Challenges in visualizing large graphs and hypergraphs @CERN

Collaboration spotting

A. Agocs, D. Dardanis, R. Forster, J.-M. Le Goff, X. Ouvrard, R. Rattinger

Speaker at GPU-Days Wigner 2017 :
X. Ouvrard, PhD student UniGe/CERN
Supervisor : S. Marchand-Maillet (UniGe)

Talk map

- I Presentation of Collaboration Spotting (team work)
- II Mathematical background (PhD related work of XO)
- III Large graphs and hypergraphs visualisation (PhD related work of XO)
- IV Questions

Talk map

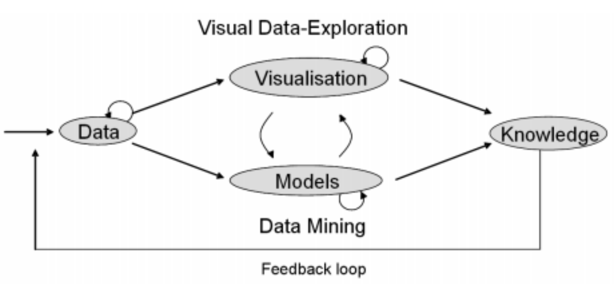
- **I Presentation of Collaboration Spotting (team work)**
- II Mathematical background (PhD related work of XO)
- III Large graphs and hypergraphs visualisation (PhD related work of XO)
- IV Questions

Project context

- CERN Project : Collaboration Spotting, team of J.M. Le Goff
  - At the beginning :
    - serve the particle physic community with a data visualization tool,
    - first use case : publications and patents data
  - Goal of project : deliver a **generic data visualization tool** that supports the **visual analytics** process
- Different applications
  - With JRC, EC : TIM : http ://www.timanalytics.eu/
  - Use in ARIADNE, LHCb
  - Other applications on study, some with Wigner institute

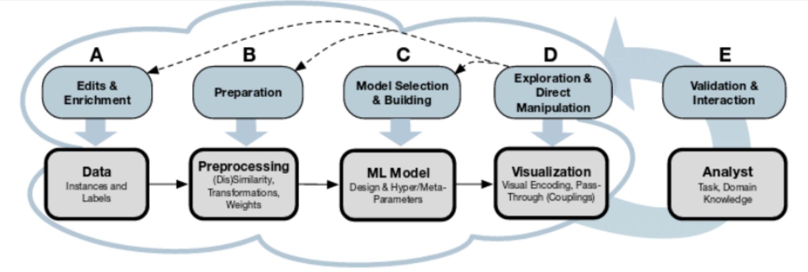
Vision of project

- Experts have the knowledge and data scientists have the skills :
  => Bring analytics to experts
- Collaboration spotting to support the visual analytics process defined by Keim & al. [8]

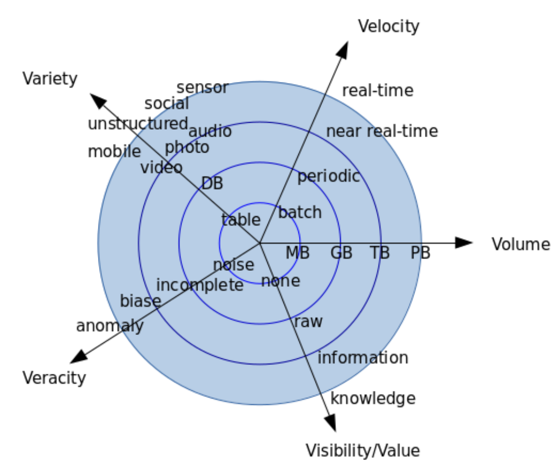


Collaboration spotting framework

- The project support the proposed conceptual framework of Sacha & al. [10]



3 Vs of META group (Gartner group) extended to 5 Vs :



Realized by X. Ouvrard

Big Data mining

- Data mining : only one step of the **knowledge discovery** processing chain from data, see for instance Han & al. [7]
- In non numerical data, choices :
  - summarize data with number of occurences
  - making links :
    - regroup data through similarity
    - retrieve **links through data** itself
- Data is stored with **metadata** attached to it
  - For instance : publications and patents : title, abstract, author, organisation, ...
- From metadata :
  - some is of interest for **analysis** : title, abstract, citations
  - some is of interest for **visualisation** : organisations, cities, keywords, ...

