

Chapter 3

Applying Visual Analytics to the Atmospheric Chemistry Network

“ I have a notion that when the mind is thinking, it is simply talking to itself, asking questions and answering them. ”

- Socrates, *The collected dialogues of Plato*

Contents

3 Applying Visual Analytics to the Atmospheric Chemistry Network	301
3.1 Introduction	307
3.1.1 Networks and their role within Visual Analytics (VA)	307
3.1.2 Graphs in Chemistry	307
3.1.3 Modeling chemistry as a directed graph.	309
3.2 Graph Design	310
3.2.1 Graph Syntactics	310
3.2.1.1 Criterion Choices	311
3.2.2 Automated Graph Drawing Layouts	313
3.2.2.1 Replication of hand-drawing methods	313
3.2.2.2 Projection Based	314
3.2.2.3 Force-Directed	315
3.2.2.3.1 Force Atlas 2	316
3.2.2.3.2 Yifan Hu	317
3.2.2.3.3 OpenOrd	317
3.2.2.4 Layout Selection	317
3.2.2.4.1 Node Distribution	318
3.2.2.4.2 Node Density	319
3.2.2.4.3 Mathematical Analysis and Layout selection	322
3.2.3 Graph Semantics	322
3.2.3.1 Limitations	323

3.2.3.2	Node Encoding	325
3.2.3.3	Edge Properties	328
3.2.3.3.1	Muti-variate edges	329
3.2.3.3.2	Edge Direction	329
3.2.3.3.3	Edge Shape	330
3.2.3.3.4	Edge Bundling	332
3.2.3.3.5	Power, Routing and Confluence graphs.	333
3.2.3.3.6	Edge Angle / Continuity	335
3.2.4	Temporal Projection	335
3.2.5	Additional Dimensions	337
3.3	A Chemistry case study	337
3.3.1	Syntatic representation	338
3.3.2	Semantic representation	338
3.3.3	Local vs Global minimum	339
3.3.3.1	Approximation of degrees of freedom	339
3.3.4	A model of Beijing	346
3.3.4.1	Trends in the Chemistry	347
3.3.4.1.1	Network shape	347
3.3.4.1.2	Pattern Matching using t-SNE	349
3.4	Summary	350

3.1 Introduction

The previous chapter established the importance of choosing the correct medium and metaphor for the understanding of complex datasets. For relational data exploring the relationship between any set of items, the sociograph style of representation was found to be the most effective. This chapter builds on that analogy and explores how this may be further improved. I start with looking at the current use of sociograms within chemistry (subsection 3.1.2), the methods in which these can be made more useful (subsection 3.2.1, subsection 3.2.3), and finally apply this to a case study representing the atmospheric chemistry within the city of Beijing, China (section 3.3).

3.1.1 Networks and their role within Visual Analytics (VA)

Networks are present everywhere - This ranges from interactions within social media to bank transactions, internet routing, animal migration (nature paper with good vis), epidemiology, transport, everywhere... [Martin Grandjean, 2016, ?, ?, ?]. This is because the sociogram (or graph) structure may be applied to any set of items which contain one or more relationships between them. With academia, networks have been invaluable to the field of gene-activation and metabolic pathways (Biology); *EXAMPLE FIX: suspects and organised crime / terrorist cells (secret service or police) and the complex networks of relationships between system components and users (software engineering) [Dwyer et al., 2006b]*. Within the information visualisation field (InfoVis), items are referred to as nodes, or vertices, and their relationships are called edges, or links [Kerren et al., 2014]. These terms can be used interchangeably and will do, throughout this thesis.

3.1.2 Graphs in Chemistry

Node link representations have been a core part of the field of chemistry for many years. They are integral in the representation of molecules, using the ball-stick (graph) style analogy, both physically (with the aid of molecular model kits) or pictorially to show various structural properties, Figure 3.1. Using such analogies aid in the tasks of identifying the features, functional groups and bond properties of the species and how they can react.

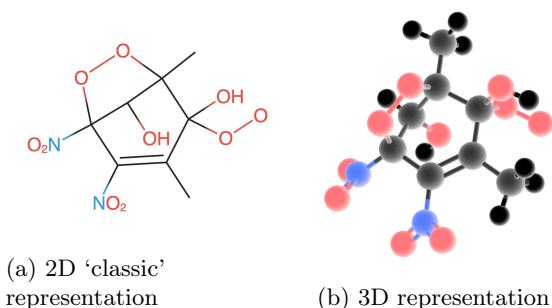


Figure 3.1: An example molecule represented in two and three dimensions. These have been created using the SmilesJS and Mol3D software, [Probst and Reymond, 2018, Bergwerf, 2019]

Within the master chemical mechanism (MCM) [?], referenced in ??, we see such properties used used to determine the reaction pathways species may undergo ?. In mechanistic development, the degradation of larger species is represented as a series of interconnected reactions in the form of a reaction cycle, Figure 3.2. Such patterns often have to be manually simplified to best represent the chemistry of the system.

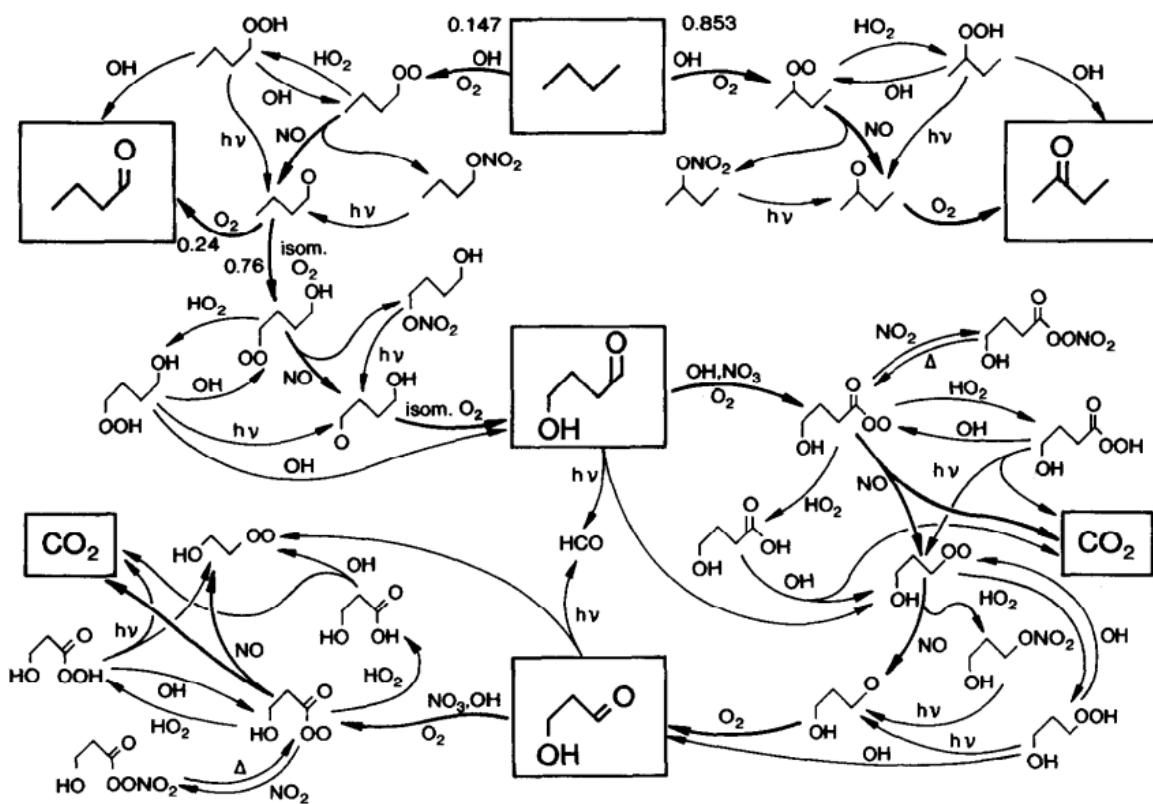


Figure 3.2: A systematic representation of the degregation of butane. Source: [Jenkin et al., 1997]

This type of sociograph has many similarities with a simple directed graph. Links are representative of the direction of reaction and show the relationships between different species (the nodes). The main difference between this and a conventional graph is that species have been duplicated to aid

clarity, a feature which will not occur with the core mechanism, and consequently any mathematical graph which may be generated from it. The flow-like nature makes this type of representation makes it easier to understand as a human, than any list of reactions could ever be. The reason for this lies in a combination of evolutionary traits, whereupon we are genetically predispositioned to comprehend patterns and temperate shapes faster than written characters.

3.1.3 Modeling chemistry as a directed graph.

Historically it is seen that the graph format has proven to be an efficient means of understanding the reactions within a mechanism. Traditionally these are constructed manually, with the designer making a series of choices on how best to place, and simplify the chemistry based on their application. As our understanding of chemistry improves and we begin to progress into automated and semi-automated mechanism construction. This makes the construction of mechanisms with tens of millions of species and billions of reaction possible. It is at this point where the manual design and simplification of reaction networks becomes infeasible.

Today automatic graph layouts allow us to generate multivariate and complex graphs easily. This means that, much like in the construction of a mechanism, we can rely on computer-aided design to generate a directed graph representation of the chemistry. [Montañez, 2016] states that "The beauty of a good information graphic is that it can tell a whole story in a single unit of visual content". This is particularly true for the use of directed graphs in chemistry where we can compare different mechanism subsets,(??) or model simulations (??).

However, as with everything several problems emerge from the complete automation of a task. Firstly real-world data very rarely reacts how it is expected to. Here networks of high edge density often obfuscate the graph data, and produce what is only described as a ‘birds nest’, ‘hairball’ or ‘ball of yarn’ within the literature, [Roberts et al., 2014]. Although such disasters and mishaps can be seen as moments of turbulence, they encourage a greater understanding of the design process and can catalyze to merge dissimilar ideas into an effective visualisation [Johnson, 2010].

(a) Full MCM

(b) Borneo

(c) London

Figure 3.3: **Looking at mechanism subsets using subgraphs.** We compare how the mechanism subsets used to model chemistry within two different environments vary. These are Borneo, red.... and show the ...

3.2 Graph Design

In the graph design process there are two main areas which require consideration - these are syntactic (subsection 3.2.3) and semantic (subsection 3.2.1) representation. In this section I will explore the different automated graph layouts which can be used to represent our data (syntactic) and the methods in which we may increase the net amount of useful information presented to the user once the graph layout has completed (semantic).

3.2.1 Graph Syntactics

Syntactic representation considers how best to distribute the information on the given page for maximum impact. An example of this is seen in Figure 3.4, whereupon the flights between airports are represented using a force-directed graph layout (top), and their geolocation (bottom). The graph provides a clearer representation on the relationships between airports, whilst the geo layout gives a more accurate representation of the distances between unconnected nodes (airports). Decisions on what syntactics best represent the feature of interest are an important part of the visualisation process.

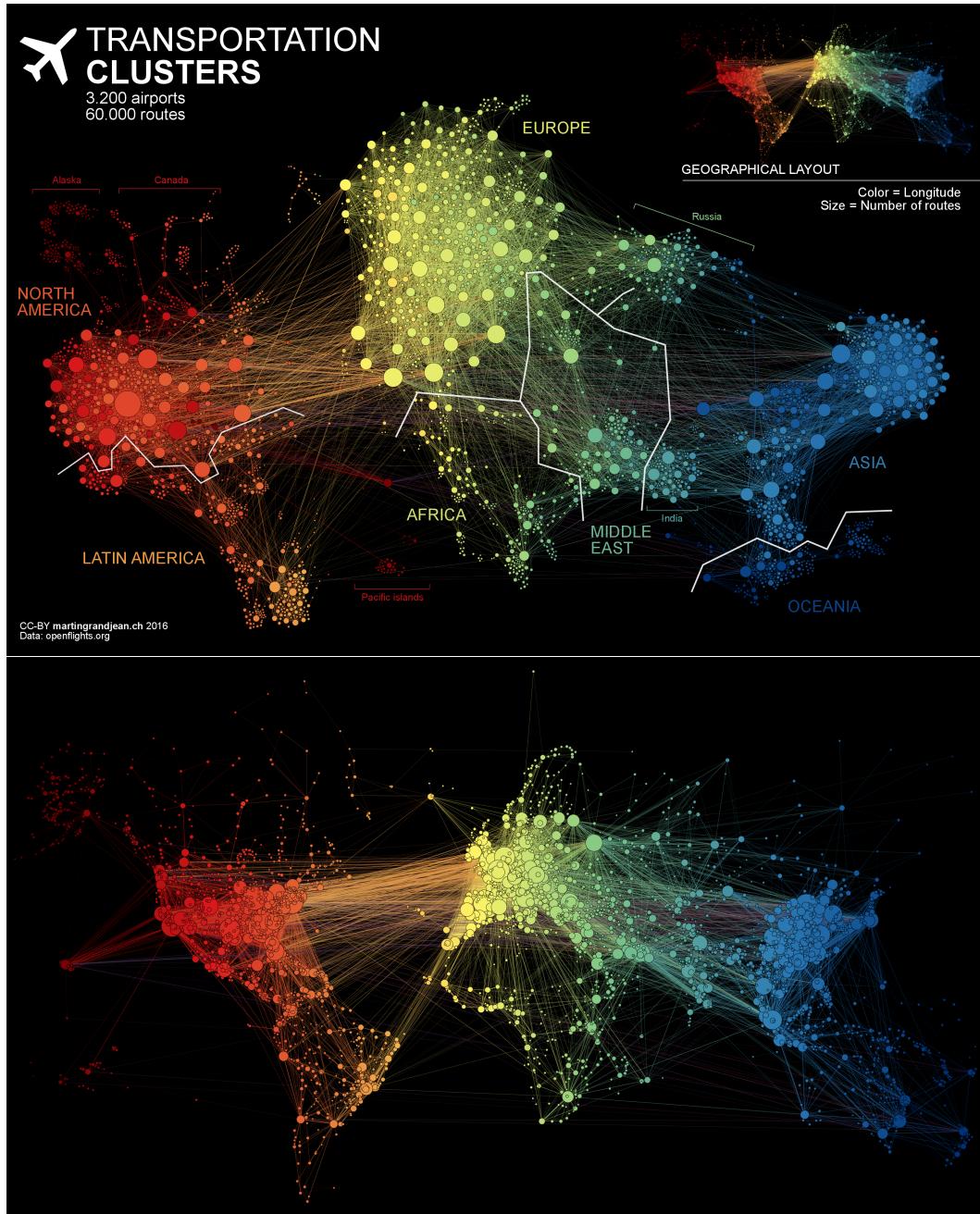


Figure 3.4: Comparison of different representations of flight data by [Martin Grandjean, 2016]. The top figure shows the data represented by a force directed graph layout (described below) and a Geo-layout showing each point at its location on the Earth.

3.2.1.1 Criterion Choices

Since chemical networks can provide a wealth of information, this can prove challenging to both user cognition and physical (computational) resources [Kerren et al., 2014]. There are many metrics for improving a visualisations aesthetics, [Purchase, 2002], however these are often evaluated only for a limited number of criterions. This makes it difficult to accurately quantify the changes in user-readability, especially if these are treated differently than was originally intended, [Pohl et al., 2009].

Edge Crossing and Bends

One of the greatest limitations to understanding a graph is the number of crossing edges [Purchase, 1997]. Experimentation with eye tracking software found that in determining information about the structure of a system users spent more time looking at the edges than the nodes themselves, [Pohl et al., 2009].

Several styles of layout algorithms (e.g. force-directed, orthogonal) aim to reduce these by introducing repulsive forces, or constraining the shape of each edge within the graph. The use of orthogonal designs has been proven for circuit chematics and transport networks, however they lose any information that may be encoded in a nodes relative location. A reduction of the accuracy is also seen in simple tasks where users are required to name the nodes along a chosen path compared to a force directed layout (58/100:92/100), [Pohl et al., 2009]. Finally the use of bends may be misleading as users can perceive them as multiple separate objects [Bennett et al., 2007].

Node Distribution

Having nodes equally distributed across the visualisation area can also increase the readability of a graph. In addition to preventing overlapping edges, equally distributed networks with medium length edges greatly improves the ‘flow’ of the diagram. Coupling this with graphical-symmetry is seen as the next most important user-ranked preference, [Purchase et al., 2003].

Node Overlap

As with overlapping edges, if two nodes are positioned in the same point in space it makes it difficult to read a graph. Historically many algorithms treat nodes as point masses which would make it difficult of designers to remove the overlap between them [Dwyer et al., 2006c].

