a localized method to construct Besh using actor nodes as creating points in wireless sensor and actor networks. Sensors constitute a connected network; actors are generally multi-hop away from each other. In the resulting Besh, edges are multi-hop routing paths composed of sensors. The idea is to simulate the ray drawing process from each actor by directional message transmission, which is in turn realized by Greedy-FACE-Greedy (GFG) routing [?]. We may construct Hypocomb and Reduced Hypocomb with minor modification to this algorithm as follows: each node where blocking happens informs the sender actors about the blocking so that the latter know about who they are blocking and whom they are blocked by. The goal is to obtain an overlay network bearing planar topology among actors so that existing communication protocols can be run directly, or with minimum modification, on it to realize, for example, actor-to-actor broadcasting, any-casting and multi-casting and sensor-to-actor broadcasting and any-casting, etc., which are the basic operations for actor-actor and sensor-actor coordination.

Indeed, Hypocomb and Reduced Hypocomb are promising for the emerging field of wireless sensor and actor networking. However, the above stated construction method does not produce exactly these graphs due to generally non-straight-line message transmission and thus inaccurate blocking relation, unless the underlying network has a grid topology. A future research direction is thus to study and improve the performance of this construction method and eventually develop new and better distributed/localized solutions. Both Reduced Hypocomb and Local Hypocomb may be used for blue-tooth scatternet formation. When Local Hypocomb is applied, in each comprising piconet there is at most one parked node. As revealed by our simulation study, Local Hypocomb is promising for localized greedy-face combined ad hoc routing [?]. Their impact, when used, on the networking process and applications (*e.g.*, data centric storage) warrants deep investigation as well.

ACKNOWLEDGEMENTS

This work was partially supported by the NSERC Post-doctoral Fellowships Program.

REFERENCES

- [1] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, "Routing with guaranteed delivery in ad hoc wireless networks." *Wireless Networks*, vol. 7, no. 6, pp. 609-616, 2001.
- [2] P. Bose, L. Devroye, W. Evans, and D. Kirkpatrick, "On the spanning ratio of gabriel graphs and beta-skeletons." *Proc. LATIN*, 2002.
- [3] H. Frey and I. Stojmenovic, "On Delivery Guarantees of Face and Combined Greedy-Face Routing Algorithms in Ad Hoc and Sensor Networks." *Proc. ACM MobiCom*, pp. 390-401, 2006.
- [4] J. Gao, L.J. Guibas, J. Hershberger, L. Zhang, and A. Zhu, "Geometric spanner for routing in mobile networks." *Proc. ACM MobiHoc*, pp. 45-55, 2001.
- [5] F. Ingelrest, N. Mitton, and D. Simplot-Ryl, "A Turnover based Adaptive HELLO Protocol for Mobile Ad Hoc and Sensor Networks." *Proc. IEEE MASCOTS*, pp. 9-14, 2007.
- [6] J.W. Jaromczyk and G.T. Toussaint, "Relative neighborhood graphs and their relatives." *Proc. the IEEE*, vol. 80, no. 9, pp. 1502-1517, 1992.
- [7] J.M. Keil and C.A. Gutwin, "Classes of graphs which approximate the complete euclidean graph." *Discrete and Computational Geometry*, vol. 7, pp. 13-28, 1992.
- [8] D.G. Kirkpatrick and J.D. Radke, "A framework for computational morphology." *Computational Geometry*, Elsevier, Amsterdam, pp. 217-248, 1985.
- [9] N. Li, J.C. Hou, and L. Sha, "Design and Analysis of a MST-Based Topology Control Algorithm." *Proc. IEEE INFOCOM*, 2003.
- [10] X. Li, N. Santoro, and I. Stojmenovic, "Localized Distance-Sensitive Service Discovery in Wireless Sensor and Actor Networks." *IEEE Trans. Computers*, vol. 58, no. 9, pp. 1275-1288, 2009.
- [11] X. Li, N. Mitton, I. Simplot-Ryl, and D. Simplot-Ryl, "A Novel Family of Geometric Planar Graphs for Wireless Ad Hoc Networks." *Proc. IEEE INFOCOM*, pp. 1934-1942, 2011.
- [12] X.-Y. Li, "Approximate MST for UDG Locally." *Proc. COCOON*, LNCS 2697, pp. 364-373, 2003.
- [13] X.-Y. Li, G. Calinescu, P. Wan, and Yu Wang, "Localized Delaunay Triangulation with Application in Ad Hoc Wireless Networks." *IEEE Trans. Parallel and Distributed Systems*, vol. 14, no. 10, pp. 035-1047, 2003.
- [14] X.-Y. Li, Y. Wang, P.-J. Wang, W.-Z. Song, and O. Frieder, "Localized low weight graph and its applications in wireless ad hoc networks." *Proc. IEEE INFOCOM*, 2004.
- [15] X.-Y. Li, Y. Wang, and W. Song, "Applications of k -Local MST for Topology Control and Broadcasting in Wireless Ad Hoc Networks." *IEEE Trans. Parallel and Distributed Systems*, vol. 15, no. 12, pp. 1057-1069, 2004.
- [16] X.-Y. Li, I. Stojmenovic, and Y. Wang, "Partial Delaunay Triangulation and Degree Limited Localized Bluetooth Scatternet Formation." *IEEE Trans. Parallel and Distributed Systems*, vol. 15, no. 4, pp. 350-361, 2004.
- [17] C. Monma and S. Suri, "Transitions in geometric minimum spanning trees." *Proc. ACM Symposium on Computational Geometry*, pp. 239-249, 1991.
- [18] Y. Wang and X.-Y. Li, "Localized Construction of Bounded Degree Planar Spanner for Wireless Networks." *Mobile Networks and Applications*, vol. 11, pp. 161-175, 2006.
- [19] A.C.C. Yao, "On constructing minimum spanning trees in k -dimensional spaces and related problems." *SIAM Journal on Computing*, vol. 11, pp. 721-736, 1982.