Collaboration spotting (1)

- In CS : we want to visualise the **multi-dimentional** network structure and **interconnectivity** from different **user-defined** perspectives.
  To this end we need to :
  - Compute collaborations with respect to a particular selection of network dimensions
  - Visualize these collaborations in a way that enhances cognitive perception.

Collaboration spotting (2)

- To achieve it :
  - learning the **intrinsic network structure** is needed such as :
    - connected components
    - node degree distribution
    - communities, ...
  - when the number of dimensions/types is large different techniques must be combined :
    - proper modeling of networks through hypergraphs
    - learning on hypergraphs
    - semantic abstraction => semantic filtering => abstraction of types in the same view

Hypergraphs

Introduced by Berge in Berge & al. [3] :
An **hypergraph** $\mathcal{H}$ on a finite set $V = \{v_1\ ;\ v_2;\ ...\ ;\ v_n\}$ is a family of hyperedges $E = (e_1, e_2, ..., e_m)$ where each **hyperedge** is a non-empty subset of V and such that $\bigcup_{i=1}^{m} E_i = V$.
Written : $\mathcal{H} = (V, E)$
Hyperedge links one or more vertices.
In Bretto [4] : $\bigcup_{i=1}^{m} e_i = V$ is relaxed. The vertices belonging to $V \setminus \bigcup_{i=1}^{m} e_i$
Order of $\mathcal{H}$: $|V|$
Size of $\mathcal{H}$: $|E|$

Talk map

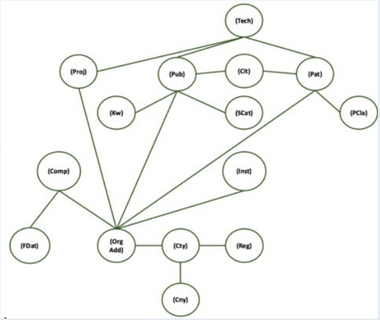
- I Presentation of Collaboration Spotting (team work)
- **II Mathematical background (PhD related work of XO)**
- III Large graphs and hypergraphs visualisation (PhD related work of XO)
- IV Questions

Reachability hypergraph

- Traditional DB structure can be seen as hypergraphs, where the hyperedges are the metadata that are grouped into one table. Normalisation forms of such DB are linked to properties of the hypergraph. For details cf Fagin & al. [5], Beeri & al. [1].
- **Reachability** in a hypergraph :
  Two nodes u and v of a hypergraph are said reachable if either u and v are identical or it exists one node w such that u and w belong to the same hyperedge and w and v are reachable.
- **Building an hypergraph** from the metadata :
  - A physical reference is chosen. It is the base for the hyperedge
  - A metadata belongs to an hyperedge, if it is held by the reference

Reachability graph

- For instance : publications contain organisations, author keywords, ...
- Compound hypergraphs are needed to have full modelization
- The reachability graph is obtained by developping the compound hypergraph



Courtesy of JM Le Goff

Building collaborations

In the reachability graph :

- choice of a reference node for collaborations
- any other node that is linked to the reference by a minimal path can be used as a visual dimension

For instance : Publication p, containing a_p metadata of type α ; it defines a set : $A_{\alpha,p} = \left\{att_1, ..., att_{a_p}\right\}$, which is the set of co-attributes of type α.

If a search S is made on publications : retrieval of $A_{\alpha,S} = \bigcup_{p \in S} A_{\alpha,p}$ set of co-α attributes.

One $A_{\alpha,p}$ per article, eventually empty, so : $\mathcal{A}_{\alpha,S} = \{A_{\alpha,p} | p \in S\}$.

$A_{\alpha,S}$ set of nodes and $\mathcal{A}_{\alpha,S}$ set of hyperedges of coattributes of type α.