To remove any node overlap there are two main methods, [Dwyer et al., 2006b]. These are:

1. Create a layout design capable of taking node size (e.g., [Friedrich and Schreiber, 2004]) into consideration. These designs tend to be layout specific and not absolute in removing all overlap between nodes.
2. This requires a level of post processing in the form of a ‘layout adjustment’. Here we reposition nodes after a chosen layout has finished computing. The drawback of this method is that information contained in the graph’s shape may be degraded. This can be done through the use of collision detection, or moving nodes to the centre of the vernouli cells [Lyons, 1992].

3.2.2 Automated Graph Drawing Layouts

In their design and evaluation, automatic graph drawing algorithms are created as to minimise a specific criterion. In this subsection I explore several graph drawing algorithms, and juxtapose them to determine which may be best suited for the application to atmospheric chemistry model representation. To do this a MCM subset representative of the APHH Beijing campaign [?ref?] is used. This provides the foundation for a real-world case study which may be simulated in a chemical model.

3.2.2.1 Replication of hand-drawing methods

With the rise of computation many traditional visualisations were adapted for computer aided generation. Fields of architecture and circuit design adopted computational software to alleviate some of the difficulties presented by large or complex designs. Similar ideas such as the use of automatically generated transit maps can be used to link chronological or topological items such as ideas, [Foo, 2019]. Figure 3.5 shows all the possible paths for the oxidation of methane to produce carbon dioxide (and water). Although such methods can be useful in showing isolated path ways, they provide a convoluted representation of large interconnected systems and require some manual intervention.

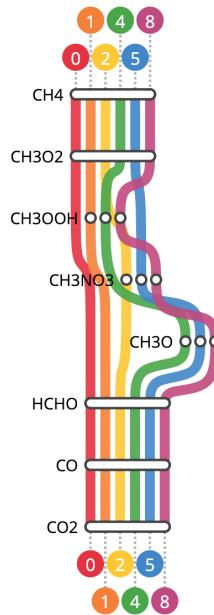


Figure 3.5: **A transit map showing all the possible routes from methane to carbon dioxide.** This was drawn using MemoryMap ([Foo, 2019]) and uses a version of the MCM methane subset, where carbon dioxide has been introduced.

3.2.2.2 Projection Based

One of the oldest fields of data visualisation fall in the realm of cartography. Here the shapes and distances between points on the surface of the earth (an oblate spheroid) are mathematically mapped onto a 1D plane for graphing purposes [Thomas, 1952]. Since the process of dimensionality reduction will produce inherent distortions within the final product, we end up with a range of map projections, with each striving to achieve a different aim, Figure 3.6. The Pierce Quincucial for example is a conformal mapping technique mapping the surface of a sphere to a square with very little deviation in scale and the ability to be tessellated in all directions. The Mercator on the other hand is a cylindrical projection which grew in popularity due to its unique ability to represent any course of constant bearing¹ as a linear segment within the shipping and navigation industry.

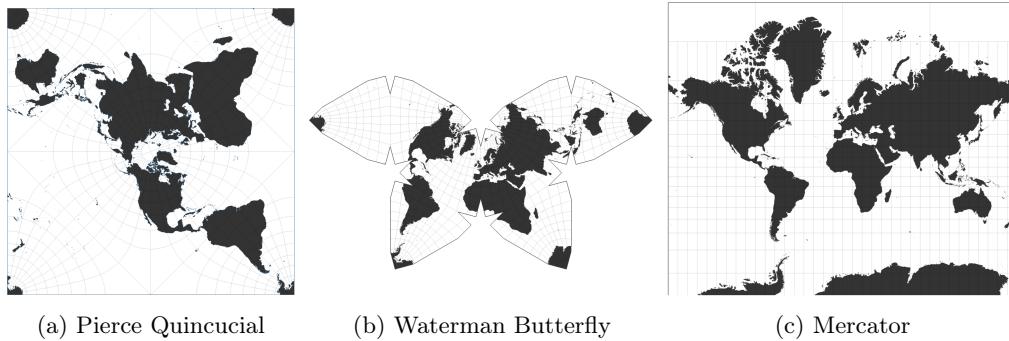
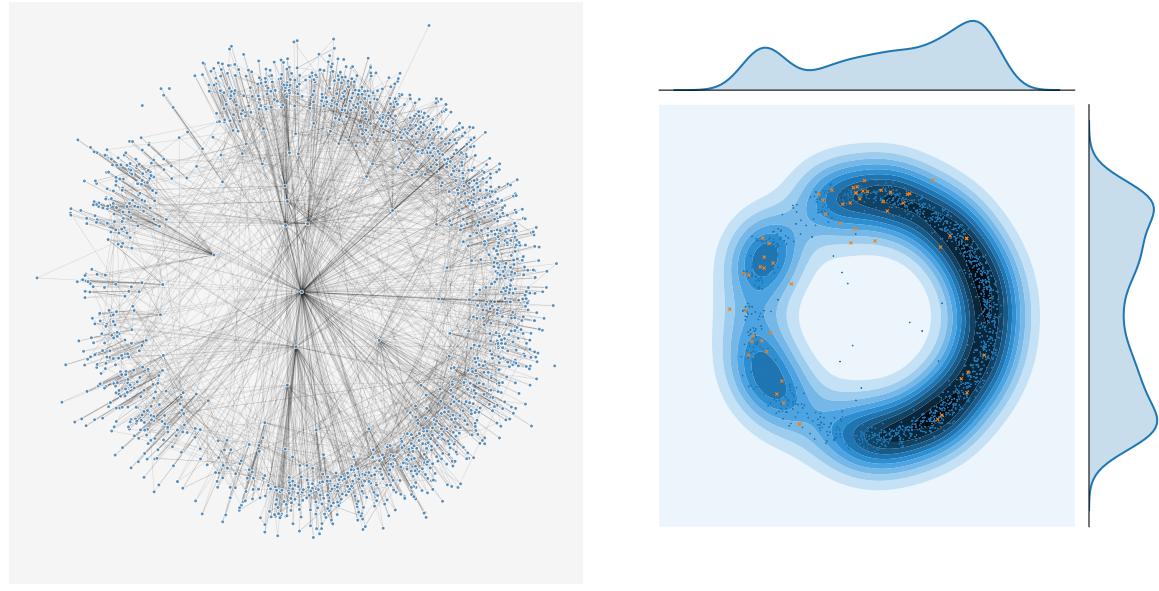


Figure 3.6: **A selection of map projections.** These have been created using DataDrivenDocuments ([?]) and show a range of methods for mapping the spheroid shape of the Earth onto a 2D plane.

More recently, the mathematics of mapping a large dimension onto a simpler one have been applied to the problem of graph representation. [García-Pérez et al., 2019] uses the hyperbolic latent geometry of the Mercator layout to provide a 2D embedding for real-world complex networks. This produces a polar representation of the system, where relationships of like species are of the same angle, with nodes of a high degree being closer to the centre (low r value). This produces a layout, Figure 3.7, where primary emitted species (orange dots) are uniformly (radially) distributed and important (highly connected) nodes close to the centre (small radial component). Although the Mercator embedding does reduce the ‘hairball’ problem experienced by other layouts, it does not take edge weight/direction or self loops. This means that it works well for the representation of the general network layout, but cannot be used for advanced data exploration concerning simulation results.

¹Also known as a ‘rhumb’, or ‘loxodrome’, and consists of an arc crossing all meridians of longitude at the same angle.



(a) The Mercator graph.

(b) Mercator species distribution

Figure 3.7: **The Mercator Projection.** (a) represents output from the mercator graph layout algorithm. (b) provides a kernel density analysis of the node distribution within this.

3.2.2.3 Force-Directed

Force directed graph layouts are the results of the Spring-Electrical model. This was first introduced by [Eades, 1984], and further improved by [Fruchterman and Reingold, 1991]. Force-directed layouts are in essence a simple physics simulation of like-charged particles (or nodes). These particles are repulsed by each other, much like protons experiencing Coulomb repulsion, and try to get as far away from each other. Should a relationship exist between two nodes, we introduce an attractive force between them. These attractive forces act much like springs drawing the nodes back together.

In the case of a weighted graph (where each link (or relationship) has a value associated with it), we can adjust the spring coefficient of the attractive force to reflect this. This results in a layout where strongly connected objects are drawn together, and weakly connected ones further away. Uses for this type of representation have been seen biology, social networks, and with this thesis atmospheric chemistry [Muelder et al., 2014, Kohlbacher et al., 2014].

Barnes Hut Algorithm

Since calculating the attractive/repulsive forces for each node of a large graph can be computationally intensive, many force directed layouts rely on the Barnes-Hut approximation.

This solves the N-body problem of pairwise reactions between nodes, $O(n^2)$, by approximating long-range reactions by grouping such nodes and applying a single action on their centre of mass- reducing the computational time to $n \log n$.

To do this, first a spatial index of each node is constructed (see below). This can either be done using a quadtree (2D) or octree (3D) algorithm. Next the centres of mass are calculated, and finally the repulsive forces for the graph are approximated.

Quadtree Construction: A quadtree is the recursive partitioning of two dimensional space into a set of quadrants (a set of 4 squares). This process is repeated, with each square then being divided into 4 itself, until there is only a single point within a cell². This converts a network, into a hierarchical tree representation of the nested quadrants in which each point resides (a quadtree), Figure 3.8.

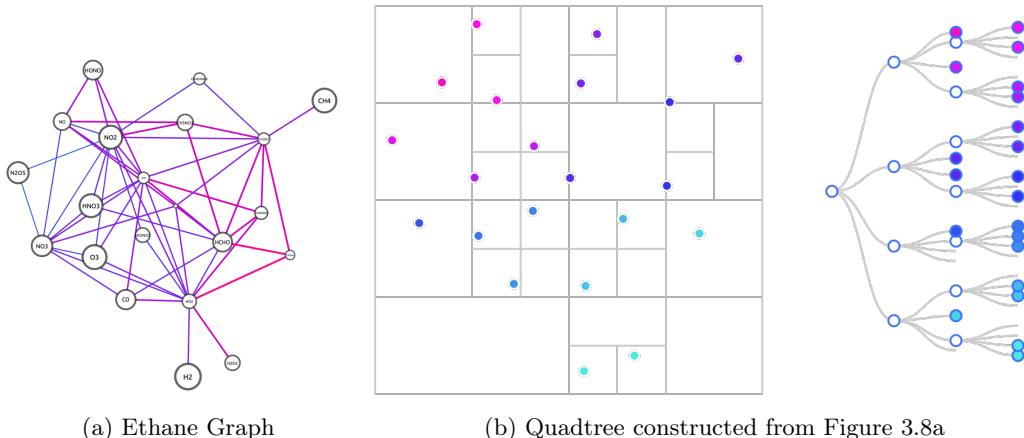


Figure 3.8: **Demonstration of the formation for a quadtree from a force directed graph of Methane (including inorganics).** (a) shows the force directed graph, (b) is the quadtree that has been constructed from this starting with the top left corner and (c) is the hierarchical tree layout of how each nodes classification based on the quadtree.

3.2.2.3.1 Force Atlas 2

The force atlas 2 [Jacomy et al., 2014] algorithm is a force directed layout designed primarily for scale-free³ network spatialisation. It is primarily designed for the use of networks consisting of 10 to 10,000 nodes and uses barns-hut approximation for the calculation of forces. Attractive forces are derived from the spring-electric model, and follow the trend of $F_a = -k.d$, where k is the spring constant and d is the distance between the two nodes. Optional features for the graph include dissuasion by degree (separating nodes with a high number of total links/reactions), logarithmic attraction forces, adjustable gravity (attraction the centre of mass of the system to prevent disconnected components from drifting away) and collision detection to prevent overlapping nodes.

²See ?? for a complete evolution of the methane quadtree.

³Definition in next chapter.

3.2.2.3.2 Yifan Hu

The Yifan Hu graph layout, [Hu, 2004], is a multi-level graph drawing algorithm which uses the barnes-hut algorithm with an octree layout. As with the force atlas algorith, Yifan Hu also has an adaptive cooling aspect to it. This means that as the algorithm is run, its energy is progressively reduced, allowing the system to settle within a low energy state.

The main difference within the algorithm however is the use of the multilevel approach. This has been applied to graph partitioning [11,12,23], matric ordering [24] and the travelling salesman problem [5]. This works by graph coarsening (coalescing neighbouring nodes and weighting them), running the algorithm on the coarse graph, prolongation and then refining the results. This produces an algorithm that runs faster than the Force Atlas, however is constrained to only working on un-directed edges.

3.2.2.3.3 OpenOrd

INSERT SOMETHING HERE

Simulated Annealing

Most iterative layouts are updated interactively from some initial configuration. In most cases this results in a minimum configuration however this is generally a local minimum rather than the desired global minimum [Davidson and Harel, 1996]. To overcome this the work of [Metropolis et al., 1953], which was later formulated in general terms by [Kirkpatrick et al., 1983] was used to lay the foundation for simulated annealing algorithms.

Annealing is usually used to describe the slow cooling applied to liquids for them to reach a crystalline (totally ordered, minimum energy) form. Relating this to the spring-electrical model, it can be seen that if the atoms(nodes) are cooled too rapidly, they will form amorphous structures representing the local minima, as opposed to the desired global one. If cooled slowly, our graph is allowed to find a thermal equilibrium at every temperature. Working from this idea, a slow cooling constant is applied, whilst occasionally supplying the system with short bursts of energy, that may allow it to overcome local minima.

3.2.2.4 Layout Selection

Criteria, such as ability to isolate the shortest path (in our case most reactive flux), are important in determining the usefulness of a graph. Comparing different layouts [Pohl et al., 2009] found 68% of user chosen routes to reflect the shortest path between them. This is due to the force directed layout

placing a greater emphasis on node positions and distance than other layouts. For comparison the same study found this to be 40% for hierarchical layouts, and only 2% for orthogonal ones. In this subsection I look at the use of different graph layouts, and their effect on the user readability of a graph.

3.2.2.4.1 Node Distribution

It is known that in partitioning the screen into quartiles with equal numbers of nodes (homogeneity) greatly improves the usability of a graph and increases symmetry, [Purchase et al., 2003]. The main problem with node-link diagrams is that in representing complex data using an algorithmic layout can often result in regions of dense, indecipherable links, called hairballs, [Ma and Muelter, 2013]. Hairballs obscure nodes and edges within a region, making it impossible to read. Methods such as the pruning of edges, [Dianati, 2016] can be applied to networks as a means of reducing the complexity. This may be applied post computation (syntactic representation), which results in the loss of information, or during the algorithmic approximation in the OpenOrd algorithm, to produce clearer node positioning, with the edges re-introduced at the visualisation level.

In deciding which layout algorithm produces the best graph-node homogeneity, a kernel density approach is used to compare node distributions across 2D space in Figure 3.7b and Figure 3.9. Here small localised areas of higher density, surrounded with sharp changes in density (shown by the contour lines) is preferable. Such a distribution would highlight the modularity of a graph and allow for the distinction between groups of species with many reactions between them, but few in another group. Graphs with a high homogeneity can be determined through the use of x and y kernel density plots. Here a homogenous graph will have a uniform distribution across both axis. However as we also wish to locate regions of chemistry with a high modularity (clustering), a uniformly distributed graph would not suffice. Instead we look for a near-uniform oscillatory distribution with an equal amplitude for each peak and trough. Using these criteria the Mercator (Figure 3.7b), tsNet (Figure 3.9d) and Force Atlas (Figure 3.9b) score the highest, with OpenOrd and Yifan containing a gaussian-like distribution across both axis which is conducive to producing a hairball.

Next we apply prior knowledge about the graph we are trying to visualise. Here we know that the chemistry within a mechanism is determined by the oxidation of a set of primary emitted VOCs. It therefore follows that for an ideal graph layout each primary emitted species should belong to its area of high density, and not entwined within the hairball. Immediately this notion eliminates the Yifan Hu (Figure 3.9a) and Mercator (Figure 3.7b) layouts, since these both contain a high density of primary emitted species (orange crosses) within a single dense region. Using this criterion, the tsNET graph (Figure 3.9d) provides the best representation, followed by the OpenOrd and ForceAtlas layout.