APPENDIX A PROOF OF LEMMA ??

Proof: We prove this lemma by case study. For all $ab, cd \in UDG(V)$, $ab \neq cd$, and ab and cd intersect, there are four combinations of the containment relations of edges bc and bd to $UDG(V)$: (1) $bc \in UDG(V)$ and

Fig. 8. Lemma ?? – Case (2)

$bd \in UDG(V)$; (2) $bc \in UDG(V)$ and $bd \notin UDG(V)$; (3) $bc \notin UDG(V)$ and $bd \in UDG(V)$; (4) $bc \notin UDG(V)$ and $bd \notin UDG(V)$. Case (1) is the desired result. Case (3) is equivalent to case (2) after vertex renaming. Case (2) has two sub-cases: $ac \in UDG(V)$ and $ac \notin UDG(V)$. The former is equivalent to case (1) after vertex renaming. We only need consider the latter case and case (4).

We first prove that case (2) with the additional constraint of $ac \notin UDG(V)$ is not possible by contradiction. Assume for the sake of contradiction that it is possible, as depicted in Fig. ?. In Δacd , $|ac| > \gamma \geq |cd|$, and therefore $\angle cda > \angle dac$. Because $\angle cda$ is contained in $\angle bda$ and $\angle dac$ contains $\angle dab$, we have $\angle bda > \angle dac$, which in turn implies $|bd| < |ab| < \gamma$. A contradiction is reached.