$\mathcal{H}_{\alpha,S} = \left(A_{\alpha,S}, \{A_{\alpha,p} | p \in S\}\right)$: hypergraph of co-attributes of type α in the search

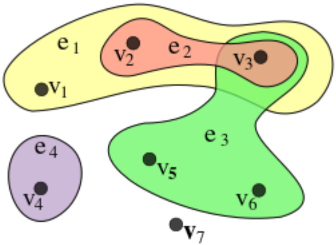
Theoretical approach : browsing

- If we want co-attributes of type α' on the same search, $\mathcal{H}_{\alpha',S} = \left(A_{\alpha',S}, \left\{A_{\alpha',p} | p \in S\right\}\right)$ is retrieved : $=>$by this way internal browsing in a search is achieved
- To know all the possible browsing possibilities :
  - In a set $\mathcal{S}$ of references : set T of types α
  - **New graph** S_{schema}.
    - Nodes = elements of T.
    - Edges : Two nodes α and α' of S_{schema} linked if attributes of type α and α' are in the same reference.
  - When a search is made : subgraph of S_{schema} is retrieved
  - S_{schema} and its restrictions helped to know the authorized navigation

Visualizing hypergraphs

Many solutions

- Venn diagrams :
  - each hyperedge is a closed curve
  - each node is represented by a point
  - major problem : not scalable



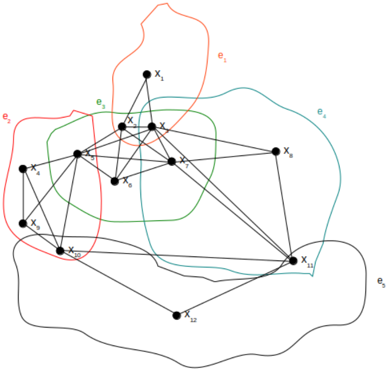
Source : Wikipédia

- Building the 2-section of the hypergraph $\mathcal{H}$:
  => graph where :
  - the nodes are the nodes of $\mathcal{H}$
  - two nodes are linked by an edge if they belong to the same hyperedge :

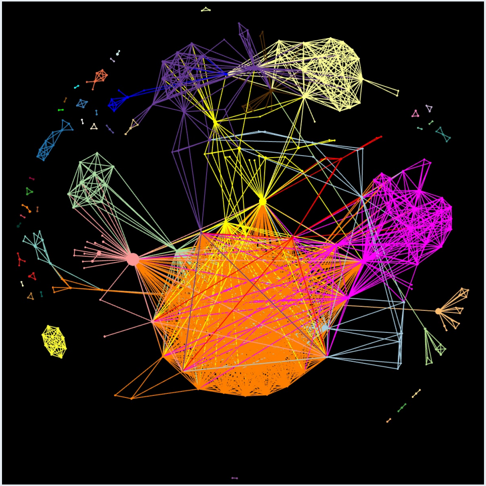
  => also called clique expansion of the hypergraph