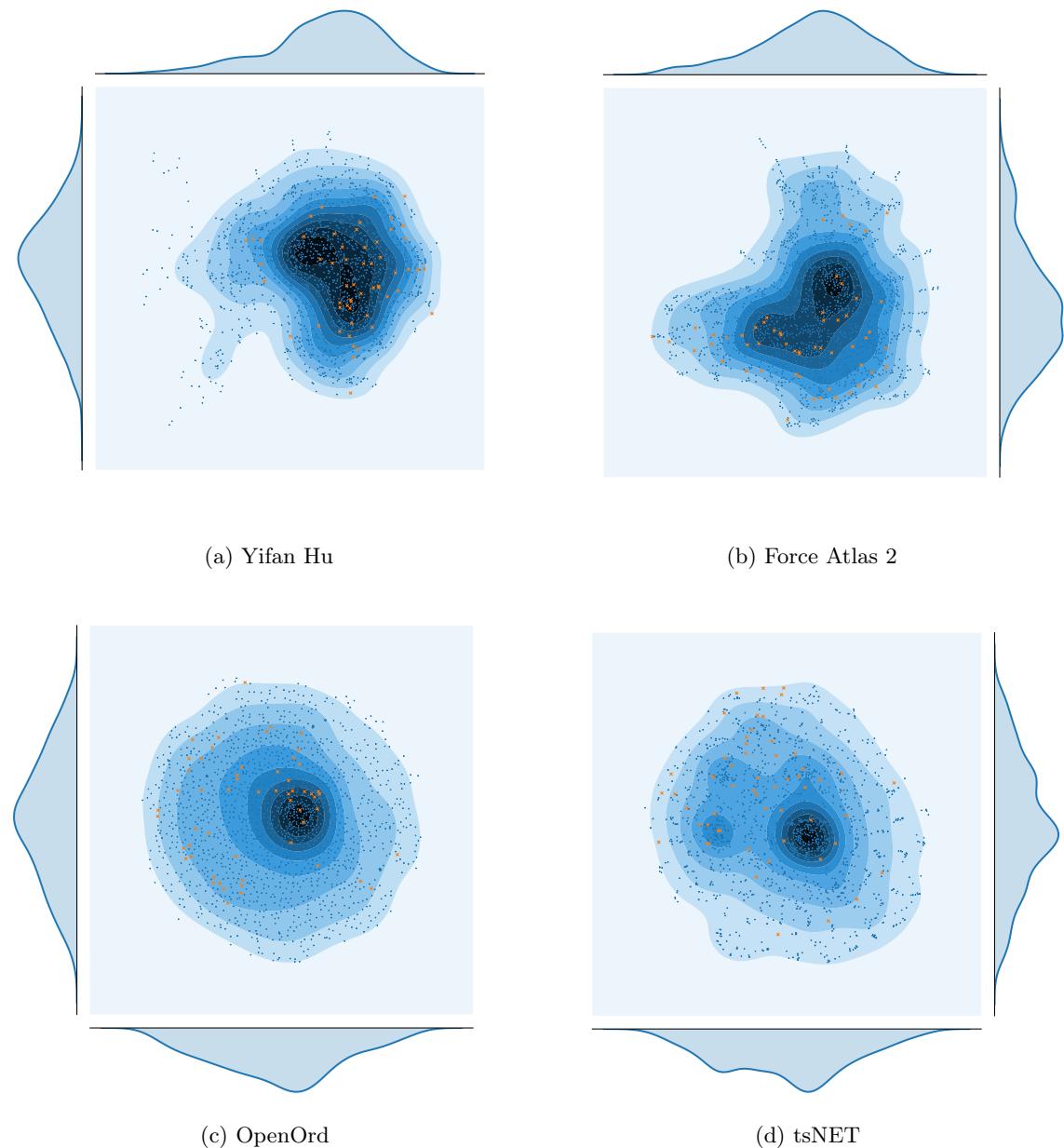


Figure 3.9

3.2.2.4.2 Node Density

Having explored the spatial distribution of nodes within a graph, it is important to determine which layout produces the best node density variation not only across the x and y directions. Here we desire a degree of regular anisotropy to produce ‘clusters’ of densely connected nodes sparsely separated in space. To calculate the distribution between dense and sparsely packed nodes, it is possible to use Voronoi tessellation. Here each node acts as a seed, and the plane is partitioned into a series of n cells, where n is the number of nodes. Each cell or polygon is calculated such that a polygon boundary is determined by all the points which lie closer to its source seed than any other- mathematically this would be defined as the perpendicular bisectors of the lines between all points. The result is somewhat similar

to a box full of bubbles, where each bubble fills the largest area it can before meeting another. Next the area of each polygon is calculated and saved to produce a dataset representative of the complete density distribution between nodes. Here larger areas represents a species with distant neighbours (spatially), and a small one, an area of high density. The method of using vernouli teselation for the calculation of density has been used in the study of neurones, [Duyckaerts and Godefroy, 2000] and areas of fixation when viewing images, [?]. The last part of this process involves colouring the based on the normalised polygon area values and plotted within Figure 3.10. This allows for the easy location of layouts with high isotropy (??,??), which only contain many cells of a similar size, and consequently only exhibit a slight colour gradient difference between points. Although such layouts are spatially efficient, they do not reveal any additional information about the network structure. The colouring can also reveal the the spatial modularity of the graph. Here it is seen that the mercator, despite having a great $x - y$ node distribution, still contains large areas of unoccupied space due to its non linear density distribution. Under this criterion the ForceAtlas and YifanHu layouts (??,??) perform best, with distinct modules of high density appearing to be distributed across the graph.

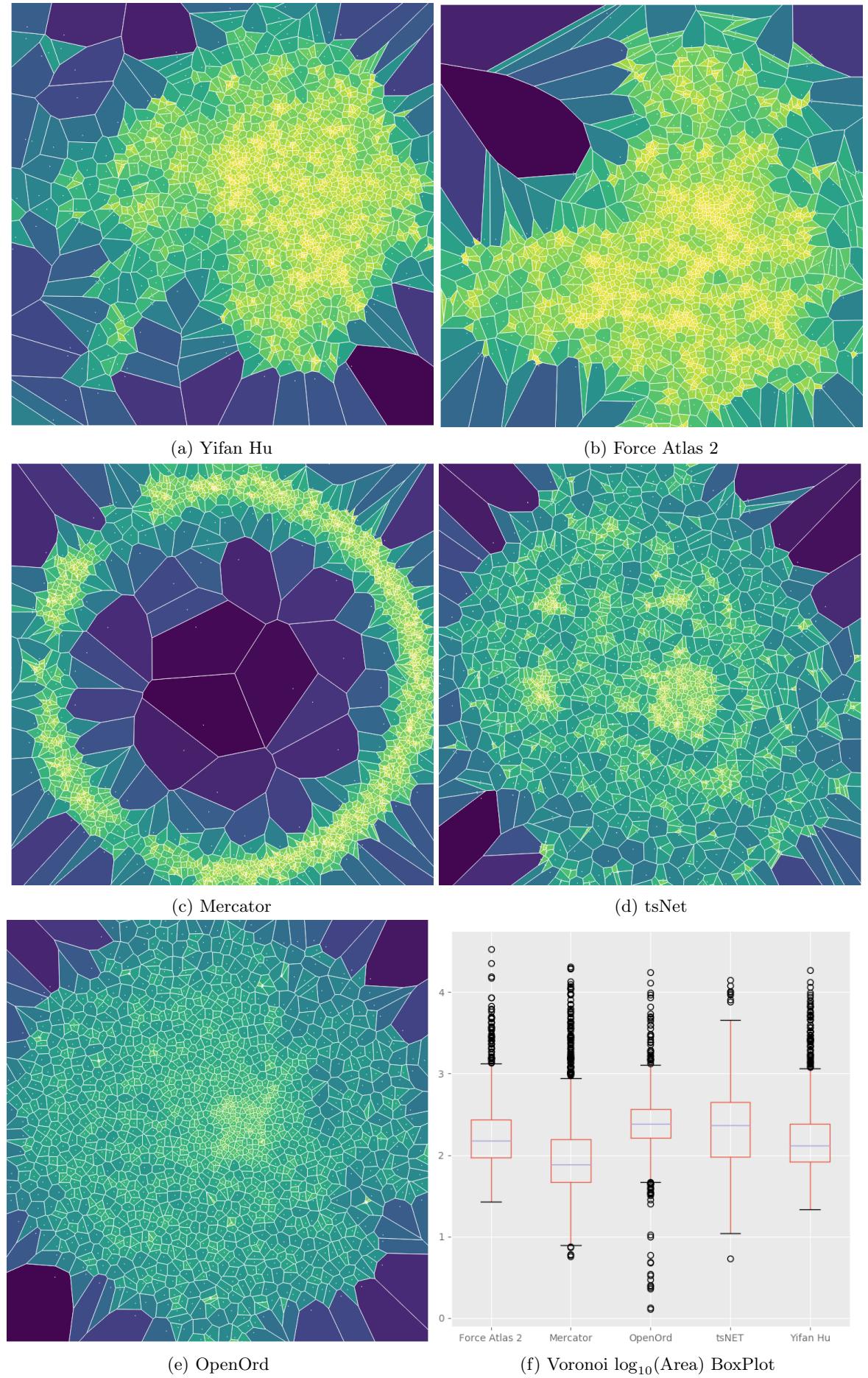


Figure 3.10: Comparing the density of nodes for different layouts using voronoi cell areas.

3.2.2.4.3 Mathematical Analysis and Layout selection

In addition to the qualitative approach through visualisation, it is also possible represent the polygon areas for each layout in the form of several boxplots, ???. The interquartile (IQR) range for each layout represents the range of polygon areas. A large IQR signifies a greater distribution between low and high density areas. In addition to this we are interested in having a greater ratio of smaller area polygons to larger ones. Within the boxplot, this would be represented by having a median which is closer to, or approaching the 25th quartile (the lower box boundary).

Applying these citerions to ??, shows Mercator to provide the best result. However in combining this with our previous observations, is is noted that although the ratios approach our ideal range, its radial shape is not conducive to the general represenation of modularity within a network. The layout with the largest IQR is produced using tsNET algorithm. Although this produces a well distributed algorithm, its inability to handle directed edges and high median rule it out as a possible candidate. The OpenOrd layout can reduce the number of hariballs within a graph through the use of simulated annealing and edgecutting, however it is also this property which in this case has resulted in a homogenous isotropic node distribution (as seen by the small IQR with a large median value). Unfortunately this is not seen as the most effective at hilighting the underlying structure of the chemical mechanism.

This leaves the Force Atlas 2 and Yifan Hu layouts. Out of these the Yfan Hu layout fares better with regards to the box plot, yeilding an overall lower box, with a similar IQR and median ratio. Here its lower median suggests more high density nodes, with a similar distribution to the Force Atlas. This makes sense, since the two algorithms share many similarities, however once again the inability to handle directed edges makes it unsuitable for our application.

This leaves the Force Atlas as the preffered layout for the visualisation of chemical mechanisms. Its directed nature coupled with intuitive design make it applicable and easy to explain, whilst still maintaining an ability to produce a clear representation of any underlying structure. In addition to this, its more uniform spatial distribution (paragraph 3.2.2.4.1) makes it a better candidate than the Yifan Hu graph, which scored the highest in the boxplot test.

3.2.3 Graph Semantics

Deciding the correct semantic representation for a visualisation is often just as important as the selecting the correct syntatic style. Semantic features are often applied post generation [Bennett et al., 2007] and have uses in the encoding of additional information and clarifying any results within the data. As a means of achieving both an aesthetically pleasing outcome, and an easy to understand visualisation,

we must first consider what features we, or the reader, are most interested in. Once this has been decided, we begin to explore various methods for representing them.

3.2.3.1 Limitations

When selecting visualisation semantics, there are several limitations that we must consider.

Visual

When it comes to Visual analytics the greatest bottleneck is due to the resolving power of the eye - this is known as an acutie. Acuities are a measure of the angle of an observed object with the viewer's eye using arcs (one arc equates to $\frac{1}{60}^{\text{th}}$ of a degree). This provides a unit of measurement for the total amount of information density we can feasibly perceive, [Ware, 2013c].

In ophthalmology there exist four types of acuities:

- **detection:** The smallest size an object can be whilst still being seen
- **recognition:** The smallest size an object can be to be recognised
- **resolution:** The smallest distance between two objects before they begin to merge
- **localization:** The smallest amount of visual change that can be measured between two objects

These provide a set of considerations which may be used to assess a visualisation. Depending on what encoding we use, it is possible to improve/hinder the readers ability to percieve information, Figure 3.11. An example of this would be that for a Macbook Pro retina screen⁴, where at 87 pixels/cm⁵ we can display at most 2 million resolvable nodes. If we wished to add links between nodes, the total resolvable items is reduced to one million [Jankun-Kelly et al., 2014].

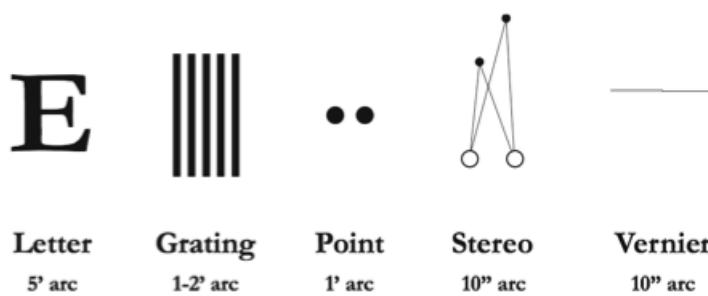


Figure 3.11: Important acuities in visualisation, Source: [Jankun-Kelly et al., 2014, Ware, 2013c]

⁴A retina screen, is half the maximum possible resolution of the human eye at a 30cm distance. Additionally the operating system interpolates in sets of 4 pixels, such that the image displayed may not be at full resolution.

⁵at 57cm from the screen

Cognitive

Although it may be possible to distinguish 1 million nodes and links visually, interpreting and understanding these presents another problem. The visual thinking laboratories, [VTL, 2019], have a range of publications exploring how presentation can improve though, cognition and communication between info-graphic and reader. [Steven Franconeri, 2018], explains that the time required to interpret a visualisation is directly related to the encoding used to hi-light the data within it. Also problems of ‘intentional blindness’ and misinterpretation are problems which are often occurred with poorly thought out encodings.

In considering the cognitive load of a visualisation, [Norman, 2005] provides a list of three categories which should be explored:

1. Firstly we have the visceral level, a subconscious process where decisions are made rapidly based on sensory inputs to the body. This is usually due to our inherent ability to locate patterns and changes due to semantic properties which shift the focus of the user.
2. Next follows the behavioural level (mostly subconscious). These are often learn reaction to changes noted as part of the visceral level. Here reactions may be honed on and influenced by past experiences and events.
3. Finally we reach the reflective level. Here the user collates all sensory input from the previous two levels and makes an informed conclusion about the underlying data. Conclusions drawn here can be used to bias the methods used within the behavioural level in future events.

Technological

In addition to human limitations, there may be restrictions due to the medium a visualisation is created/presented on. In addition to monitor resolution issue earlier, much scientific research is constrained by the size, resolution and colour quality of the presentation mediums used for talks, printing or posters. [Ware, 2013c] explains that a printer capable of producing 1200 dots per inch squared, can only do this for black/white binary images. If for instance 256-greyscale is used, the resulting resolution is then at-least 10 times smaller. This is because printers a Monet style approach to create shading and colour. It therefore follows that at full CYMK, the output resolution will be worse.

It is also important to have a graph fitting the same overall shape of the canvas on which it is presented, [Taylor and Rodgers, 2005]. This not only makes optimal use of any space available, but also reduces the visual complexity as it minimises the number of distinct shapes available to the user.

3.2.3.2 Node Encoding

Within a graph, the nodes represent the set of items we are exploring. Each of these often contain a multitude of features and properties relating directly to them, be it the user details for a retail/fraud network, or the chemical composition and concentration of a species in the MCM. Features of a node describe and additional properties and may be used to determine its interaction with other nodes⁶, [Aumont et al., 2005]. It is for this reason that graph convoluted neural networks, [Klicpera et al., 2018], require a ‘feature matrix’ describing each node, in addition to the network structure and edge weightings. Within a visualisation, a node may be represented in a range of ways.

Circle attributes

The simplest of these range from the use of colour, shape, size, thickness and stroke (outline) to indicate a group. Here it is possible to provide information such as a species concentration based on its size, its importance with its colour, its degree with its opacity and its category with its stroke colour [Ware, 2013b, Ware, 2013a]. Such decisions depend on what properties you are trying to show. For instance red species in Figure 3.13 are primary emitted VOCs, orange species exist between both the MCM and the CRI (see figure caption) mechanism, and blue ones are lumped species which do not appear as part of the MCM.

Chemical Structure

Traditional chemical diagrams use the chemical structure to depict the types of reaction that occur , Figure 3.2. This make it intuitive to extract information about functional group and bond changes within species. Such a method of representation, is indeed useful, however when visualising hundreds, if not thousands of nodes on a page, it results in occlusion, or labels too small to resolve visually.