Now we prove $ac, ad \in UDG(V)$ in case (4) (which makes it equivalent to case (1)). Let c' and d' be the intersection points of cd and $C_\gamma(b)$, as shown in Fig. ?. Because $|c'd'| < |cd| \leq \gamma$, we have $\angle c'bd' < \pi/3$. In order for ab to intersect cd , a must be in the arc segment of $\angle c'bd'$, and therefore $\angle d'ac' > \pi/2$. Because $\angle dac > \angle d'ac'$, we know cd is the longest side of Δcad with $|cd| \leq \gamma$ and consequently $cb, bd \in UDG(V)$. \square

APPENDIX B ILLUSTRATION OF VARIOUS PLANAR GRAPHS

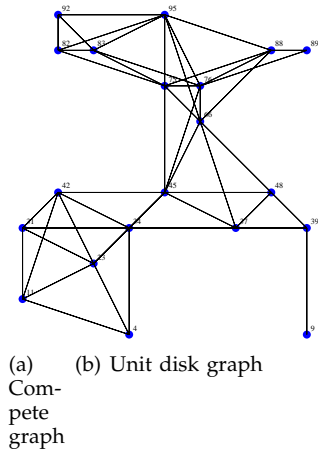


Fig. 9. Original graphs

APPENDIX C
HYPOCOMB FAMILY GRAPHS CONTAIN NO MST

Observe the complete graph in Fig. ?. The corresponding Besh and Hypocomb are shown by dashed thin lines and thick links, respectively. Because vertices a and b have no blocking relation, edge ab does not belong to Hypocomb; whereas, all the other edges do. Notice that, fixing a and b , we may translate c and d arbitrarily far to the west and east, respectively, without alerting the shown blocking relations and making ab the shortest edge in the complete graph. Since the shortest edge always belongs to MST, we conclude that MST is not necessarily part of Hypocomb. From Corollary ??, this conclusion holds for Reduced Hypocomb too. Hypocomb can be considered as a special case of Local Hypocomb, where the input UDG is a complete graph. Thus it is possible that MST is not a subgraph of Local Hypocomb.

APPENDIX D
NUMERICAL RESULTS OF SPANNING RATIO

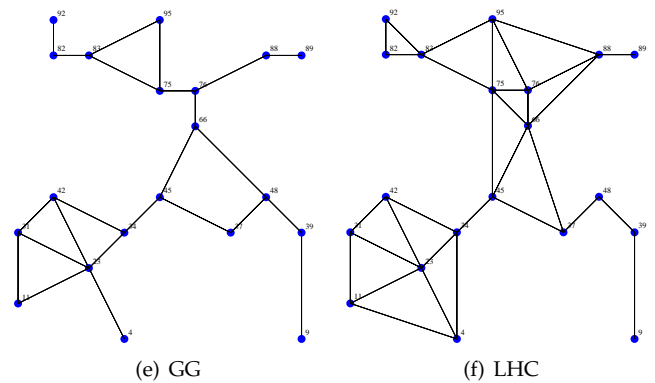
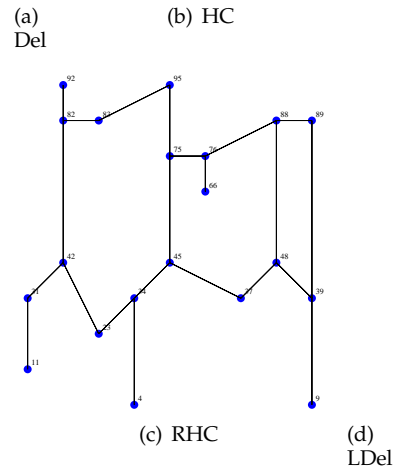
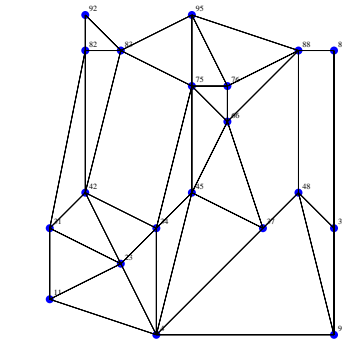


Fig. 10. Planar graphs corresponding to Fig. ?

Fig. 11. A counter example

Fig. 12. Spanning ratio