Graph viz, Hypergraph viz and Processing

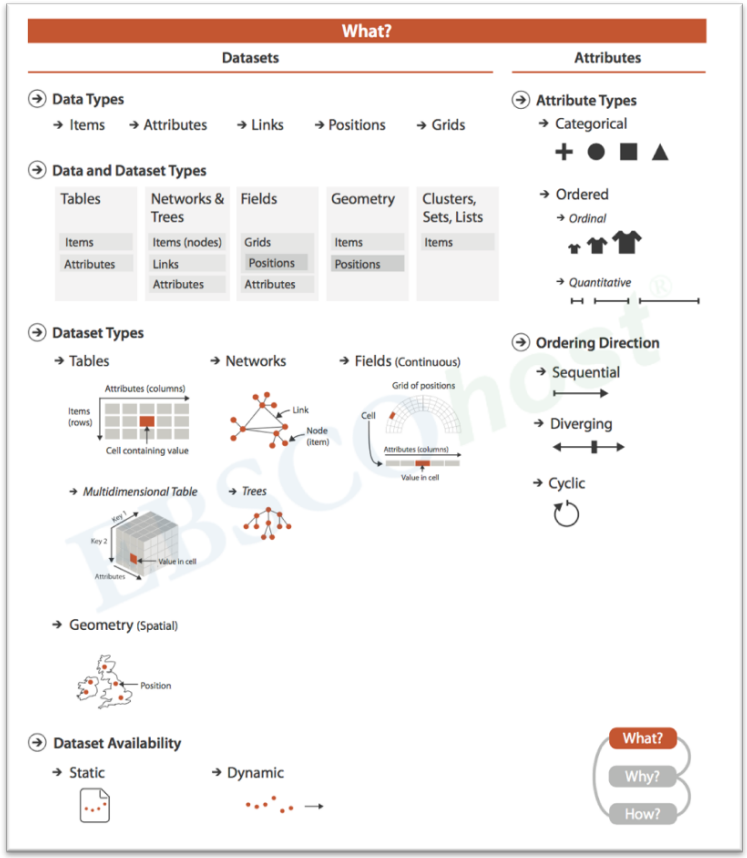
Naheed Anjum Arafat

National University of Singapore

November 25, 2020

Graph Visualization & Gephi

Network Data & Task Taxonomies



- Network **Dataset**: Zachary's karate club Network, Metadata. **Dataset Type**: Network (Node, Link, No attribute) **Representation:** GML format
  - **Semantic**: Link = 'Interaction outside the club' i.e. 'Friendship'
- **Availability**: Static (1970)

Network Data & Task Taxonomies

Tasks:

- Analyze => Draw the network/graph. (Consume?/Produce?)
- Analyze => Finding Community structure (group of densely connected nodes). Consume?/Produce?
- Query => Filter: We often filter nodes which have no links (i.e. Degree 0)

Network Data & Task Taxonomies

Tasks:

- Analyze => Draw the network/graph. (Consume?/Produce?)
- Analyze => Finding Community structure (group of densely connected nodes). Consume?/Produce?
- Query => Filter: We often filter nodes which have no links (i.e. Degree 0)

A graphviz tool supports:

- Different **representations** of network data. (Data representation)
- Different visual encoding i.e. graph **layouts** (Position of the nodes in 2D, Rendering)
- Essential analytic **tasks** and **queries**.

Gephi:- A tool for Graph viz and Analytics

- Load the Dataset into Gephi
- Preview (Settings, Data Table)

Gephi:- A tool for Graph viz and Analytics

- Load the Dataset into Gephi
- Preview (Settings, Data Table)
- Overview:
  - Drawing **Layouts**:- Fruchterman-Reingold, ForceAtlas, ForceAtlas 2

Gephi:- A tool for Graph viz and Analytics

- Load the Dataset into Gephi
- Preview (Settings, Data Table)
- Overview:
  - Drawing **Layouts**:- Fruchterman-Reingold, ForceAtlas, ForceAtlas 2
  - Viz **operations**: Play around with 'Zoom','Pan', Adjust 'Edge thickness', 'Reset Position'.

Gephi:- A tool for Graph viz and Analytics

- Load the Dataset into Gephi
- Preview (Settings, Data Table)
- Overview:
  - Drawing **Layouts**:- Fruchterman-Reingold, ForceAtlas, ForceAtlas 2
  - Viz **operations**: Play around with 'Zoom','Pan', Adjust 'Edge thickness', 'Reset Position'.
  - Coloring Nodes by Degree. Resizing Node radius by Degree.