Species Name

Much like the chemical structure, a species name is proven useful in explaining to the user its chemical properties (often due to prior knowledge, or the ability to look this up). Unfortunately since names have differing lengths, this can cause problems, especially with large numbers of closely located nodes. A solution to this may be to adjust the font size to fit in within the circle radius of the node. However this does come with its problems - for instance very small nodes may have text smaller than a pixel,

⁶This is further explored in Chapter 4

or the misleading notion that longer names are less important, since they are represented by a smaller font.

Interactivity

Ben Shneiderman's famous mantra goes: '*overview first, zoom and filter, details on demand*' [Shneiderman, 1996]. This goes hand in hand with the philosophy used within the design of an interactive visualisation.

For complicated systems, interactivity plays a vital role in unravelling complexity and reducing clutter [Shneiderman, 1997]. It allows the user to actively query only the items that they are interested in whilst still displaying all the information in a single location [Görg et al., 2007].

A comprehensive list of all available interaction types and styles are provided in [Wybrow et al., 2014].

Some examples of interaction are:

Hi-lighting	Visual Structure-Level Interaction
<ul style="list-style-type: none"> • Hovering • Brushing and Linking • Magic Lenses (see hidden objects) 	<ul style="list-style-type: none"> • Selection • Changing layout/mapping attributes • Changing representation
Navigation	Data Level Interactions
<ul style="list-style-type: none"> • Pan / Zoom • View Distortion (fisheye) 	<ul style="list-style-type: none"> • Adding / Filtering • Search / Query

Table 3.1: A selection of interactive methods.



Figure 3.12: **Using mouseover edge-selection to hilight all links related to a node.** This figure shows how in using interactivity it is possible to reduce clutter and filter the information presented by a densely populated graph. In this case the mercator projection (subsubsection 3.2.2.2) is used, with reactions relating to Carbon Monoxide (centre) highlighted. Orange lines represent reactions producing CO whilst the red (some of which may be hidden) are of reactions with CO.

External Labeling

In cases where interactivity is not possible, such as papers, books and this thesis, an alternative approach to data selection has to be employed. Here nodes which are central to the explanation of a certain point are filtered by the author, and displayed through the use of external labels. It is found that having links at 45 and 90 degree angles (such as in transport maps) lead to a clearer layouts and better distinction from the links already within the graph. Automatically generated labels within the thesis are made using [Lu, 2019].

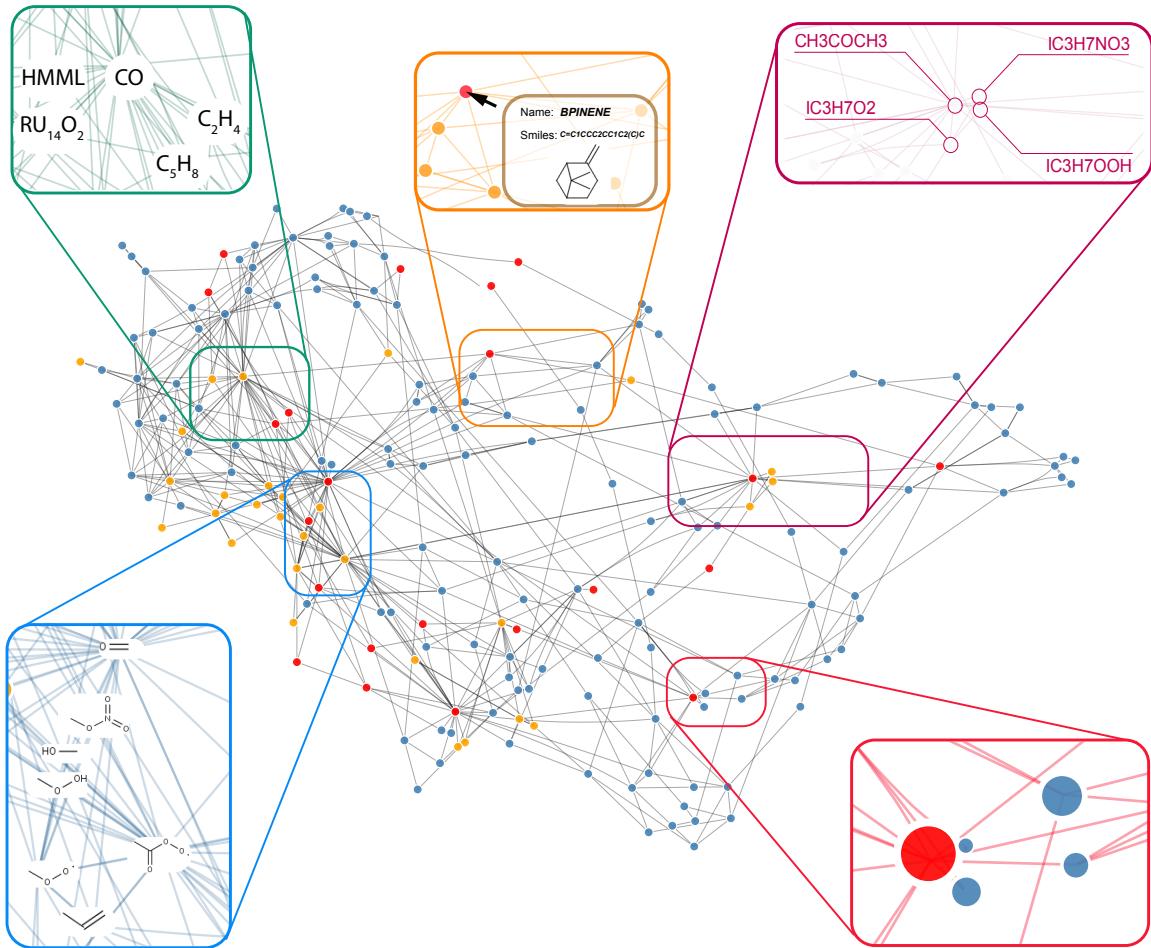


Figure 3.13: **A graph showing 5 different node encoding methods.** These are Circle Atributes (red), Chemical structure (blue), Species Name (green), External Labels (maroon) and interactive selection (orange). The network shows the Common Representative Intermediate species (CITE) mechanism. Node colours represent primary emitted VOCs (red), MCM species (orange) and lumped CRI-only species (blue).

3.2.3.3 Edge Properties

Defining the purpose of graph-energy models as: a means for creating a visualisation from which the viewer can infer properties of the data [Noack, 2004], it can be seen that this criterion is easily met in small and sparse graphs. However non-planar examples with high edge density (lots of links) can easily result in tangled results with impractical running times [Kumar and Garland, 2006]. In most cases attaining an optimal solutions here seems to be computationally infeasible [Davidson and Harel, 1996]. This is generally because graphs primarily focus on highlighting a specific purpose or following a set of aesthetic heuristics [Pohl et al., 2009].

Butane model

3.2.3.3.1 Muti-variate edges

Since there are multiple relationships between species, it is important to decide if simplifying the network would be of benefit. Although it is possible to Figure 3.14.. this may cause unnecessary clutter for larger networks. Instead it is often useful to simplify the graph, and encode the edge properties within the vector object. This allows the user to retrieve any additional information by hovering over the edge or connecting nodes, as required. Should the topic of interest require a specific property, then it would also be possible to remove, or hide, all edges which do not contain it. This produces an interactive graphic containing all the required information, as and when needed, without the unnecessary clutter of having every reaction shown.

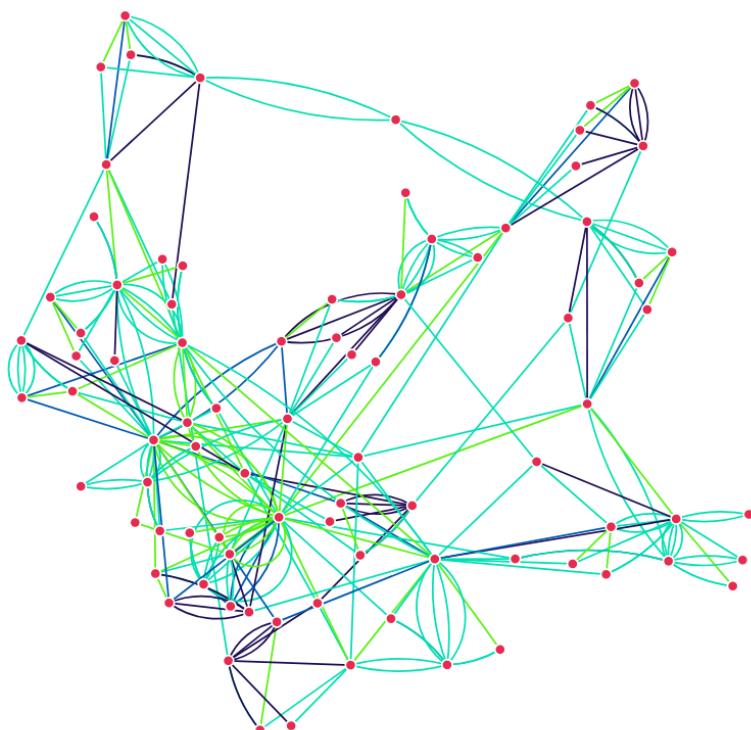


Figure 3.14: **Multiple edges coloured by type of reaction.** Using the overarching categories of reaction type (see fig wuclan) each type of reaction between two nodes can be visualised using the multi-link format. Photolysis (Bright green), Radical/Other (Amazonite / Teal) Decomposition (Honolulu Blue), RO2(Space Cadet / Purple)

3.2.3.3.2 Edge Direction

When using a directional graph it is convention to use arrow heads to represent this. However in high density regions it is often found that arrow heads take up precious real estate in the drawing area [Dwyer et al., 2006a]. As an alternative, colour and line-type can be used to represent the direction instead. This example can be seen in the routing networks presented by [Di Battista et al., 2004]. One example applicable for chemistry would be the use of dashed lines to represent mono-directional

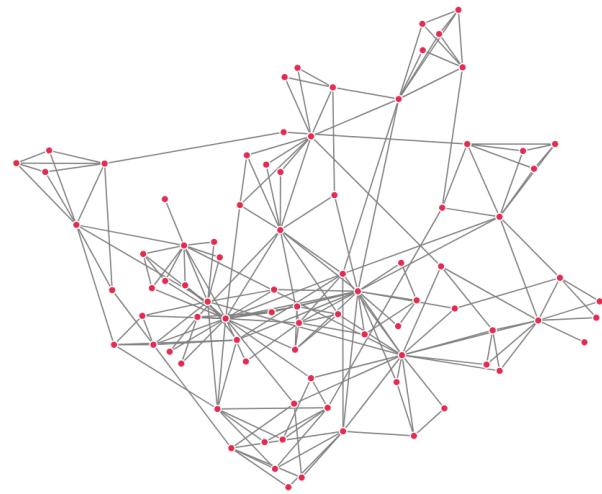
relationships, and continuous lines for bidirectional ones.

3.2.3.3.3 Edge Shape

Edge shape is important, as it is the medium we use to represent relationships within a graph. For orthogonal graphs, poly-line curved edges are used to provide a layout which is simpler and easier to read [Di Battista et al., 1994]. For asymmetric graph drawings circular Lombardi-style curves and cubic brezier lines have been used to reduce the clutter in high edge-density drawings, [Chernobelskiy et al., 2012, Goodrich and Wagner, 1998]. Figure 3.15 shows a selection of different edge types for the Butane MCM subset. The linear network (Figure 3.15a) consists of straight lines between nodes. If a multi-edge graph is required, it is impossible to represent this as all edges between two nodes follow the same path. To improve on this a quadratic arc (Figure 3.15 b) can be used. This presents a symmetric representation where each edge is revealed. Finally bezier curves (described below) can be used to show an asymmetric representation of the multi-edge graph (Figure 3.15c). Both sets of curved representation rely on a set of control points, allowing the designer to control the curve shape, steepness and asymmetry.

Bezier Curves

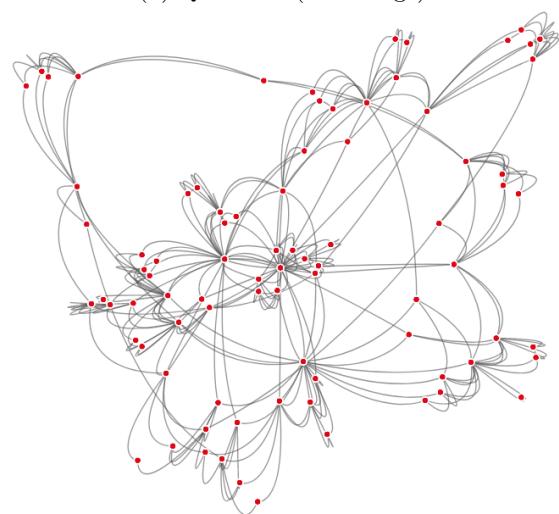
Bezier curves are named after Pierre Bezier who used them in the bodywork design of Renault cars in the 1960s [Hazewinkel, 1997]. Since then they have been widely used in graphics, computer graphics, font design and animation/interactivity response [Goodrich and Wagner, 1998, Hazewinkel, 1997, Mortenson, 1999]. Bezier curves come in a range of possible dimensions, cubic beziers are the most commonly used within network visualisation. These contain four control points respectively which can be used to determine the shallowness of the curve through design. In general relatively shallow curves are preferred, as these do not introduce unnecessary edge crossing or abrupt changes, which have been shown to hinder a user's ability to isolate items of interest, [Purchase et al., 2003].



(a) Linear (single-edge)



(b) Quadratic (multi-edge)



(c) Bezier (multi-edge)

Figure 3.15: A selection of edge shapes for the butane network.

3.2.3.3.4 Edge Bundling

Pioneered by [Holten, 2006], edge bundling techniques are an effective way to reduce visual clutter. Much like a force graph, edges are represented as a string of lined points. This allows for edges to be pulled together (attracted to one another) and produces a visualisation akin to moving water droplets on a hydrophobic surface. Figure 3.21 shows how in changing the amount of attraction between edges, it is possible to reduce clutter in a visualisation.

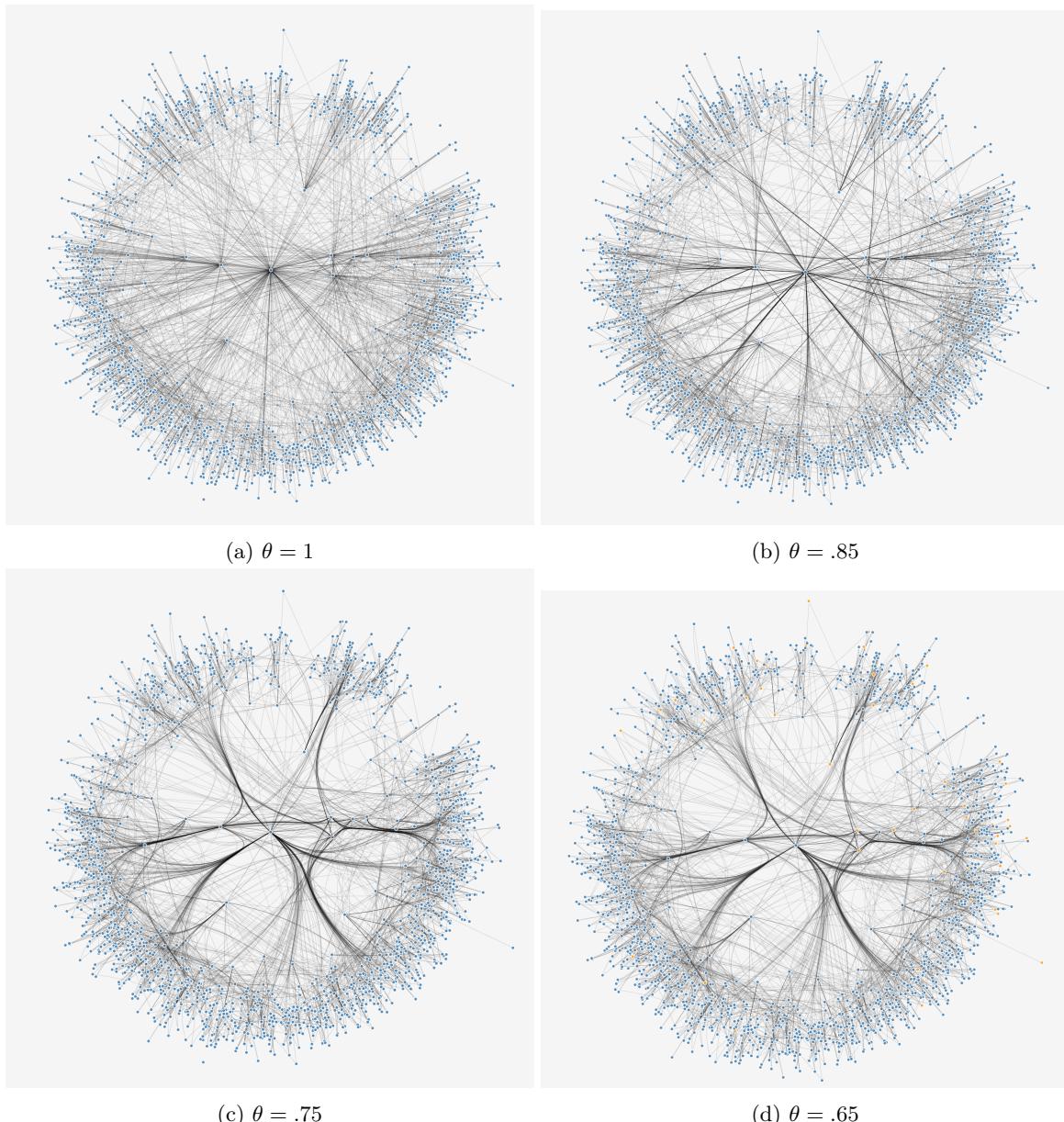


Figure 3.16: **How the compatibility threshold affects edge bundling.** In increasing the amount edges are attracted it is possible to improve the clarity of a graph. However there reaches a point where this distortion can worsen the result, confusing the reader, or creating a false positive. For this reason, I generally use only a slight bundling value > 0.7 .