  It is the traditionnal approach in sociograms

Hypergraphs visualization

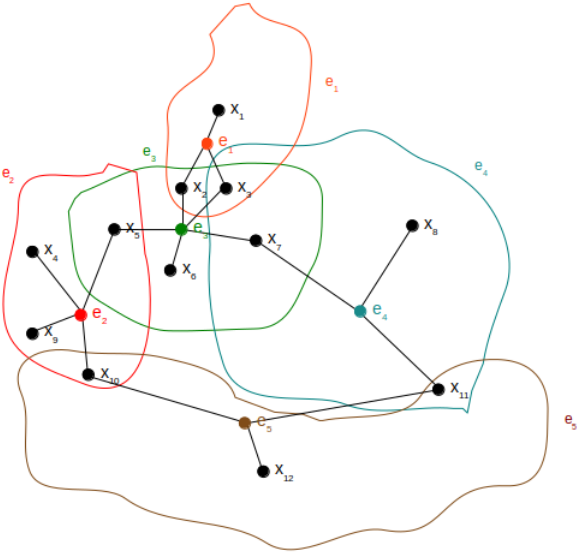


Clique expansion

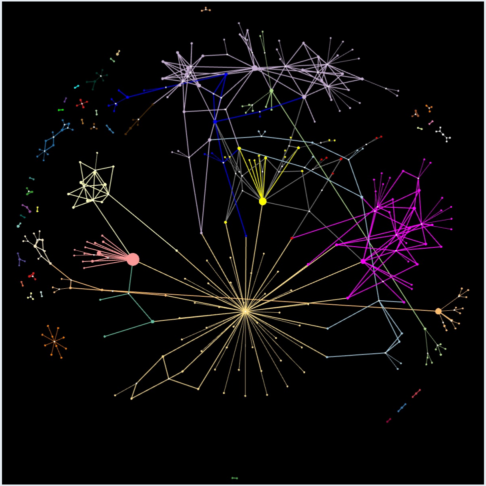


- Other approach : incident graph of the hypergraph $\mathcal{H} = \left(V, E = (e_i)_{i \in I}\right)$:
  Bipartite graph, also called extra-node graph and written $X(\mathcal{H}) = \left(V', E'\right)$ such that :
  - two nodes in $X(\mathcal{H})$ are the elements of V and those of V_X,set of nodes corresponding to each $e_i \in E$ with $i \in I$, which are called extra-nodes and abusively written e_i. Hence : $V' = V \cup V_X$ and $V \cap V_X = \emptyset$.
  - two nodes v and e of V' are linked if $v \in V$ and $e \in V_X$ and $v \in e$ in $\mathcal{H}$.

Visualizing hypergraphs



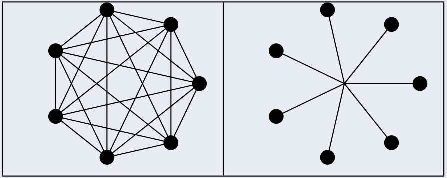
Extra-node view



Comparison between the two approaches

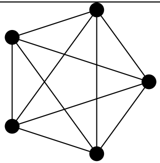
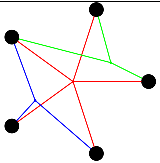
Possible gain : $\frac{n(n-3)}{2}$, as soon as : $n > 3$

Example : n=7



Comparison in between the two approaches

Unfavorable cases exist :

Clique view	Extra node view
	
10 edges, 5 nodes	11 edges, 5 nodes, 3 extra nodes

Comments :

- Collaborations distribution has to be analysed
- Importance of evaluating the gain in edges, but also in the retrieved information

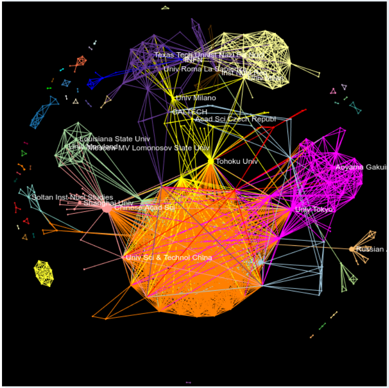
Conclusion for visualization of hypergraphs (1)

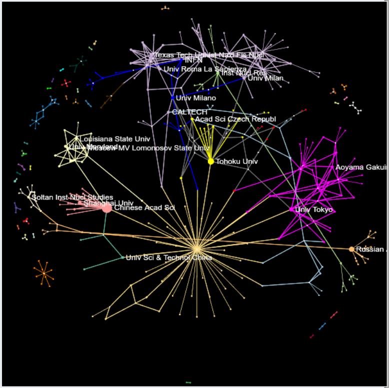
Hypergraphs :

- Allow navigability
- Visualisation can be improved with the extra-node view
- Importance of experimental evaluation to evaluate real gain.

=> experimental evaluation has been made that shows there is a real gain in visualization

Comparison for visualisation of hypergraphs (2)





Talk map

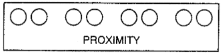
- I Presentation of Collaboration Spotting (team work)
- II Mathematical background (PhD related work of XO)
- **III Large graphs and hypergraphs visualisation (PhD related work of XO)**
- IV Questions

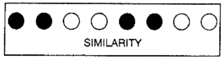
Remaining problem :
How to visualize large graphs with maximal knowledge discovery, nice layouts in a time acceptable for the user ?

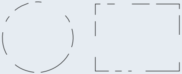
Making readable graphs when it scales up raises different challenges :

- graphs should have nice aesthetics
- they should give meaningfull information
- compute fast in a reasonnable time (0.5-10 s).