Gephi:- A tool for Graph viz and Analytics

- Load the Dataset into Gephi
- Preview (Settings, Data Table)
- Overview:
  - Drawing **Layouts**:- Fruchterman-Reingold, ForceAtlas, ForceAtlas 2
  - Viz **operations**: Play around with 'Zoom','Pan', Adjust 'Edge thickness', 'Reset Position'.
  - Coloring Nodes by Degree. Resizing Node radius by Degree.
  - **Statistics**:-
    - **Detect Communities: Compute Modularity Classes.**
    - **Colorize the communities:- Partition nodes by the Modularity Classes.**

Gephi:- A tool for Graph viz and Analytics

- Load the Dataset into Gephi
- Preview (Settings, Data Table)
- Overview:
  - Drawing **Layouts**:- Fruchterman-Reingold, ForceAtlas, ForceAtlas 2
  - Viz **operations**: Play around with 'Zoom','Pan', Adjust 'Edge thickness', 'Reset Position'.
  - Coloring Nodes by Degree. Resizing Node radius by Degree.
  - **Statistics**:-
    - **Detect Communities: Compute Modularity Classes.**
    - **Colorize the communities:- Partition nodes by the Modularity Classes.**
  - **Filter**:- Hide nodes whose degree $<= 1$.
- Refer to:- `https://gephi.org/tutorials/gephi-tutorial-quick_start.pdf` for more.

Hypergraph Drawing using Force-directed Placement.

Published in 'Database and Expert Systems Applications, 2017'.

Naheed Anjum Arafat, Stéphane Bressan

Hypergraphs

Hypergraph: A finite collection of *set of objects*. The objects are called *vertices*. The sets are called *hyperedges*.

- Hypergraphs can capture multi-ary relationships.
- Hypergraphs generalize graphs (binary relationships).

Example:

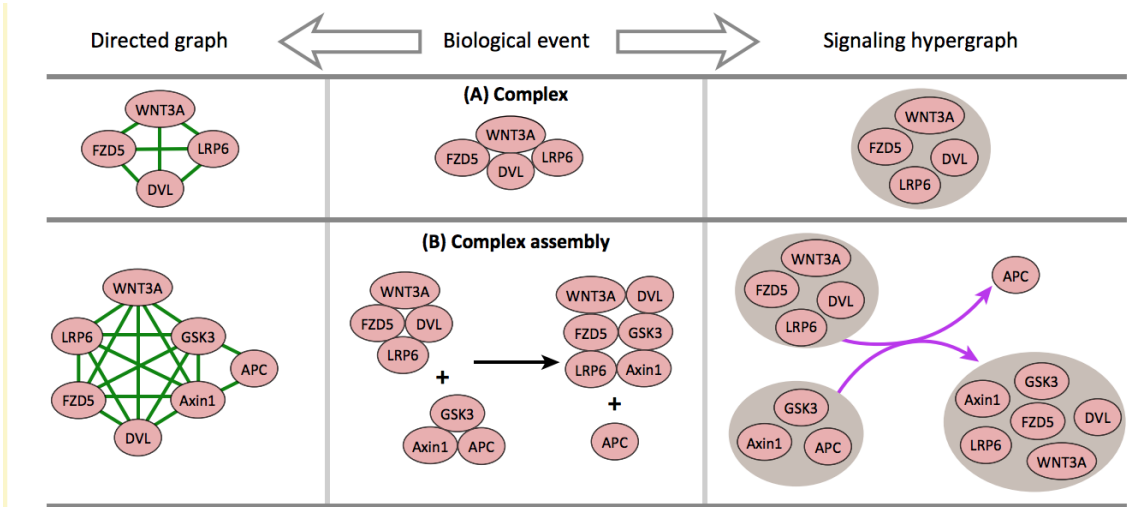


Figure 1: Protein complex represented as a complete graph (top row, leftmost) and a hyperedge(top row, rightmost). Reaction between complexes (Complex Pathway) represented as directed hypergraph (bottom row, right).

Applications

- Social Networks: In modeling communities[1], Tagging relationships in music social networks[2].
- Database: In representing Database Schema[3].
- Biology: In representing Yeast protein network[4], Biochemical reaction network[5].

[1] Michael Brinkmeier, Jeremias Werner, and Sven Recknagel. "Communities in graphs and hypergraphs". In: *Conference on information and knowledge management*. ACM. 2007.

[2] Jiajun Bu et al. "Music recommendation by unified hypergraph: combining social media information and music content". In: *Proceedings of the 18th ACM international conference on Multimedia*. ACM. 2010.

[3] Ronald Fagin. "Degrees of acyclicity for hypergraphs and relational database schemes". In: *Journal of the ACM (JACM)* (1983).

[4] Emad Ramadan, Arijit Tarafdar, and Alex Pothen. "A hypergraph model for the yeast protein complex network". In: *Parallel and Distributed Processing Symposium, 2004. Proceedings. 18th International*. IEEE. 2004.

[5] Can Özturan. "On finding hypercycles in chemical reaction networks". In: *Applied Mathematics Letters* (2008).

Hypergraph Visualization Literature

There are two basic methods for drawing a hypergraph.

- **Subset based**:- A hyperedge is drawn as a closed curve **enveloping** its vertices.
- **Edge based**:- A hyperedge is drawn as a set of curves **connecting** its vertices.