3.2.3.3.5 Power, Routing and Confluence graphs.

Confluent graphs use a graph drawing method in which edges are not drawn as individual distinguishable geometric objects, but rather as a crossing free system of arcs and junctions. [Förster et al., 2019]. Their design is similar to that of the edge bundling algorithm, except that rather than bundling edges spatially (a design which may introduce ambiguity), the bundling is done based on connectivity and can help reduce clutter by grouping multiple edges where the all target nodes are also connected to all the source nodes, Figure 3.17,[Bach, 2020].

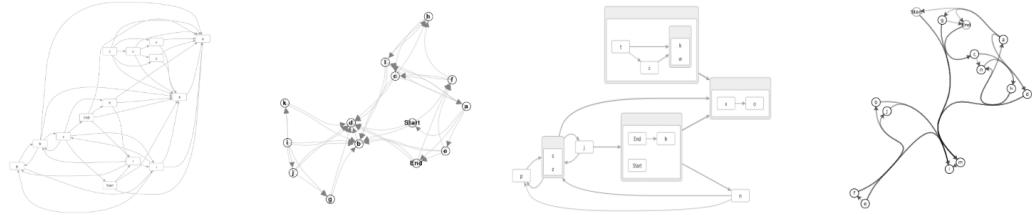


Figure 3.17: **An example of confluent bundling.** From left to right - A traditional network, Edge bundling, Power Graph and Confluent graph representations. Source: [Bach, 2020]

Using butane as an example the construction of a confluent shall be covered. The first step in the process is to create a power graph of our network. Power graphs are a representation of complex networks where sets of items identical source and target links are lumped or grouped within a single item. This is then converted into a routing graph, Figure 3.18. To do this multiple edges which may be bundled have a ‘routing’ node added to guide them. Next basis-splines, using the routing nodes as control points, are used to map the graph⁷, Figure 3.19. Finally crossing links are removed, leaving the confluent graph, Figure 3.20.

Confluent drawings have been found to have many applications (e.g. the ego-centric author network and social interaction graph), they generally perform best in sparse networks with locally dense clusters of a tree like structure [Bach et al., 2017]. Although sparse, the cyclic nature of atmospheric chemistry does not allow for a sufficient reduction in complexity to make them a suitable improvement over traditional graphs. The use of very close fitting basis-splines in addition to a routing graph (confluent graph with crossing artifacts), may however help to simplify specific layouts or mechanism subsets with a certain amount of tweaking.

⁷These are similar to bezier curves but require a degree, p , $n + 1$ control points, and a knot vector of $m + 1$ points. Note: Knots are the things that make the curve continuous

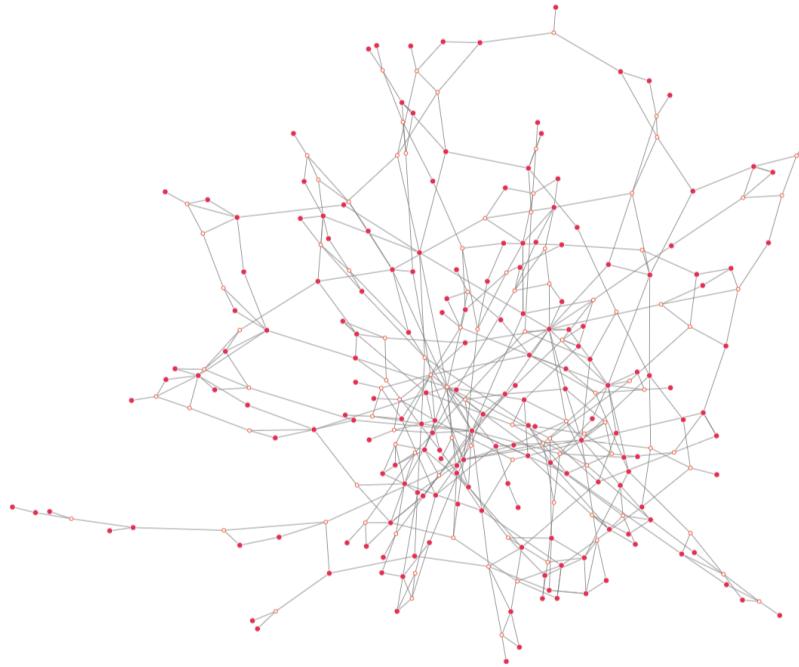


Figure 3.18: **The routing graph of the butane mechanism.** Here paths which contain two or more bundles have an extra ‘routing’ node introduced (orange stroke)



Figure 3.19: **Confluent graph with crossing artifacts.** The routing graph with the addition of basis-splines using the orange routing nodes in Figure 3.18 as control points.

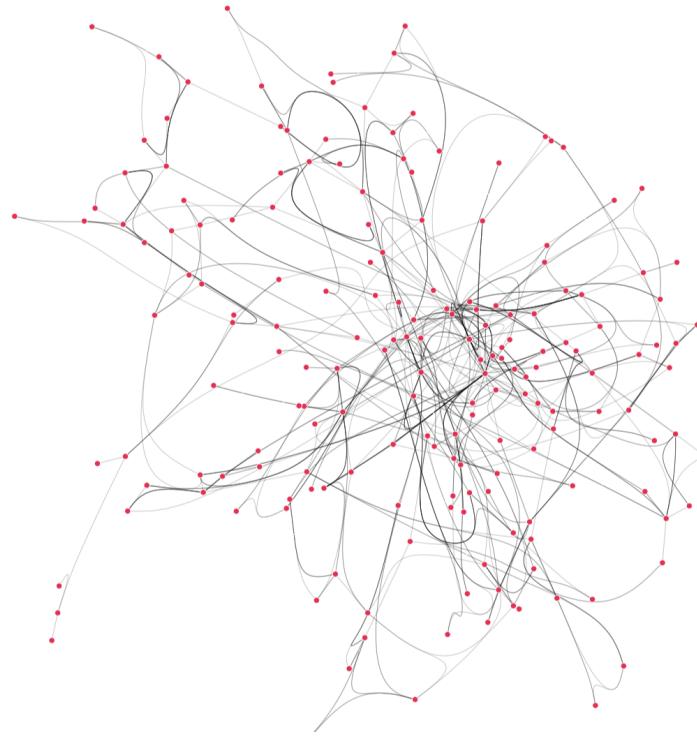


Figure 3.20: **Confluent graphs without crossing artifacts.** The remaining confluent graph with crossing edges removed.

3.2.3.3.6 Edge Angle / Continuity

Visual representation utilises our conscious and unconscious pattern recognition and intuition abilities [Dixon, 2012]. To avoid apophenia (finding patterns where they do not exist), careful consideration has to be placed in the design of a graph layout. Although edge crossing is often thought of as the most important aesthetic metric, finding a continuity between inward and outbound edges of a node was found to be of equal importance [Ware et al., 2002].

Reducing the angle between related edges increases readability and allows the behavioural process to correctly infer information about a graph. This process can be compared to predicting the direction of turbulent vs laminar flow. In addition to this edges should be spaced evenly around node, maximising the minimum-edge-angle between all edges of a node [Bennett et al., 2007].

3.2.4 Temporal Projection

Story-telling has been an effective method to convey information, experience and cultural values for almost as long as people have been around. Many real-life physical processes occur over time and thus allow the use of a story-telling analogy. [Gershon and Page, 2001] provides a generic structure which begins with creating a general overview of the subject. Events are then animated in order of occurrence and defined as we go along. Finally any remaining conflicts and uncertainty is addressed, and these

are rectified. Using this as a template for our graphs, we find that the content is usually given in the form of a title or figure description, the evolution as the visualisation, and finally the reflection and resolution through the use of user interaction (e.g., node hi-lighting, zoom or animation).

Since very few graph layouts support dynamic time-varying graphs [Kumar and Garland, 2006], several methods of visualising temporal events have been developed. Although storylines can be useful for drawing the evolution of simple systems, these break down when dealing with large numbers of dependant variables. Force-directed layouts may be adapted, to suit these better, whereupon the initial positions of the previous node endpoints are used as the initial positions for consequential simulations.

Three methods of representing these are shown in Figure ??.

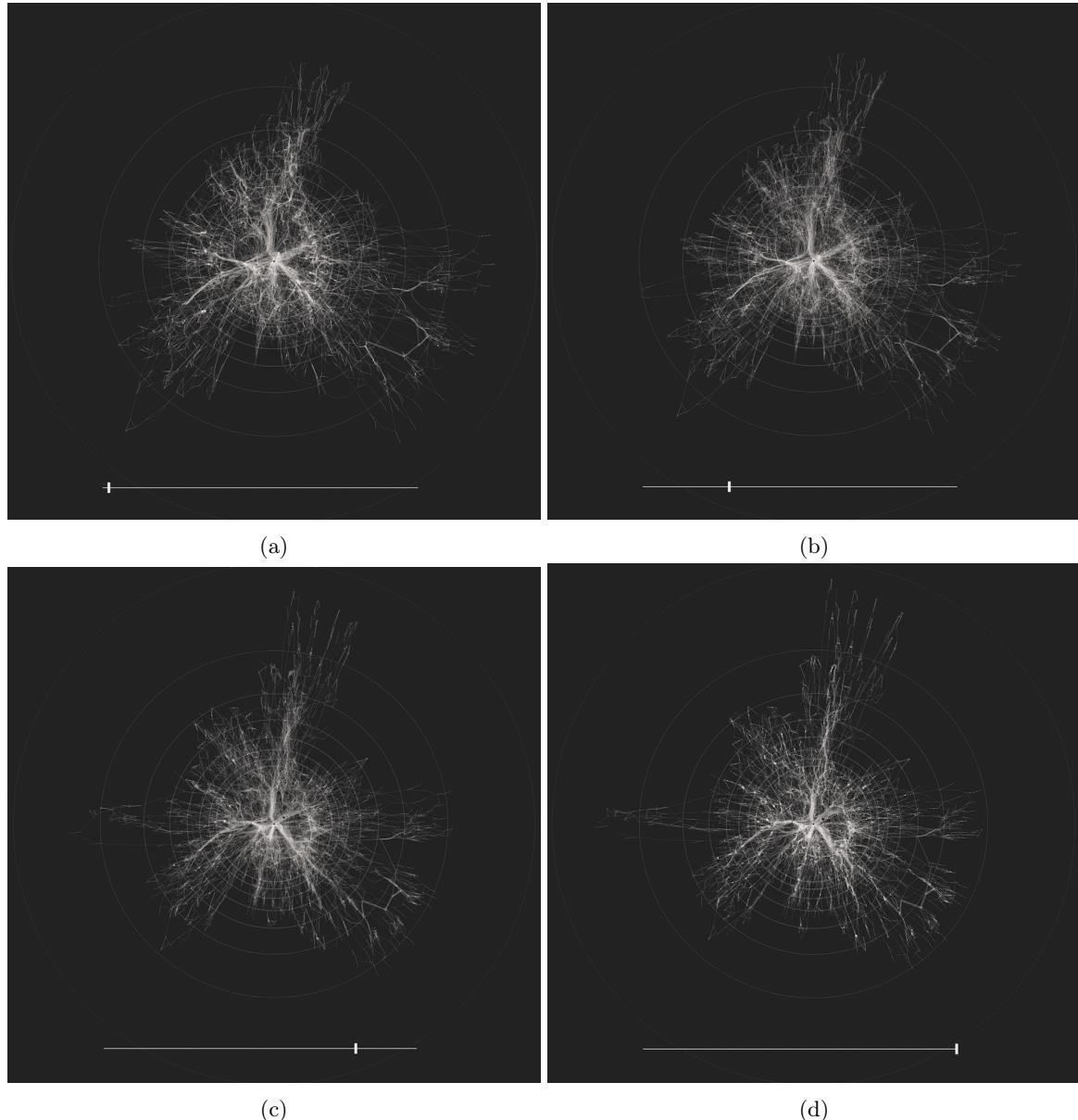


Figure 3.21: Film style representation of temporal changes in a network. Showing the temporal changes from a model simulation of the beijing atmosphere. (a) shows a weighted graph at midnight. With the addition of daylight, the chemistry speeds up causing the force graph to contract, changing the overall network shape (the faster reactions have a stronger attractive force).

Finally, user-interaction such as hi-lighting key nodes/links, zoom and animation⁸ may be used to clarify information at the reflection stage.

3.2.5 Additional Dimensions

Additional dimensions can be used to emphasise certain aspects of our graphs. For instance multiple layers may be used in a directional graph to separate the importance of the nodes [Dwyer et al., 2006b]. ?? shows the first, second and third generation species of a mechanism containing isoprene in three dimensions. Such a visualisation may be explored interactively, with the aid of a computational input device (a mouse, keyboard or device gyroscope), or with the aid of red-cyan 3D glasses (for non-interactive mediums such as print).

Different layers can be used to separate primary VOCs, from species which result in their production (+1 layers) and loss (-1 layers). Temporal data, such as that in ?? can also be presented in this format. The only drawback is the high possibility of obfuscation which may result from many layers of overlapping information.

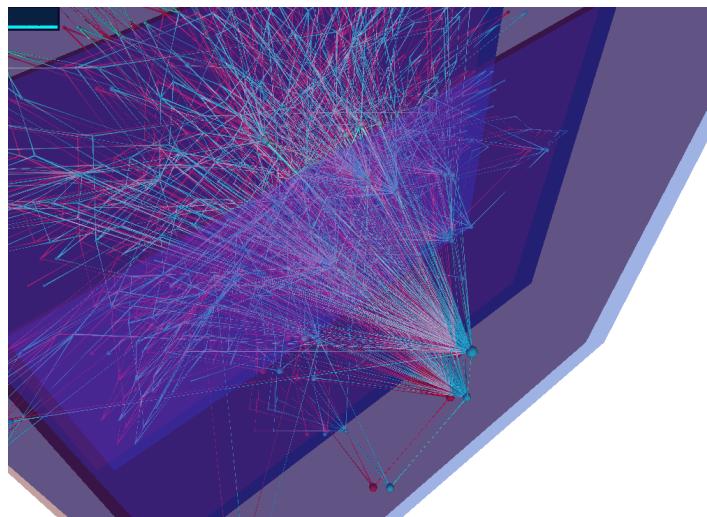


Figure 3.22: **A 3D representation of a graph to hilight certain features.** The first, second and third generation species of isoprene shown as an interactive 3D anaglyph.

3.3 A Chemistry case study

To conclude we apply many of the tools described above to a simple case study. We select a MCM subset containing methane as the only primary emitted species, and run it through the Dynamically simple model of atmospheric chemical complexity (DSMACC) [ref] using the initial conditions of XYZ.

⁸[Archambault et al., 2014] notes that animation poses high demands on the users visual memory, and that snapshots are likely to miss underlying patterns. For this reason an interactive techniques that can allow retrospective selection of timesteps allows for a good compromise between these.

We run this forwards to steady state and extract the flux between species on noon. The edge weight is the net flux (product of the species concentration * the rate of reaction for all reactions), normalised to a value between 1 and zero.

This allows a simplified view of the different properties which affect the visualisation of the graph produced.

3.3.1 Syntactic representation

Since we shall be using simulation data, we require a layout which deals with both direction and edge weights. In the spirit of zero and Protagoras⁹, we opt of the spring-like description presented by the Force Atlas 2 algorithm. This feature hi-lights fast reactions by bringing nodes together. Such a property has been observed to help users select the shortest path within a network, [Pohl et al., 2009]. Here users picked the shortest path an average of 68% for force directed graphs, compared to 40% for hierarchical and 2% for orthogonal layouts. Such properties can help us locate any trends in fast reactions which may control the chemistry within a system.