Aesthetics for graphs : based on Gestalt principles of groupings, see Wertheimer & al. [12], Rock & al. [9]







Closure

Good continuation

Figures from Rock & al. [9] and Wikipedia

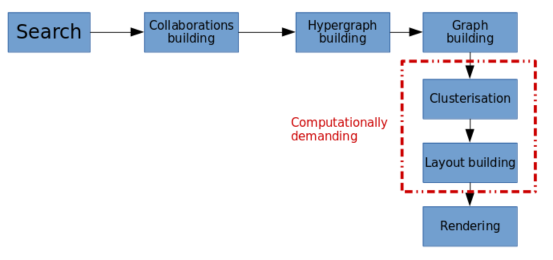
Grouping law	Nodes	(Hyper)Edges	Graph	Hypergraph
Proximity			Usage of clustering Layout algorithm	Usage of clustering Layout algorithm
Similarity	Shape Color (Texture) Size	Color Shape Size		
Closure		Avoid undesirable intersections		
Good continuation		Representation of hyperedges by bunch of edges		
Enclosure			Separation of connected components	
Connectedness				Importance of collaborations : 2-adic vs n-adic

Graph aesthetics

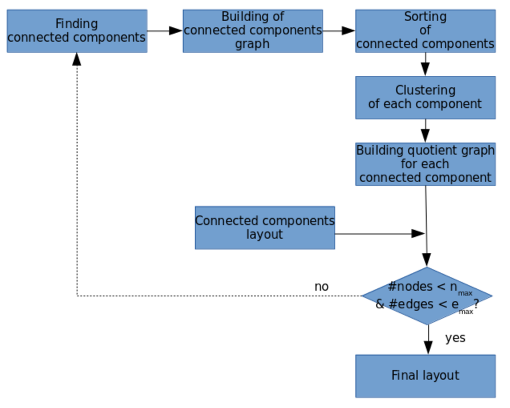
Graph drawing aesthetics as cited by Benett & al. [2] (kind column is added)

Concern	Kind	Aesthetic	Perceptual support
Nodes	Similarity	Clusterize similar nodes	symmetry, proximity
	Distribution	Distribute nodes evenly	
		Keep nodes apart from edges	limits of human eye resolution
		Nodes should not overlap	connectedness
		Maximize node orthogonality	orientation
Edges	Length	Keep edge lengths uniform	similarity
		Minimize total edge length	proximity
		Minimize maximum edge length	proximity
	Bends	Keep angle of edge bends uniform	similarity
		Keep position of edge bends uniform	similarity
		Number of bends in polyline should be minimized	orientation, good shape
	Crossings	Number of crossings should be minimized	continuation
	Angle	Maximize orthogonality : arcs and segments as parallel as possible to horizontal and vertical	orthogonality, good shape
		Maximize minimum angle for incident edges	limits of human eye resolution
	Directed	Maximize flow direction in directed graphs	similarity, orientation
Graph	Local	Maximize local symmetry	symmetry
	Global	Maximize global symmetry	symmetry
		Maximize convex faces	good figure
		Keep correct aspect ratio	good figure
		Area of the graph drawing should be minimized	good figure