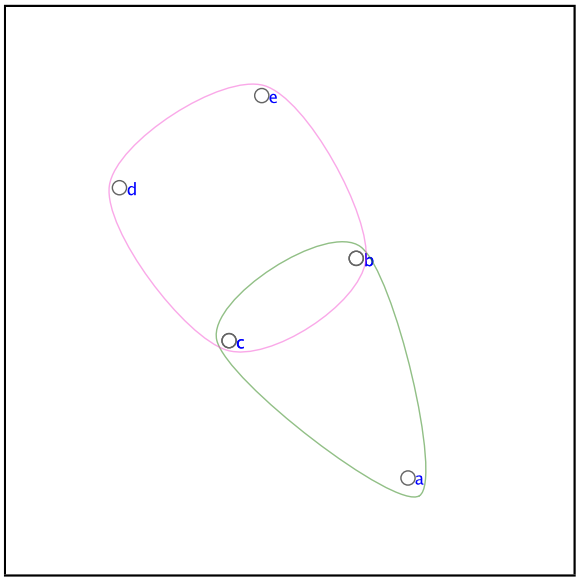


Figure 2: **Subset-based drawing**.

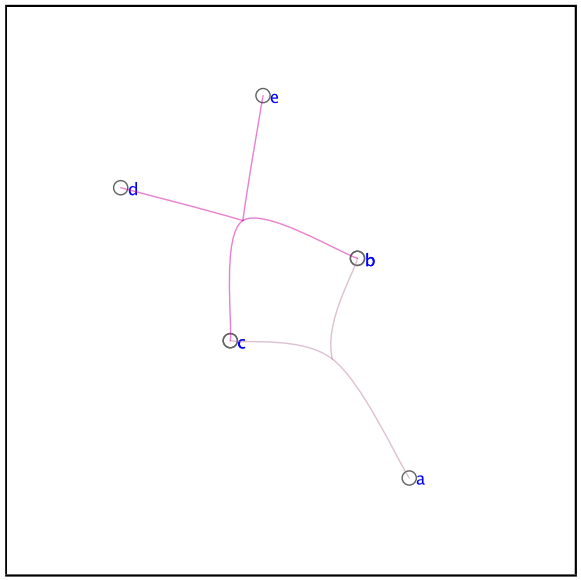


Figure 3: **Edge-based drawing**.

Set visualization approaches

- Euler diagram, Venn diagram:

  - Venn diagrams are special kind of Euler diagrams (with constraints such as, all possible intersections must be displayed).
  - Euler diagrams are special kind of 'subset based' drawings (with constraints such as, empty zones are not allowed).

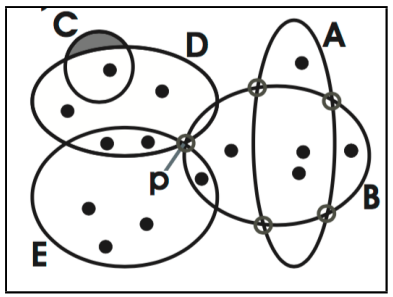


Figure 4: A 'Subset based' drawing. Zones as shaded in C are not allowed in Euler diagram[6].

[6] Rodrigo Santamaría and Roberto Therón. "**Visualization of intersecting groups based on hypergraphs**". In: *IEICE TRANSACTIONS on Information and Systems* (2010).

Set Visualization approaches (contd.)

- Bubble Sets[7]: Sets are visualized using continuous, **isocontours**.
- LineSets: Sets are visualized using continuous **curves**.

Figure 5: Bubble Sets and LineSets of three set of hotels on the map.[8]

[7] Christopher Collins, Gerald Penn, and Sheelagh Carpendale. "Bubble sets: Revealing set relations with isocontours over existing visualizations". In: *IEEE Transactions on Visualization and Computer Graphics* (2009).

[8] Basak Alper et al. "Design study of linesets, a novel set visualization technique". In: *IEEE transactions on visualization and computer graphics* (2011).

Problem Statement

- We want to have **aesthetically pleasing** drawing of hypergraphs in subset standard.
  - **We propose a family of algorithms.**

Problem Statement

- We want to have **aesthetically pleasing** drawing of hypergraphs in subset standard.
  - **We propose a family of algorithms.**
- We want to evaluate the drawing **quality** by some **measurable criteria**.
  - We propose several metrics.

Algorithms

A Detour to Graph Drawing

A Detour to Graph Drawing

Fruchterman-Reingolds (FR) Force-directed algorithm:

- Vertices: Objects in a physical system.
- Vertices connected (not connected) by edges attract (repel) each other.
- Advantages: Simple, Uniform edge length, Symmetry