3.3.2 Semantic representation

Since the graph presented contains only a handful of species, our screen real-estate allows the listing of names for each node. Node sizes are scaled to represent the concentration of each species at that time point, and edges are coloured to represent the strength of each relationship between them.

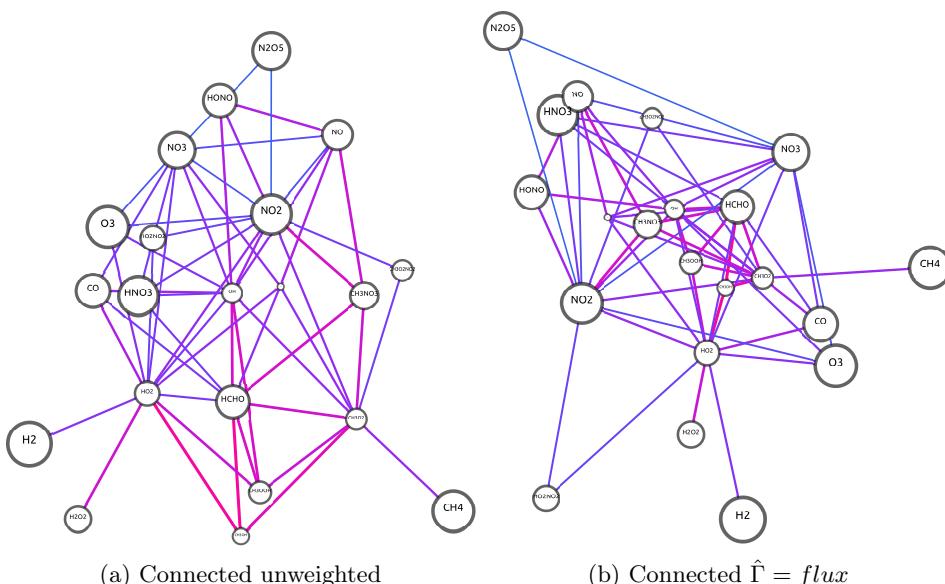


Figure 3.23: Basic steps within in the the graph production process

⁹Famous for the phrase ‘man is the measure of all things’ suggesting that we are constrained by our experiences

3.3.3 Local vs Global minimum

Since computers are deterministic, so are the calculations they perform¹⁰. In starting nodes from the same physical location and exerting the same force upon them, a force directed simulation will result in the nodes settling in the same locations. Since we use this ‘final’ resting as the basis of our visual analysis, it is important to find out if it represents the global minimum of the graph, or is merely a combination of the layout parameters and random seed.

To do this we take a selection of 360 randomly initiated simulations, applying a monte carlo approach to simulation parameters and initial node placement. The force directed algorithm is then allowed to run with progressively decreasing energy until it reaches a stable equilibrium. Once this has been done we

To ensure a global minimum has been met, a monte carlo approach with initial conditions is employed. Since a graph may hold both symmetrical and rotational symmetry, it is necessary to constrain the number of degrees of freedom it contains. The simplest way to do this would be to fix several nodes, and allow the force graph to evolve around this. Here we select the NO_x species, $\{\text{NO}, \text{NO}_2, \text{NO}_3\}$, since their links form a triangle, and in doing so we can eliminate the rotational component of the algorithm.

From here we allow the force directed algorithm to run until the nodes have settled, and record their positions. Once this has been done, all non-fixed nodes are given a random position and this whole process is repeated.

3.3.3.1 Approximation of degrees of freedom

Nodes with no connections can exist anywhere. Nodes which only have one connection, can only exist within a circle from the node they are bound to. If a node is connected to two other nodes, it must therefore exist somewhere between them. This means that in general the more links a node has (its degree), the more constrained its location within the graph layout.

¹⁰Within a certain precision...

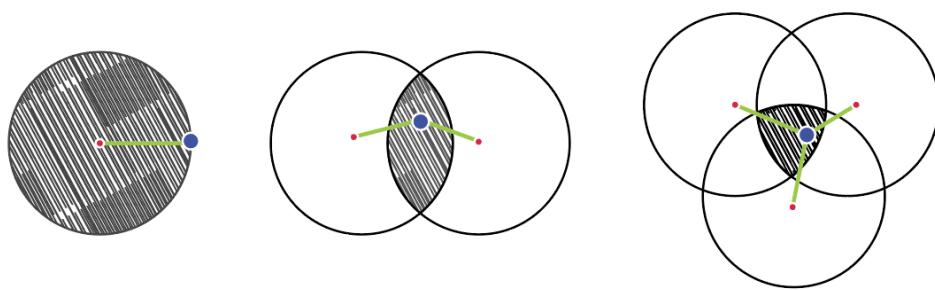


Figure 3.24: A demonstration on how the area / degrees of freedom (grey hashed area) of a node (blue) are limited by the number of neighbouring nodes it is anchored to (red)

To test this we compare node degree with the dispersion between nodes in the above setup of the methane model. However since graphs are multi-link systems, the number of local minima for a specific node is not limited to direct links to its neighbours, their neighbours and so forth. To take this into account we devise an *approximate* metric - outlined below.

A metric of establishing how mobile a node is within the force graph layout

As with all mathematical approximations we begin using a set of axioms. For this example we assume that:

1. We have a set of fixed species, which we use to constrain the rotational symmetry of the graph with. In this case they are the NO_x species.
2. A nodes ‘freedom’ is mostly determined through the shortest path to all fixed species.
3. We calculate the freedom by propagating the fraction of this constraint based on the number of links for each node, and propagating this forwards through the system
4. This metric is used as an effective indicator and not a complete calculation (where all possible paths should be used)

A worked example

To demonstrate this example we take an example shortest path (A-B-C-D) where A is fixed. Here B has 2 links, C has 3 and D has 4. The calculation of the freedom of D therefore equates to:

$$A_{dof} = 1 \text{ (A fixed node)}$$

$$B_{dof} = A_{dof}/2 = 1/2$$

$$C_{dof} = B_{dof}/3 = 1/6$$

$$D_{dof} = C_{dof}/4 = 1/24$$

(3.1)

Finally inverse the values $(1 - D_{dof})$ - such that the greater the output, the more freedom a node has in its final destination.

Plotting correlation

Using a setup of the NO_{1-3} species (the red nodes: Figure 3.25a) constrained in a triangle we run 1200 randomly initiated force directed graphs for the methane network and map the local minima obtained by each node at the end of each one.

We then use this to calculate the deviation between each of these resting positions, their degree of freedom metric output and the node degree for each one. In looking only at the nodes connected to at-least one of the three fixed species, Figure 3.25b, we see a pattern similar to that of Figure 3.24. Here the species CO, $CH_3O_2NO_2$, HO_2NO_2 and CH_3O_2 all have a single link to a fixed species, and can disperse themselves evenly around these. Species with links to two fixed species, N_2O_5 , HONO, O_3 , CH_3O and CH_3NO_3 , are bulled between the two NO_x species that they react with. Finally we have the hydroxide radical which reacts with all of the fixed species and is located at the centre of the triangle¹¹.

¹¹As a note to clarify: although CH_3O_2 reacts with all three fixed groups, the construction methodology for this specific graph only take reactant-product pairs into account. As all of these are reactant-reactant reactions, they are not included within the graph.

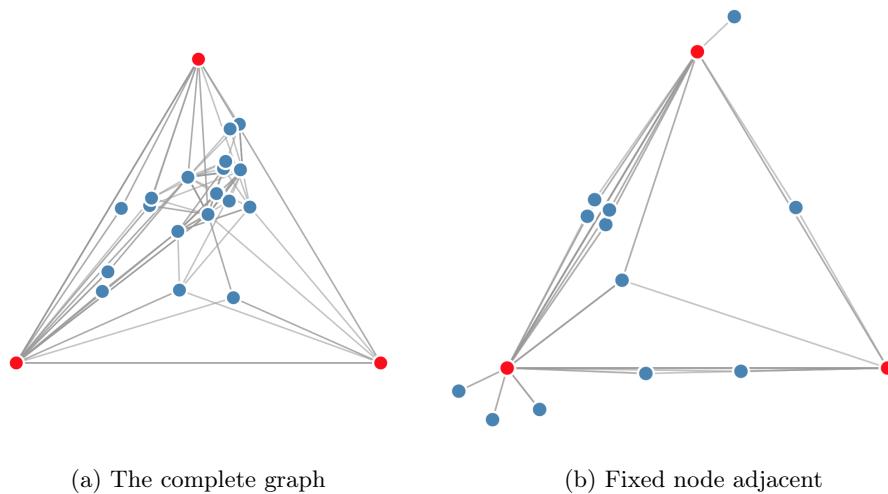


Figure 3.25: Showing the experimental force graph setup, with NO, NO₂ and NO₃ (red) fixed in position (3.25a). This

To plot the correlation we use a Force Directed Representation of Multivariate Data (FRMD), written to show relationships between variables for data [Ellis, 2016]. This takes multivariate data and plots it in a inverse, fully connected graph¹². Figure 3.26 shows the a three dimensional comparison between a species deviation in resting position, degree and metric value. Here nodes are connected to each category with a series of fixed ‘rods’ or links. The length of each rod is adjusted such that it reflects the normalised value of its influence due to its origin category. This works by effectively ‘pushing’ the node towards its category name. This allows us to partition the data into groups with similar properties, or extremes - for example CH₃O₂, HO₂ and OH all have high degree values (>10 links) and low deviation in their local minimia locations. H₂, CH₃OH, CH₃OOH on the other hand have a wide range of possible local minima (deviation), which may be attributed to their low degree values. Formaldehyde on the other hand has both a high location deviation, as well as degree in general. These results suggest that for the higher metric values (small node size and near the bottom due to the 1-metric data set) the results are split between highly constrained species (due to a large number of links and a small deviation) and those which have many local minimia, and few links.

¹²This is similar to the t-SNE dimensionality reduction technique [REF SECT]

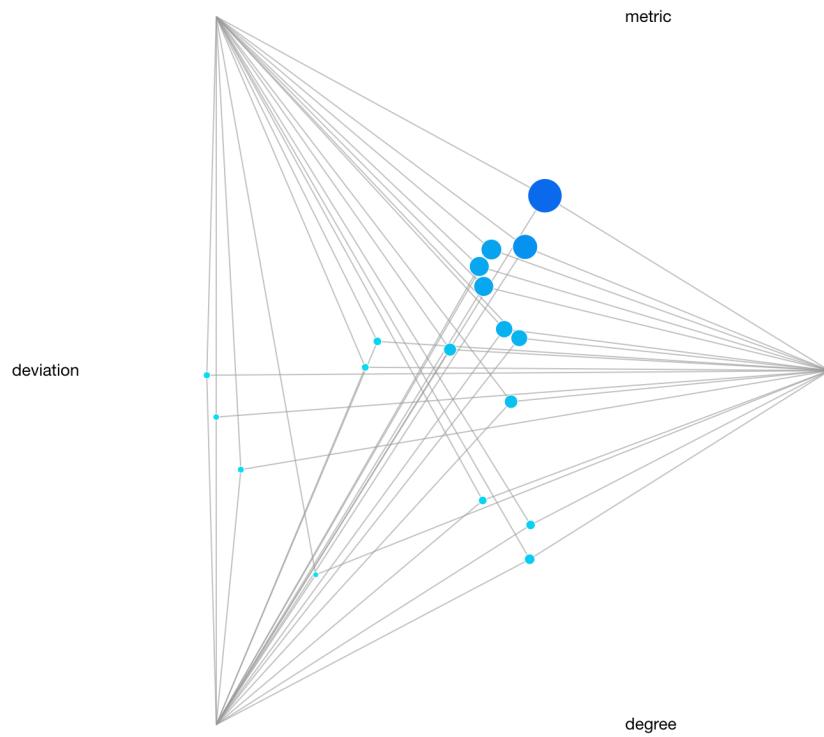


Figure 3.26: A plot showing the relationships between all three categories, and the links between them. Here we may identify groups of similar relationships, with the nodes close to a title having a high value in that feature, and ones near the middle containing a high value for all variables. **Values for the metric are 1-metric for this example.**

We can also reduce the number of dimensions of the FRMD visualisation from 3 to two. In taking only two features and encoding the third through either size or colour we can sometimes present a clearer correlation between items. Within these visualisations we have imbalanced nodes on the horizontal, with higher values pushing the node towards its respective label. In cases where the relationship is approximately equal with both features the nodes exist on the vertical plane between each feature. Here low ranked equality lies close to the center line. Higher correlated matches are result in rods longer than the horizontal distance between both features. This causes a vertical displacement separating the nodes from the horizontal.

Figure 3.27 shows the relationship between the location deviation and a nodes degree. Here, as with the 3 dimensional plot, we have a group of species which have a low degree coupled with a high location deviation, and a set of highly constrained species with >10 links. Similarly HCHO, having a high number in both, is pushed away from the horizontal. As the metric value is encoded in the colour, we see that this has a stronger correlation with degree than deviation, which is expected since this plays a central role in the calculation of the freedom metric.

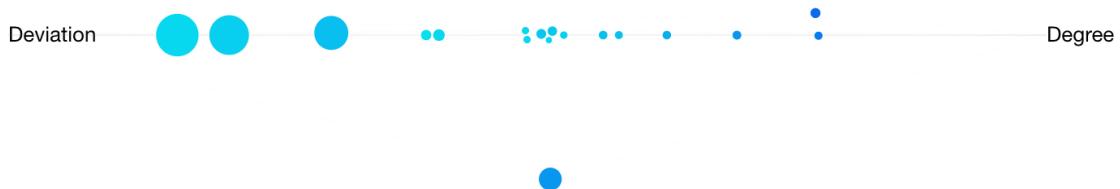


Figure 3.27: Comparing deviations in node location with the number of links that species has within the reaction graph (degree) using a 2D version of the FRMD software. A node's importance with respect to the metric is embedded within the nodes colour and size.

Having seen how node degree fares against the actual deviation of each node, it is time to compare this against our metric. In Figure 3.28 we see a much narrower band of nodes along the horizontal. This suggests a greater correlation between the two values - the ranking within each category are of a similar magnitude.

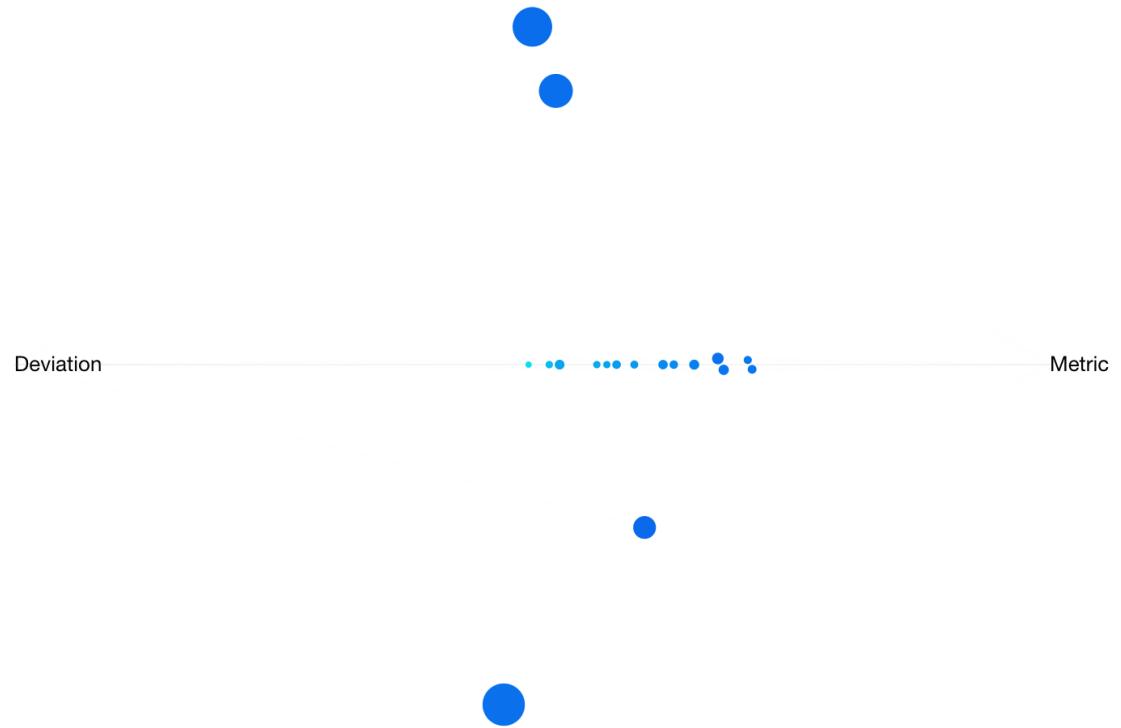


Figure 3.28: Exploring how node deviation varies with the freedom metric above. Node colour and size represent the degree of each node, and are represented using the FRMD model.