Challenge of computations



Layout building process

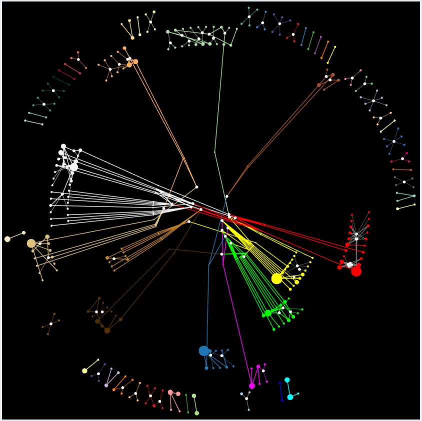


Ideas behind

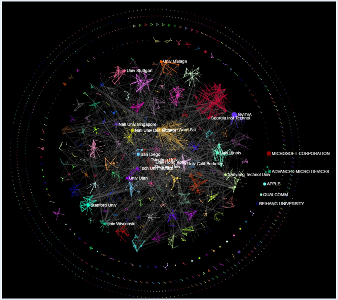
- Direct computing with force-directed algorithms has two problems for large graphs, cf Tamassia & al. [11]
  - complexity at each iteration : $O\left(|E| + |V|^2\right) = O\left(|V|^2\right)$ => can be reduced to $O(|V| \log |V|)$ by Barnes-Hut optimization
  - computation time can be reduced by parallelisation, vectorisation (cf R. Forster talk)
  - but main problem : a lot of local minima => very annoying for graphs above 60 to 80 nodes => low quality of the layout obtained => hard to improve
- Circular layout, cf Gansner & al. [6] :
  - complexity in $O(|V|)$
  - if optimization on edge cuts : $O(|V| + |E|)$ at each iteration
  - edge bundling can be made

Ideas behind (2)

- Multi-circular layout approach on hypergraph :
  - complexity is low at a first level : $O(|V|)$
  - calculation of the quotient graph : placement of clusters
  - if placement of clusters and nodes to minimize edge cuts increase the worst complexity to $O(|C|^2 + |C| \max(\text{size}(C)))$
  - improve knowledge discovery, but center is occupied



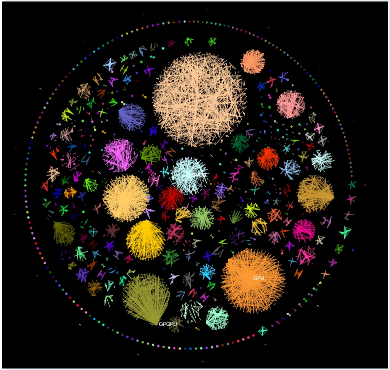
- Combine the circular approach and the directed layout :
  - **Divide and conquer** approach :
    - computing the quotient graph based on the community
    - layout for each community
    - layout for the quotient graph
    - final layout, combining the two

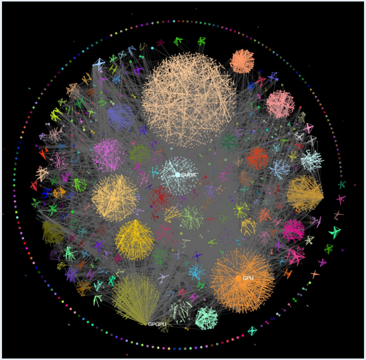


Intercluster edges are drawn in grey.
CS allow to hide them.

Quotient graph based layouts

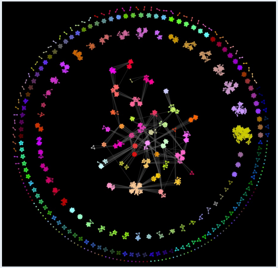
- The **quotient graph** corresponds to the graph of the communities obtained in the clustering.

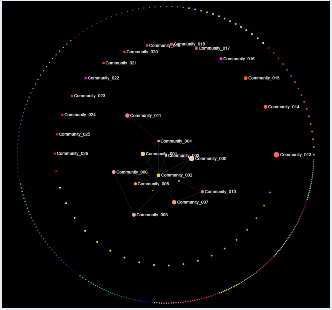




Ideas behind (4)

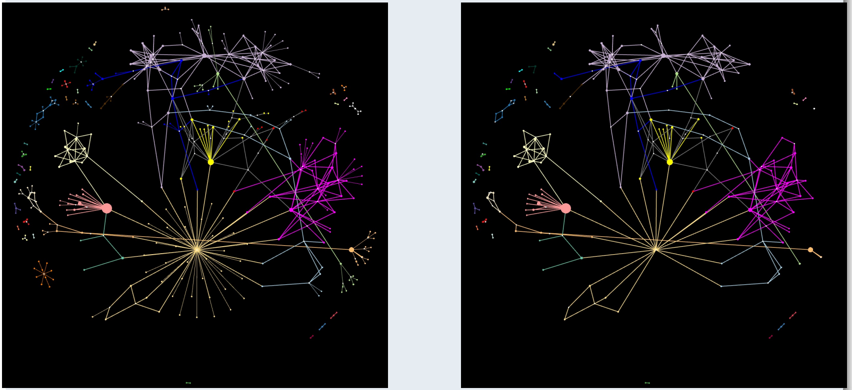
- Important things should be at the center :
  - Approach by **connected components**
  - Sorting connected components, displaying them by circular layout
- When the number of nodes or edges is above a threshhold : display the **quotient graph**.
  => meaning of communities : domain specific : needs of ontology