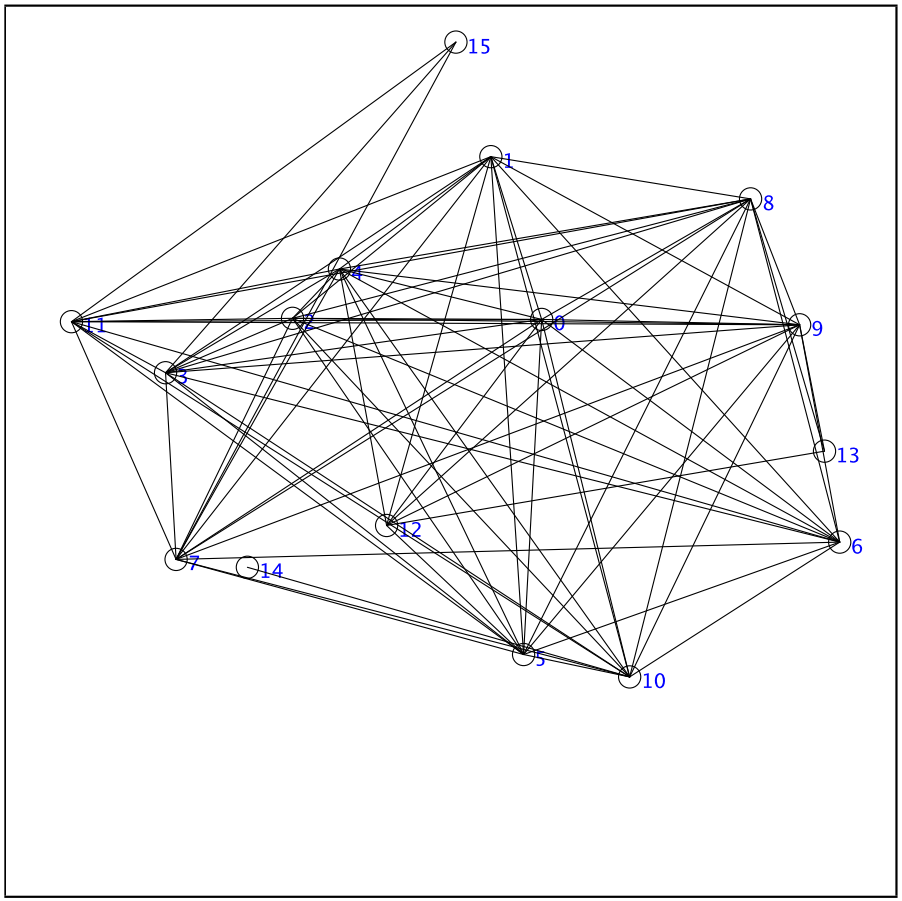


Figure 6: **Randomly initialized drawing of a graph.**

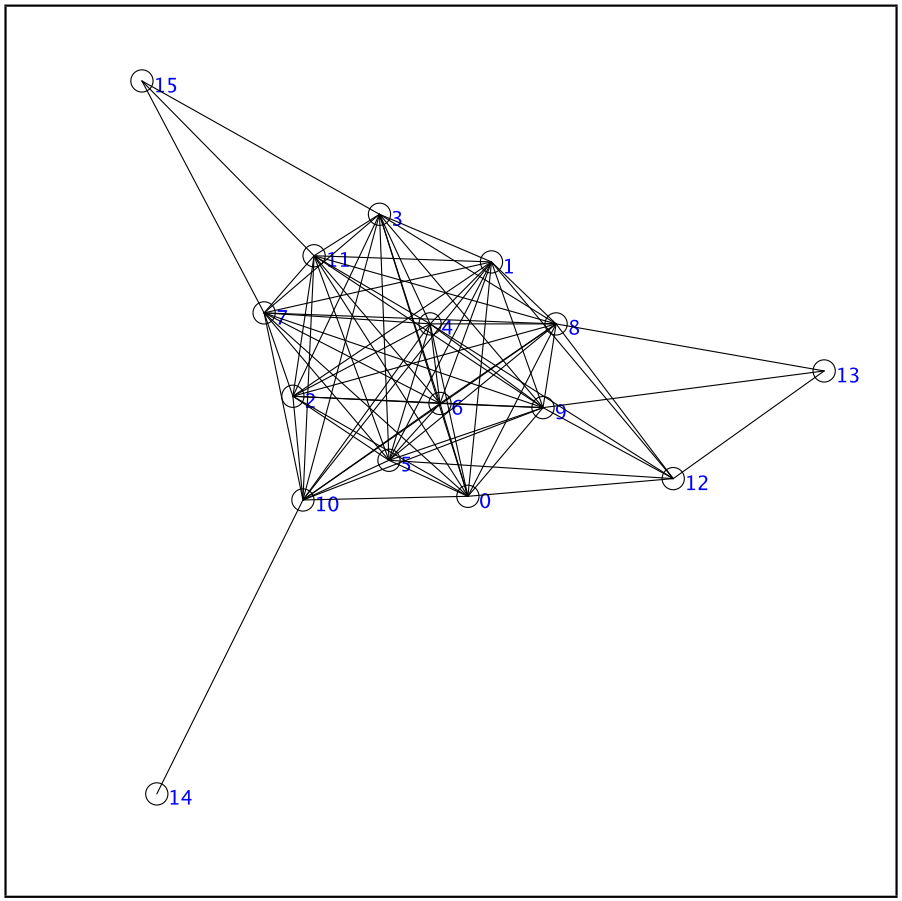


Figure 7: **The same graph drawn by FR algorithm.**

Algorithm for Hypergraph drawing: Subset based

Algorithm for Hypergraph drawing: Subset based

Given a Hypergraph, $H = \{\{a, b, c, d\}\}$

Algorithm for Hypergraph drawing: Subset based

Given a Hypergraph, $H = \{\{a, b, c, d\}\}$

- Transform the Hypergraph to a Graph (namely, the *Associated graph* of a hypergraph).