3.3.4 A model of Beijing

Using a spun up model initiated from the campaign results XX Beijing (Where did I get these?) we compare the distribution of links within a model. In f[FIG XX] we see the graph shape change due to the presence of photons.

To perform a sensitivity study on the initial positions of nodes within the force atlas algorithm, a graph consisting of links and weightings is constructed using a box model simulation of the Beijing summer environment at mid-day and feed it the gephi software, [Bastian et al., 2009] - an open source software designed for the exploration of networks. We then script the java code to perform the functions in Figure 3.29. As part of this, nodes are initiated with a random position, the force atlas 2 layout is then run and then the graph is rotated and translated such that it is centred around carbon monoxide and has a 45 degree angle between this and formaldehyde. This step constrains the general orientation of the graph, allowing us to analyse the generated graphs for global and local minima. The final step is to save a copy of the generated graph layout and repeat to generate a data set, a subset of which is shown in Figure 3.30

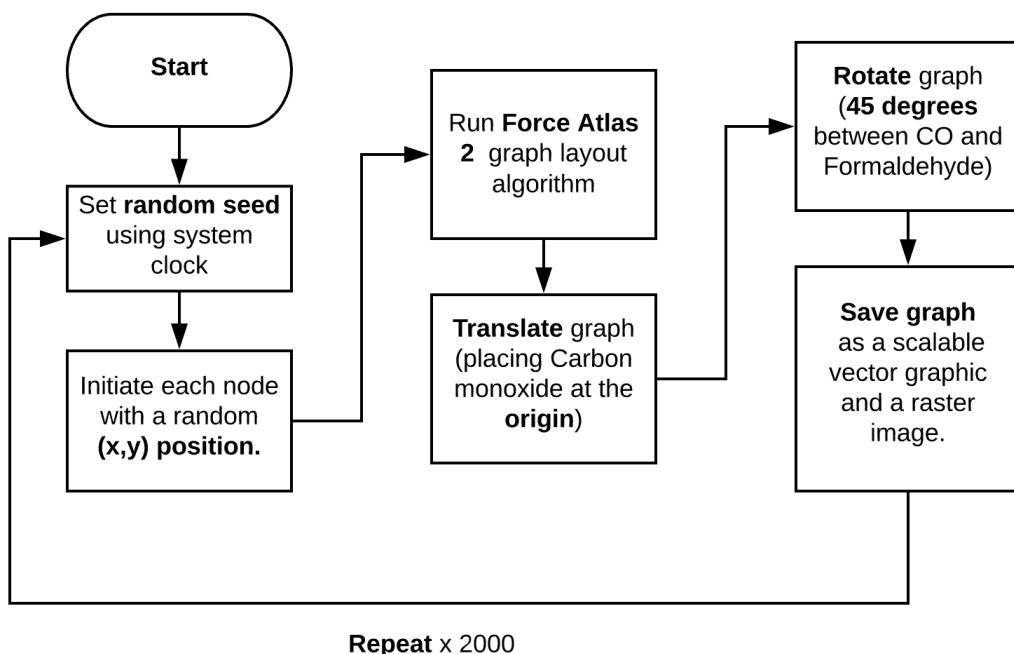


Figure 3.29: A flow chart of the process performed by the custom gephi script used to generate the data set

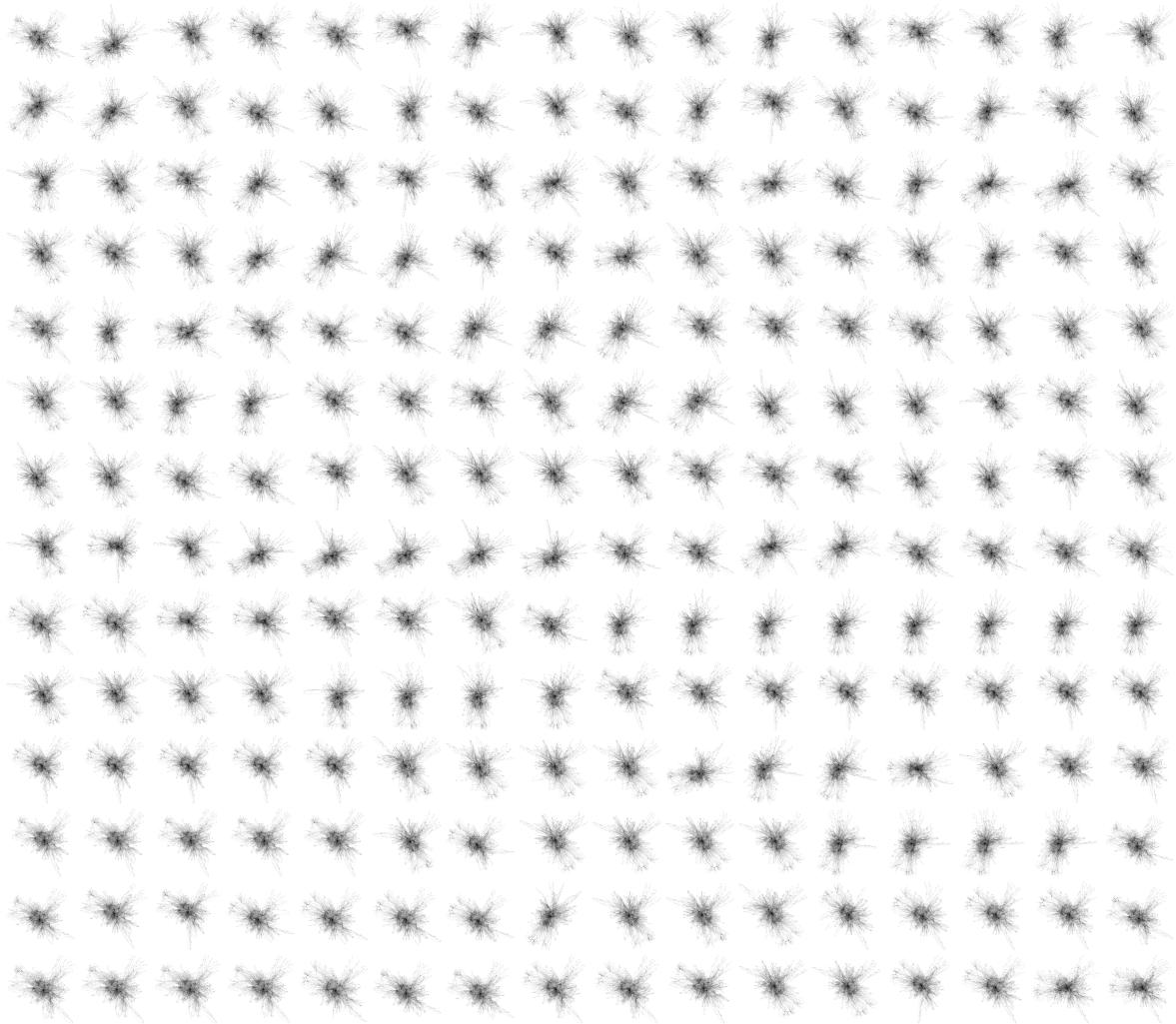


Figure 3.30: A sample of the 900 graphs generated using the force atlas 2 algorithm for the simulation output representative of the summer beijing chemistry at noon.

3.3.4.1 Trends in the Chemistry

Due to the construction protocols of the master chemical mechanism, ??, primary emitted compounds are oxidised to produce a cascade of species, ultimately ending at carbon dioxide¹³ and water. As this process is central to the construction of the mechanism, it follows that they may be used to explain any features uncovered using the network layout.

3.3.4.1.1 Network shape

Using Figure 3.30 the pattern recognition capabilities of the human mind identify a certain shape associated with many of the networks. Upon closer inspection it may be hypothesized that the chemistry is split into three main branches. Figure 3.31 categorises all the primary emitted species,

¹³The MCM conserves the number of carbons, allowing CO₂ to be introduced.

and then uses voronoi tessellation¹⁴ to colour neighbouring nodes and their links by the classification of the closest primary emitted species. Using this it is possible to separate the MCM network into an aromatic branch, a terpene branch, an alkane and straight chain alkene branches. Such branches not only help us identify changes of chemistry due to biogenic or anthropogenic sources, but also emphasise the path taken to carbon dioxide and water. Since the MCM does not contain CO₂ we see all the different groups converge on Carbon Monoxide (white, centre). Using this format, we may now compare the orientation of the many automatically generated layouts.

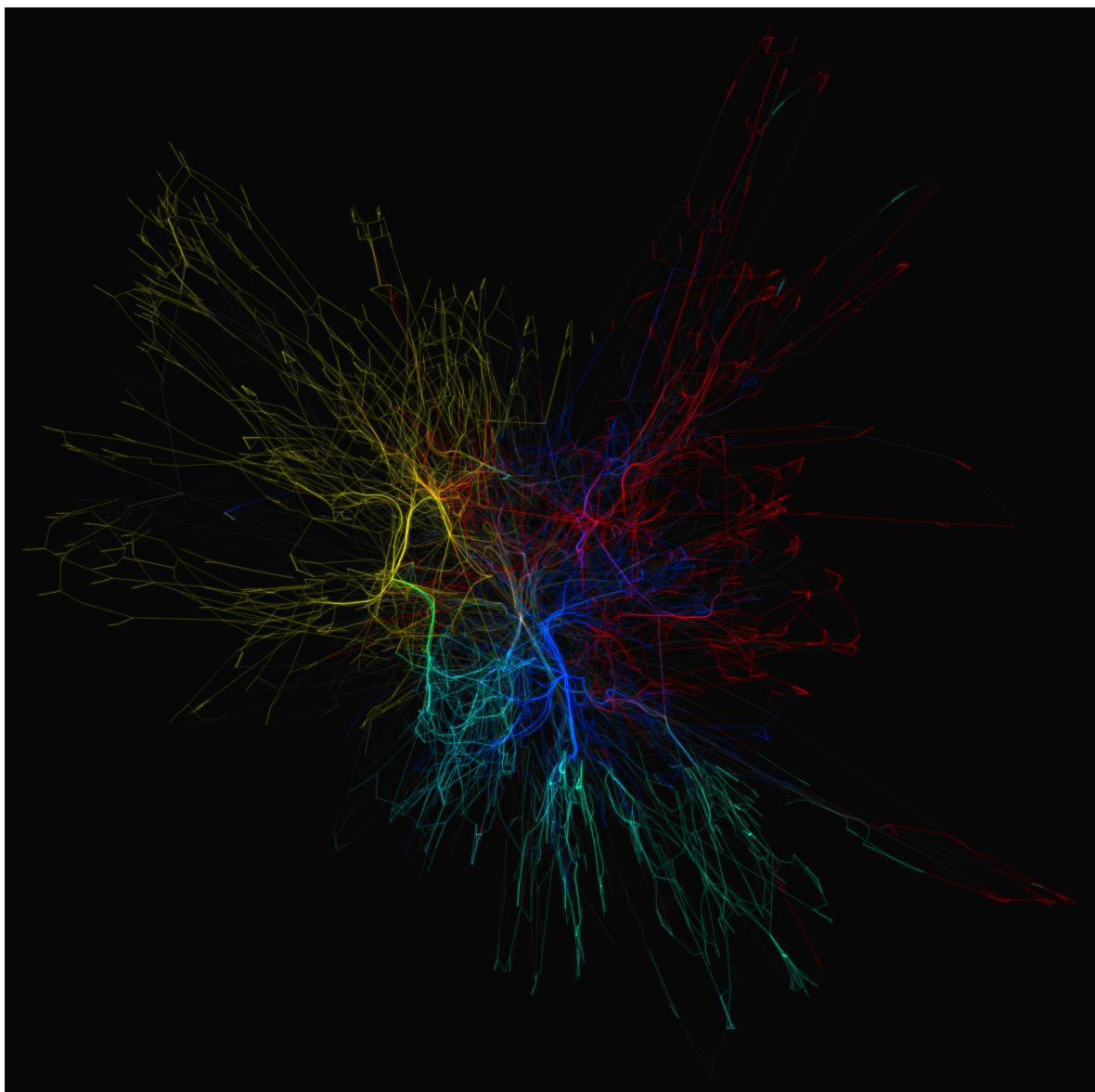


Figure 3.31: Hi-lighting the groups of species, and their products within the MCM network graph. These are **Aromatics** (gold) , **terpenes** (turquoise) and **Alkane/Alkene** carbon chains (red/blue)

¹⁴see chapter xxx for an example of this

3.3.4.1.2 Pattern Matching using t-SNE

t-Distributed Stochastic Neighbor Embedding (t-SNE) is a dimensionality reduction technique used in automatic categorisation of images or photographs [Stefaner, , Sangkloy et al., 2016]. This is the same process as referenced in EARLIERREF and described in detail within Chapter...

To compare the generated networks, we flatten the pixel matrix for each centered image in the dataset, and assign the output list to each filename. The resultant dataframe is then fed into the t-SNE algorithm in the Scikit Learn package, [Pedregosa et al., 2011]. This reduces the logical list of pixels for each image into a two dimensional representation of their similarity. We plot each file, for its (x, y) coordinate, and isolate clusters of similarity using density contours in

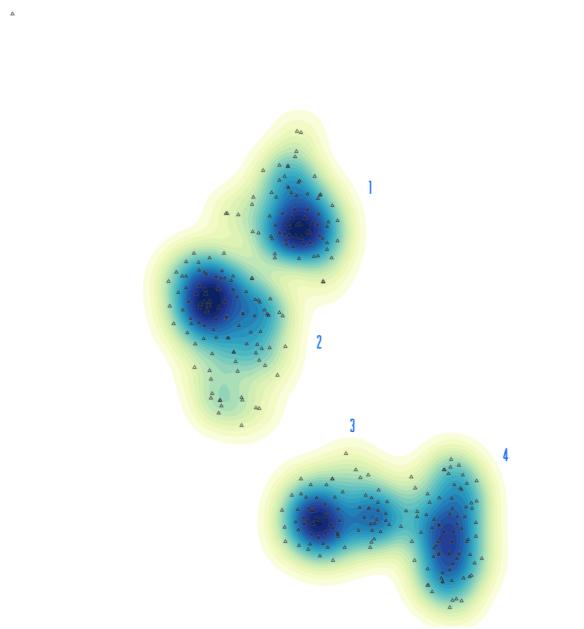


Figure 3.32: **A normalised scatter plot of 2D space produced by the t-SNE algorithm.** Each triangle represents a different file, and the colours/density contours show the regions in which we find similar images/graphs.

Using interactivity and/or vector cluster detection techniques it is possible examine which files contribute to an area of high density. Figure 3.33 shows a sample of four graphs from each corresponding cluster. Although individual node locations may vary, patterns on the macro scale start to emerge, with similar groups exhibiting symmetrical symmetry, e.g. groups 1/2 and 3/4. This suggests a constraint in the overall degree of freedom can be attributed solely to the network structure, and consequently the chemistry which forms this. The non-random nature of the produced graph layouts mean that it would be possible to juxtapose a variety of mechanisms using the force atlas 2 layout.