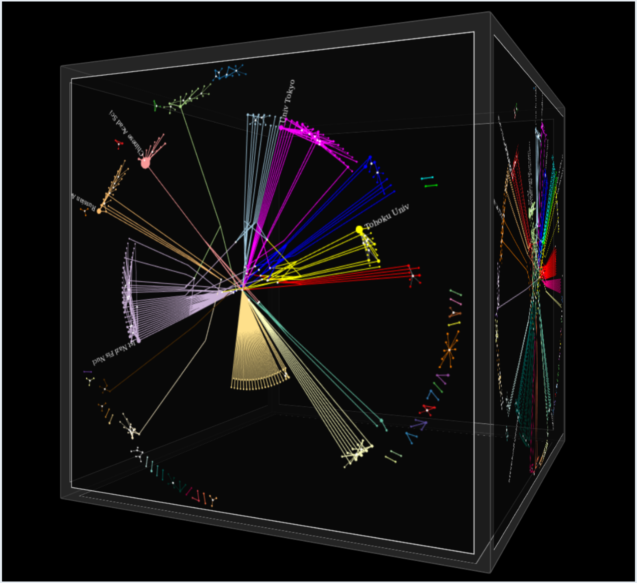


Random graph : 500 collaborations (25000 initial nodes), 1996 nodes, 5976 edges, 349 clusters (39 interconnected), 311 connected components

Filtering with hypergraphs



DataCube : navigation through dimensions



Further works

- Full implementation of hypergraphs in CS framework :
  - impact on clustering
  - impact on layout
- Importance of the quality of data for nice visualisation
- Importance of the clustering algorithms chosen :
  - Louvain algorithm is :
    - fast for a clustering algorithm in $O(n \log n)$,
    - based on Newman's modularity, which refer to a null model
    - also small clusters are structurally hard to detect : small depends on the size of the graph the clustering is made
    - $=>$ connected components detection is a way to surrounding part of this problem
    - problem of the initial ordering
  - $=>$ need of investigating other clustering methods
- Investigating automatic tuning of graphs layout depending on the features of the graph

Talk map

Thank you

Questions ?

Catriel Beeri, Ronald Fagin, David Maier, and Mihalis Yannakakis.
On the desirability of acyclic database schemes.
Journal of the ACM (JACM), 30(3) :479–513, 1983.

Chris Bennett, Jody Ryall, Leo Spalteholz, and Amy Gooch.
The aesthetics of graph visualization.
Computational Aesthetics, 2007 :57–64, 2007.

Claude Berge and Edward Minieka.
Graphs and hypergraphs, volume 7.
North-Holland publishing company Amsterdam, 1973.

Alain Bretto.
Hypergraph theory.
An introduction. Mathematical Engineering. Cham : Springer, 2013.

Ronald Fagin.

Degrees of acyclicity for hypergraphs and relational database schemes.
Journal of the ACM (JACM), 30(3) :514–550, 1983.

Emden R Gansner and Yehuda Koren.
Improved circular layouts.
In *International Symposium on Graph Drawing*, page 386–398. Springer, 2006.

Jiawei Han, Jian Pei, and Micheline Kamber.
Data mining : concepts and techniques.
Elsevier, 2011.

Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jörn Kohlhammer, and Guy Melançon.
Visual analytics : Definition, process, and challenges.
In *Information visualization*, page 154–175. Springer, 2008.

Irvin Rock and Stephen Palmer.
The legacy of gestalt psychology.

Scientific American, December 1990, 1990.

Dominik Sacha, Michael Sedlmair, Leishi Zhang, John Aldo Lee, Daniel Weiskopf, SC North, and DA Keim.
Human-centered machine learning through interactive visualization : Review and open challenges.
In *Proceedings of the 24th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, 2016.

Roberto Tamassia, editor.
Handbook on Graph Drawing and Visualization.
Chapman and Hall/CRC, 2013.

Max Wertheimer.
Uber gestalttheorie.
1925.