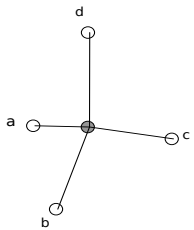


Figure 8: **Star Associated graph**

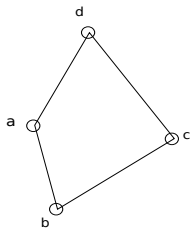


Figure 9: **Cycle Associated graph**

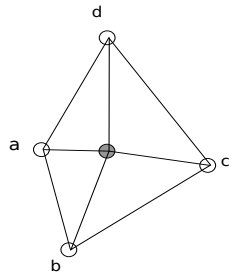


Figure 10: **Wheel Associated graph**

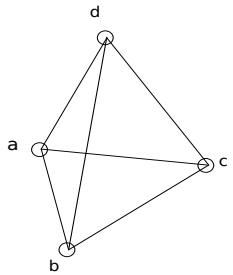


Figure 11: **Complete Associated graph**

Algorithm for Hypergraph drawing (contd.)

- Each of the transformations induces an algorithm (Star/Cycle/Wheel/Complete algorithm). Draw the Associated graph using FR algorithm (or any Force-directed graph layout algorithm).

Algorithm for Hypergraph drawing (contd.)

- Each of the transformations induces an algorithm (Star/Cycle/Wheel/Complete algorithm). Draw the Associated graph using FR algorithm (or any Force-directed graph layout algorithm).

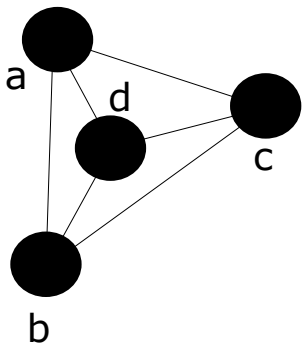


Figure 12: The layout of the *Complete associated graph* after applying FR algorithm.

Algorithm for Hypergraph drawing (contd.)

- Each of the transformations induces an algorithm (Star/Cycle/Wheel/Complete algorithm). Draw the Associated graph using FR algorithm (or any Force-directed graph layout algorithm).
- Draw a closed curve enveloping the vertices of each hyperedge.

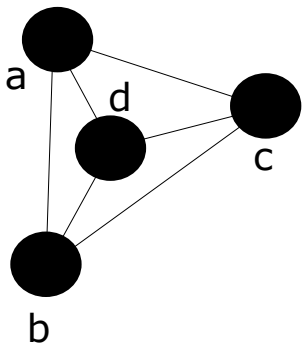


Figure 12: The layout of the *Complete associated graph* after applying FR algorithm.

Algorithm for Hypergraph drawing (contd.)

- Each of the transformations induces an algorithm (Star/Cycle/Wheel/Complete algorithm). Draw the Associated graph using FR algorithm (or any Force-directed graph layout algorithm).
- Draw a closed curve enveloping the vertices of each hyperedge.

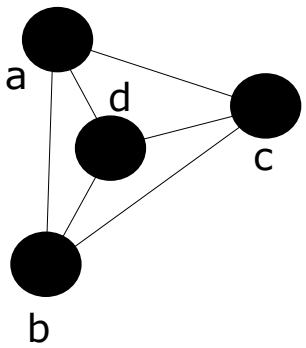


Figure 12: The layout of the *Complete associated graph* after applying FR algorithm.

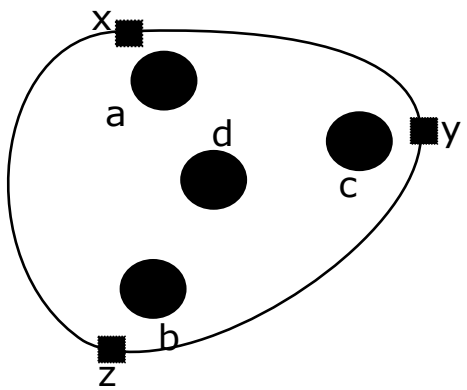


Figure 13: A closed curve is drawn enveloping the vertices.

In Details.

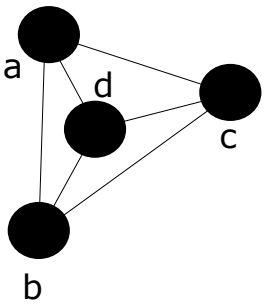


Figure 14: The vertices of a hyperedge $\{a, b, c, d\}$ after drawing its Complete associated graph.

In Details.

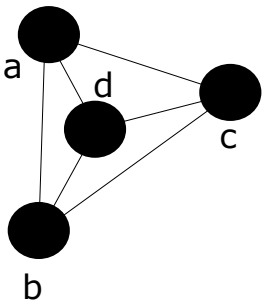


Figure 14: The vertices of a hyperedge $\{a, b, c, d\}$ after drawing its Complete associated graph.

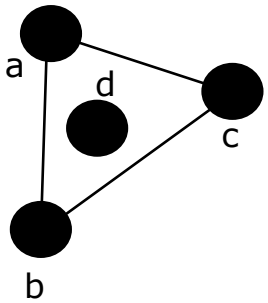


Figure 15: Convex hull of the vertices a, b, c, d and its bordering vertices a, b, c.