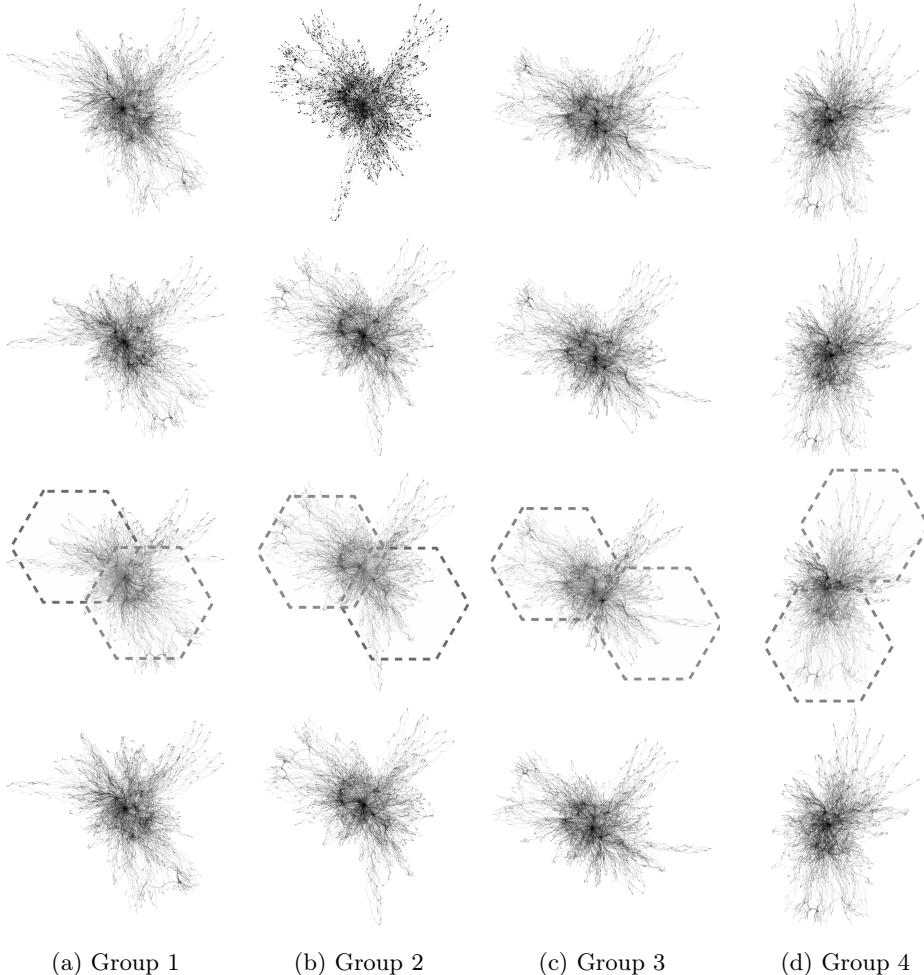


Figure 3.33: **A selection of graphs corresponding to the labeled clusters in Figure 3.32.** These reveal that symmetric similarity between like-positioned points within the t-SNE output.

3.4 Summary

Representing data in a visual format can be used to (utilise) the pattern recognition side of the human brain and alleviate the cognitive strain produced by numerical data. This is a technique used by the Samaritans (YEAR) with the use of cuneiform, and proved useful throughout.

In designing a visualisation it is important to use storytelling and select metaphors familiar to the reader. This should be paired with the correct encoding, as to reduce the time spent trying to comprehend a figure, and increase the knowledge transfer. [ref chapter 1]

When considering relationships, one such analogy lies in the ball and stick analogy. Much like holding hands, this symbolises a similarity between connected items and is the basis of a mathematical graph, or network. Such representations can be applied to the chemical complexity seen in species within the atmosphere.

In representing the chemistry within a mechanism as a graph we may visualise it with the use of a

force-directed layout. These are in essence a simple physical simulation, whereupon each graph node is repelled (like-charge), and connected nodes joined by a spring-like attractive force. It is found that the force atlas 2 algorithm not only produces the best visual aesthetic, but also conceptual understanding. Using this it is possible to see patterns such as the the partitioning of each network into aromatic, terpene and straight chain chemistry.

Although graph layouts have a range of local minima, the overall network structure of the MCM is constrained by its construction protocol (due to the allowed chemical reactions), and thus can be used to produce comparable graphs. This method of visualisation, in combination with interactive querying techniques, can aid in the comparison and understanding of large/complex chemistry simulations. This can be particularly useful in the explanation of specific interactions within a mechanism, or the exploration of temporal changes within a simulation.

In the next chapter, I shall extend the graph metaphor for atmospheric chemistry systems beyond that of just visualisation.

Bibliography

- [Archambault et al., 2014] Archambault, D., Abello, J., Kennedy, J., Kobourov, S., Ma, K.-L., Miksch, S., Muelder, C., and Telea, A. C. (2014). *Temporal Multivariate Networks*, pages 151–174. Springer International Publishing, Cham.
- [Aumont et al., 2005] Aumont, B., Szopa, S., and Madronich, S. (2005). Modelling the evolution of organic carbon during its gas-phase tropospheric oxidation: development of an explicit model based on a self generating approach. *Atmospheric Chemistry and Physics*, 5:2497–2517.
- [Bach, 2020] Bach, B. (2020). Confluent Graphs. <https://aviz.fr/~bbach/confluentgraphs/>. Accessed: 2020-1-9.
- [Bach et al., 2017] Bach, B., Riche, N. H., Hurter, C., Marriott, K., and Dwyer, T. (2017). Towards Unambiguous Edge Bundling: Investigating Confluent Drawings for Network Visualization. *IEEE transactions on visualization and computer graphics*, 23(1):541–550.
- [Bastian et al., 2009] Bastian, M., Heymann, S., and Jacomy, M. (2009). Gephi: An open source software for exploring and manipulating networks.
- [Bennett et al., 2007] Bennett, C., Ryall, J., Spalteholz, L., and Gooch, A. (2007). The Aesthetics of Graph Visualization. In *Computational Aesthetics in Graphics, Visualization, and Imaging*. The Eurographics Association.
- [Bergwerf, 2019] Bergwerf, H. (2019). MolView. <http://molview.org/>. Accessed: 2019-10-10.
- [Chernobelskiy et al., 2012] Chernobelskiy, R., Cunningham, K. I., Goodrich, M. T., Kobourov, S. G., and Trott, L. (2012). *Force-Directed Lombardi-Style Graph Drawing*, pages 320–331. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Davidson and Harel, 1996] Davidson, R. and Harel, D. (1996). Drawing graphs nicely using simulated annealing. *ACM Trans. Graph.*, 15(4):301–331.
- [Di Battista et al., 1994] Di Battista, G., Eades, P., Tamassia, R., and Tollis, I. G. (1994). Algorithms for drawing graphs: An annotated bibliography. *Comput. Geom. Theory Appl.*, 4(5):235–282.

- [Di Battista et al., 2004] Di Battista, G., Mariani, F., Patrignani, M., and Pizzonia, M. (2004). *BG-Play: A System for Visualizing the Interdomain Routing Evolution*, pages 295–306. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Dianati, 2016] Dianati, N. (2016). Unwinding the hairball graph: Pruning algorithms for weighted complex networks. *Phys. Rev. E*, 93:012304.
- [Dixon, 2012] Dixon, D. (2012). *Analysis Tool or Research Methodology: Is There an Epistemology for Patterns?*, pages 191–209. Palgrave Macmillan UK, London.
- [Duyckaerts and Godefroy, 2000] Duyckaerts, C. and Godefroy, G. (2000). Voronoi tessellation to study the numerical density and the spatial distribution of neurones. *Journal of Chemical Neuroanatomy*, 20(1):83 – 92.
- [Dwyer et al., 2006a] Dwyer, T., Koren, Y., and Marriott, K. (2006a). Drawing directed graphs using quadratic programming. *IEEE Transactions on Visualization and Computer Graphics*, 12(4):536–548.
- [Dwyer et al., 2006b] Dwyer, T., Koren, Y., and Marriott, K. (2006b). IPSEP-COLA: an incremental procedure for separation constraint layout of graphs. *IEEE Trans. Vis. Comput. Graph.*, 12(5):821–828.
- [Dwyer et al., 2006c] Dwyer, T., Marriott, K., and Stuckey, P. J. (2006c). *Fast Node Overlap Removal*, pages 153–164. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Eades, 1984] Eades, P. (1984). A heuristic for graph drawing. In *proceedings*.
- [Ellis, 2016] Ellis, D. (2016). wolfiex/frmd: 2016.
- [Foo, 2019] Foo, B. (2019). Memory Underground - Convert Your Memories Into A Subway Map - Home. <http://memoryunderground.com/>. Accessed: 2019-12-12.
- [Friedrich and Schreiber, 2004] Friedrich, C. and Schreiber, F. (2004). Flexible layering in hierarchical drawings with nodes of arbitrary size. In *Proceedings of the 27th Australasian Conference on Computer Science - Volume 26*, ACSC '04, pages 369–376, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.
- [Fruchterman and Reingold, 1991] Fruchterman, T. M. J. and Reingold, E. M. (1991). Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164.
- [Fürster et al., 2019] Fürster, H., Ganian, R., Klute, F., and Nöllenburg, M. (2019). On strict (outer-)confluent graphs.

- [García-Pérez et al., 2019] García-Pérez, G., Allard, A., Ángeles Serrano, M., and Boguñá, M. (2019). Mercator: uncovering faithful hyperbolic embeddings of complex networks. *arxiv*.
- [Gershon and Page, 2001] Gershon, N. and Page, W. (2001). What storytelling can do for information visualization. *Commun. ACM*, 44(8):31–37.
- [Goodrich and Wagner, 1998] Goodrich, M. T. and Wagner, C. G. (1998). *A Framework for Drawing Planar Graphs with Curves and Polylines*, pages 153–166. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Görg et al., 2007] Görg, C., Pohl, M., Qeli, E., Xu, K., Ebert, A., and Meyer, J. (2007). *Visual Representations*, pages 163–230. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Hazewinkel, 1997] Hazewinkel, M. (1997). *Encyclopaedia of Mathematics: Supplement*. Number v. 1 in Encyclopaedia of Mathematics. Springer Netherlands.
- [Holten, 2006] Holten, D. (2006). Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):741–748.
- [Hu, 2004] Hu, Y. (2004). Efficient, High-Quality Force-Directed Graph Drawing. *web*.
- [Jacomy et al., 2014] Jacomy, M., Venturini, T., Heymann, S., and Bastian, M. (2014). ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software. *PloS one*, 9(6):e98679.
- [Jankun-Kelly et al., 2014] Jankun-Kelly, T. J., Dwyer, T., Holten, D., Hurter, C., Nöllenburg, M., Weaver, C., and Xu, K. (2014). *Scalability Considerations for Multivariate Graph Visualization*, pages 207–235. Springer International Publishing, Cham.
- [Jenkin et al., 1997] Jenkin, M. E., Saunders, S. M., and Pilling, M. J. (1997). The tropospheric degradation of volatile organic compounds: a protocol for mechanism development. *Atmospheric environment*, 31(1):81–104.
- [Johnson, 2010] Johnson, S. (2010). *Where Good Ideas Come From*. Penguin Publishing Group.
- [Kerren et al., 2014] Kerren, A., Purchase, H. C., and Ward, M. O. (2014). *Introduction to Multivariate Network Visualization*, pages 1–9. Springer International Publishing, Cham.
- [Kirkpatrick et al., 1983] Kirkpatrick, S., Gelatt, C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598):671–680.
- [Klicpera et al., 2018] Klicpera, J., Bojchevski, A., and Günnemann, S. (2018). Predict then Propagate: Graph Neural Networks meet Personalized PageRank. *Arxiv*.

- [Kohlbacher et al., 2014] Kohlbacher, O., Schreiber, F., and Ward, M. O. (2014). *Multivariate Networks in the Life Sciences*, pages 61–73. Springer International Publishing, Cham.
- [Kumar and Garland, 2006] Kumar, G. and Garland, M. (2006). Visual exploration of complex time-varying graphs. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):805–812.
- [Lu, 2019] Lu, S. (2019). d3-annotate. <https://d3-annotation.susielu.com/>. Accessed: 2019-12-16.
- [Lyons, 1992] Lyons, K. A. (1992). Cluster busting in anchored graph drawing. In *Proceedings of the 1992 Conference of the Centre for Advanced Studies on Collaborative Research - Volume 1, CASCON '92*, pages 7–17. IBM Press.
- [Ma and Muelder, 2013] Ma, K. and Muelder, C. W. (2013). Large-scale graph visualization and analytics. *Computer*, 46(7):39–46.
- [Martin Grandjean, 2016] Martin Grandjean (2016). Connected World: Untangling the Air Traffic Network. <http://www.martingrandjean.ch/connected-world-air-traffic-network/>. Accessed: 2019-12-16.
- [Metropolis et al., 1953] Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21(6):1087–1092.
- [Montañez, 2016] Montañez, A. (2016). How Science Visualization Can Help Save the World. <https://blogs.scientificamerican.com/scientificamericanscience/2016/01/04/how-science-visualization-can-help-save-the-world/>. Accessed: 2020-1-4.
- [Mortenson, 1999] Mortenson, M. (1999). *Mathematics for Computer Graphics Applications*. Industrial Press.
- [Muelder et al., 2014] Muelder, C., Gou, L., Ma, K.-L., and Zhou, M. X. (2014). *Multivariate Social Network Visual Analytics*, pages 37–59. Springer International Publishing, Cham.
- [Noack, 2004] Noack, A. (2004). *An Energy Model for Visual Graph Clustering*, pages 425–436. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Norman, 2005] Norman, D. (2005). *Emotional Design: Why We Love (or Hate) Everyday Things*. Basic Books.
- [Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau,

- D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- [Pohl et al., 2009] Pohl, M., Schmitt, M., and Diehl, S. (2009). Comparing the Readability of Graph Layouts using Eyetracking and Task-oriented Analysis. In *Computational Aesthetics in Graphics, Visualization, and Imaging*. The Eurographics Association.
- [Probst and Reymond, 2018] Probst, D. and Reymond, J.-L. (2018). SmilesDrawer: Parsing and Drawing SMILES-Encoded Molecular Structures Using Client-Side JavaScript. *Journal of chemical information and modeling*, 58(1):1–7.
- [Purchase, 1997] Purchase, H. (1997). *Which aesthetic has the greatest effect on human understanding?*, pages 248–261. Springer Berlin Heidelberg, Berlin, Heidelberg.
- [Purchase, 2002] Purchase, H. C. (2002). Metrics for graph drawing aesthetics. *Journal of Visual Languages and Computing*, 13(5):501 – 516.
- [Purchase et al., 2003] Purchase, H. C., Colpoys, L., Carrington, D., and McGill, M. (2003). *UML Class Diagrams: An Empirical Study of Comprehension*, pages 149–178. Springer US, Boston, MA.
- [Roberts et al., 2014] Roberts, J. C., Yang, J., Kohlbacher, O., Ward, M. O., and Zhou, M. X. (2014). *Novel Visual Metaphors for Multivariate Networks*, pages 127–150. Springer International Publishing, Cham.
- [Sangkloy et al., 2016] Sangkloy, P., Burnell, N., Ham, C., and Hays, J. (2016). The sketchy database: Learning to retrieve badly drawn bunnies. *ACM Trans. Graph.*, 35(4).
- [Shneiderman, 1996] Shneiderman, B. (1996). The eyes have it: A task by data type taxonomy for information visualizations. In *Proceedings of the 1996 IEEE Symposium on Visual Languages*, VL '96, pages 336–, Washington, DC, USA. IEEE Computer Society.
- [Shneiderman, 1997] Shneiderman, B. (1997). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 3rd edition.
- [Stefaner,] Stefaner, M. Truth & Beauty - Multiplicity. <https://truth-and-beauty.net/projects/multiplicity>. Accessed: 2020-1-2.
- [Steven Franconeri, 2018] Steven Franconeri (2018). OpenVis Conference Proceedings.
- [Taylor and Rodgers, 2005] Taylor, M. and Rodgers, P. (2005). Applying graphical design techniques to graph visualisation. In *Ninth International Conference on Information Visualisation, 06-08 July 2005, London, England: Proceedings*, pages 651–656. IEEE Computer Society.

[Thomas, 1952] Thomas, P. (1952). *Conformal Projections in geodesy and cartography*. Special publication. Coast and Geodetic Survey.

[VTL, 2019] VTL (2019). Visual Thinking Lab. <http://visualthinking.psych.northwestern.edu/>. Accessed: 2019-12-15.

[Ware, 2013a] Ware, C. (2013a). Chapter four - color. In Ware, C., editor, *Information Visualization (Third Edition)*, Interactive Technologies, pages 95 – 138. Morgan Kaufmann, Boston, third edition edition.

[Ware, 2013b] Ware, C. (2013b). Chapter three - lightness, brightness, contrast, and constancy. In Ware, C., editor, *Information Visualization (Third Edition)*, Interactive Technologies, pages 69 – 94. Morgan Kaufmann, Boston, third edition edition.

[Ware, 2013c] Ware, C. (2013c). Chapter two - the environment, optics, resolution, and the display. In Ware, C., editor, *Information Visualization (Third Edition)*, Interactive Technologies, pages 31 – 68. Morgan Kaufmann, Boston, third edition edition.

[Ware et al., 2002] Ware, C., Purchase, H., Colpoys, L., and McGill, M. (2002). Cognitive measurements of graph aesthetics. *Information Visualization*, 1(2):103–110.

[Wybrow et al., 2014] Wybrow, M., Elmqvist, N., Fekete, J.-D., von Landesberger, T., van Wijk, J. J., and Zimmer, B. (2014). *Interaction in the Visualization of Multivariate Networks*, pages 97–125. Springer International Publishing, Cham.