In Details.

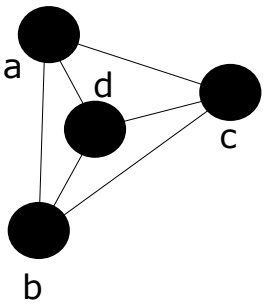


Figure 14: The vertices of a hyperedge $\{a, b, c, d\}$ after drawing its Complete associated graph.

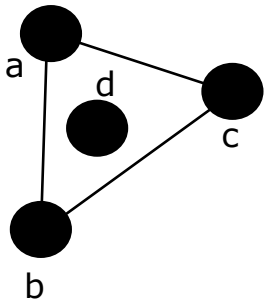


Figure 15: Convex hull of the vertices a, b, c, d and its bordering vertices a, b, c.

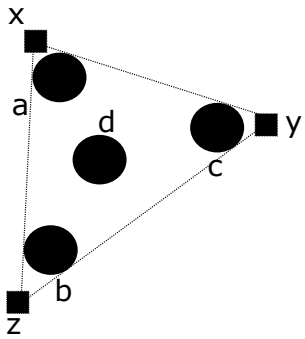


Figure 16: Pair-wise outtangents of the bordering vertices and points x, y, z as their intersections.

In Details.

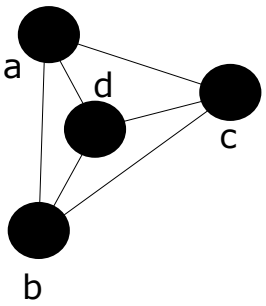


Figure 14: The vertices of a hyperedge $\{a, b, c, d\}$ after drawing its Complete associated graph.

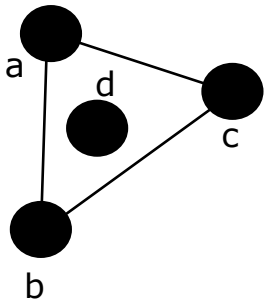


Figure 15: Convex hull of the vertices a, b, c, d and its bordering vertices a, b, c.

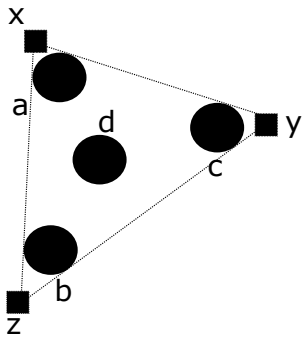


Figure 16: Pair-wise outtangents of the bordering vertices and points x, y, z as their intersections.

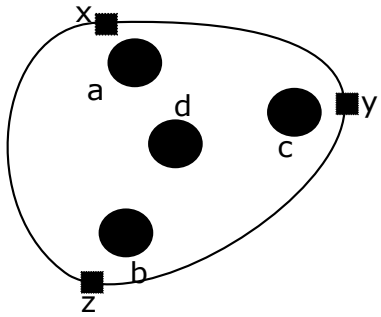


Figure 17: The hyperedge drawn as a closed Catmull-Rom Spline going through x, y, z.

Some drawings

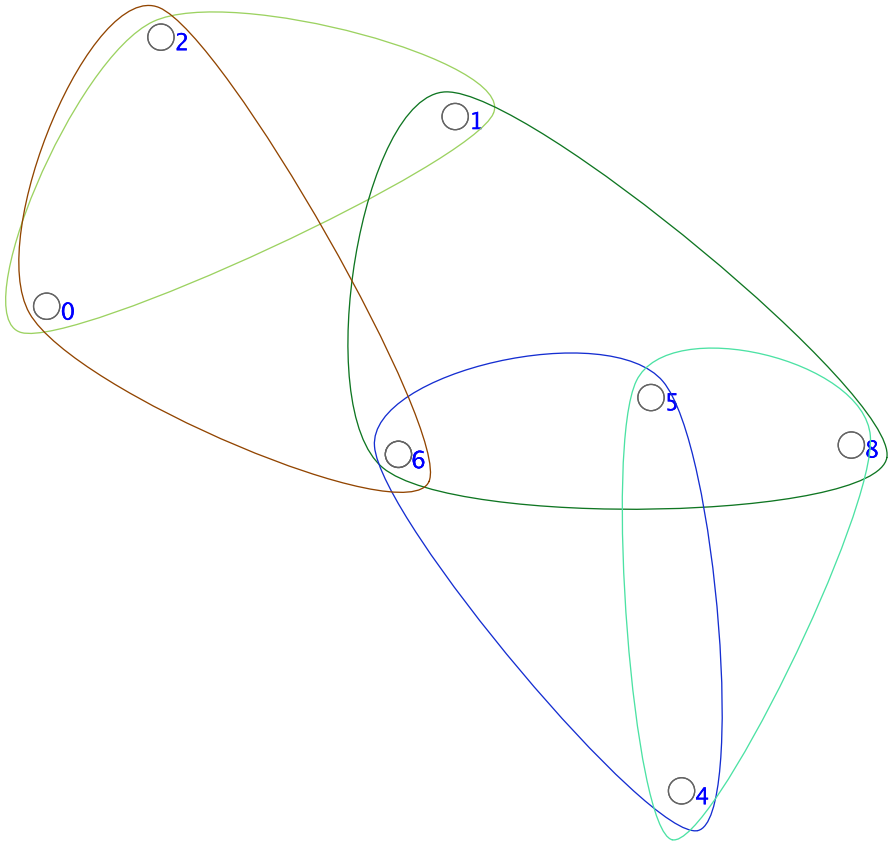
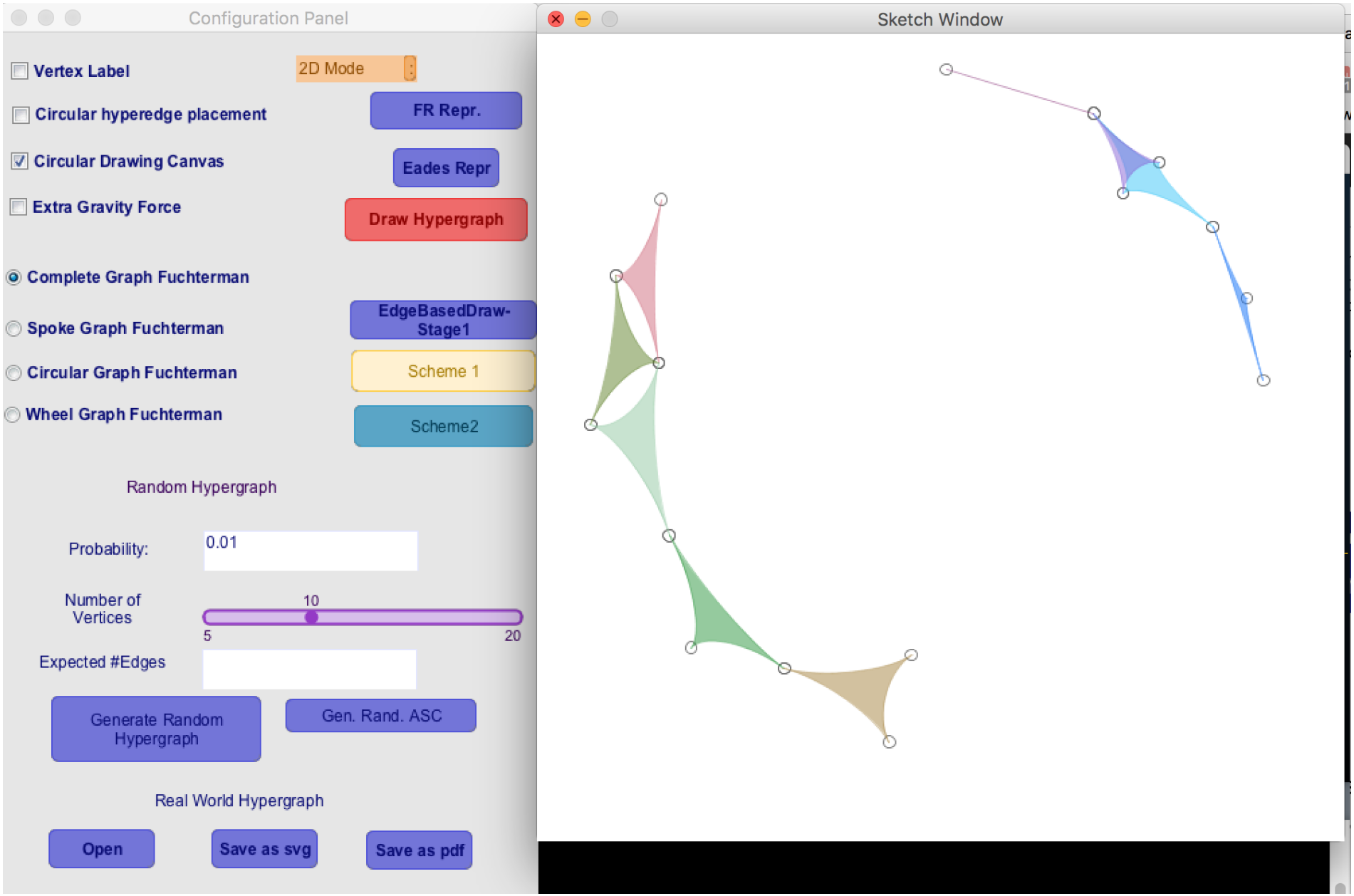


Figure 18: A 3-uniform hypergraph with 5 hyperedge.

Some drawings

Figure 19: A hypergraph with 100 hyperedges randomly sampled from the DBLP dataset

Quick peek at the current implementation



Conclusion

- We propose a family of algorithms for drawing hypergraphs.
- We propose measurable criterion to evaluate the goodness of the drawings.
- Possible improvement:
  - Model hyperedges as elastic manifolds.
  - Exploit the hierarchical structure of sets and subsets. (Hierarchical drawing)
- Collaborations:

We have few ideas. Interested?

Processing

Processing

Processing

- is a 'java'-ish Open-Source programming language.
- codes are compiled in Java virtual machine (JVM) internally before they are executed.
- is way simpler than *Java*.
- was born in MIT Media Lab in 2001 (Casey Reas and Benjamin Fry).
- comes with its own IDE (Sketchbook)
- has a very smooth learning curve, even if you do not know *Java*

Cool fact: Write a processing code. You can embed it inside any HTML *canvas* using **processing.js** — a JavaScript port of processing.

A Flavor of Processing 3

Example Code:

- Drawing Circles, lines.

A Flavor of Processing 3

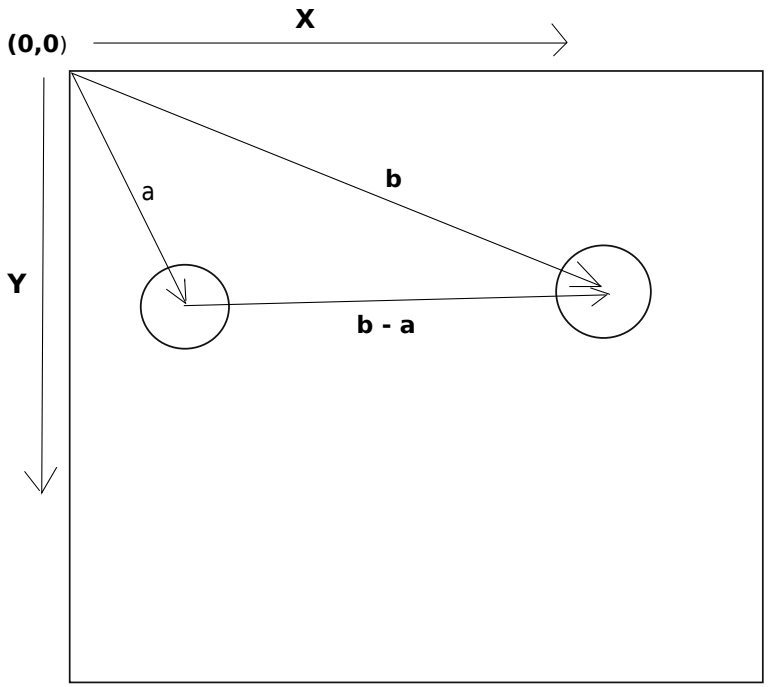
Example Code:

- Drawing Circles, lines.
- Simulating Attraction between two circular bodies connected by elastic rubber.

A Flavor of Processing 3

Example Code:

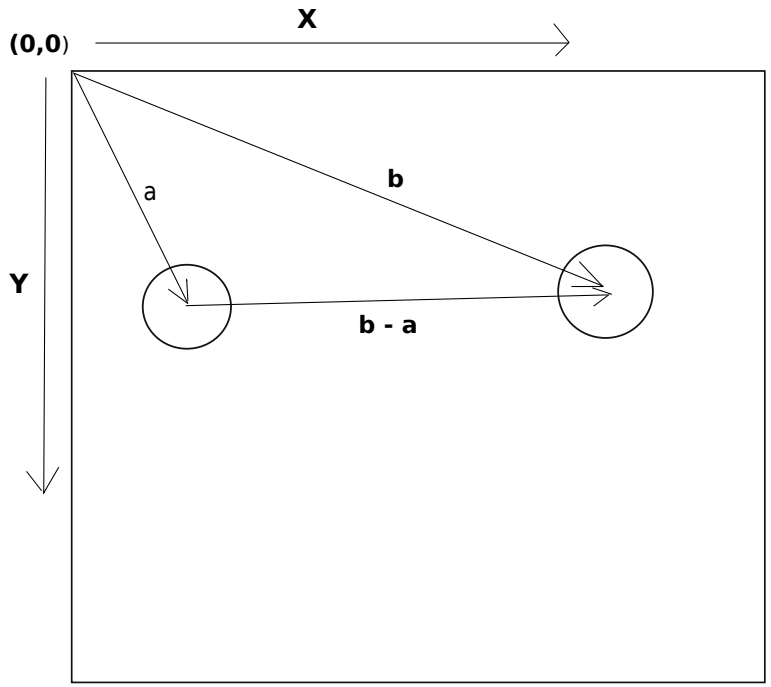
- Drawing Circles, lines.
- Simulating Attraction between two circular bodies connected by elastic rubber.



- Lets add repulsion too! (Try it!!)

Example Code:

- Drawing Circles, lines.
- Simulating Attraction between two circular bodies connected by elastic rubber.



- Lets add repulsion too! (Try it!!)
- Did we just implement a **Force-directed algorithm for a 2-node graph?**

Things I like about processing:-

- Almost any native java package can be used in processing (Though, you are advised to not use some e.g. awt, swing etc.)
- Any third party java package can be used. (Useful when you run statistical tests)
- Processing has its own repository of Libraries. (Hats off to *G4p GUI builder*!)
- Native support for exporting the sketch in pdf, svg format. (thanks to beginRecord(),endRecord() functions)
- You will be amazed to see its core library functions:- https://processing.org/reference/
- Low level graphics (like OpenGL):- Rotation, translation, scaling, shearing.
- Advanced graphics:- Lighting, Texture mapping, Shader.
- Neat documentation.

Questions-

naheed_anjum@u.nus.edu