JOINING AND SPLITTING GRAPHS

KUNAL MANDALIA

ABSTRACT. This paper is concerned with pulling together information from separate graphs to form a single graph, the process of joining graphs. The reverse process is then considered, splitting a single graph into smaller graphs. Finally, the interaction between joining and splitting graphs is considered and how these two processes can be used to achieve efficiency.

1. PRELIMINARIES

A graph is set of points called nodes connected to each other by paths. Figure 1 shows a graph, $G(A)$, with three nodes where node a is linked to node b and a is linked to c and c is linked to a.

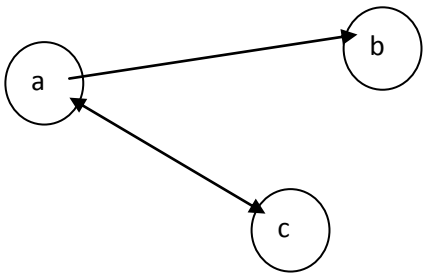

Figure 1: Graph *G(A)*

This graph can be represented as a matrix. When a matrix is used to represent a graph in this way it is called an adjacency matrix.

$$Adjacency\ matrix\ of\ graph\ G(A) = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

Here the convention is that the first row and column correspond to node a, the second row and column correspond to node b and the third row and column correspond to node c. Element $a_{(i,j)}$ corresponds to the path from node i to node j. A one has been used to indicate a path and 0 to indicate no path (or zero path). Generally, any value can be assigned to an element in an adjacency matrix and its interpretation will depend upon its context. For example the graph $G(B)$ in figure 2 could have the nodes represent cities and the numbers between the links could represent the distance from one city to another.

Date: May 17, 2011.
E-mail: Kunalmandalia@hotmail.com.

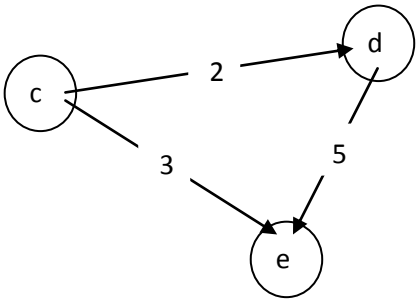

Figure 2: Graph *G(B)*

$$Adjacency\ matrix\ of\ graph\ B = \begin{bmatrix} 0 & 2 & 3 \\ 0 & 0 & 5 \\ 0 & 0 & 0 \end{bmatrix}$$

2. MOTIVATING EXAMPLE

Consider the graphs $G(A)$ and $G(B)$**.** If both graphs are set in the same contextual frame, for example both graphs correspond to cities and their associated distances then piecing together the two graphs into a single graph would be useful for representing all information compactly. This would allow you to identify all links with clarity. The end product of joining graphs $G(A)$ and $G(B)$ is $G(C)$, I will call this the resultant graph, shown below.

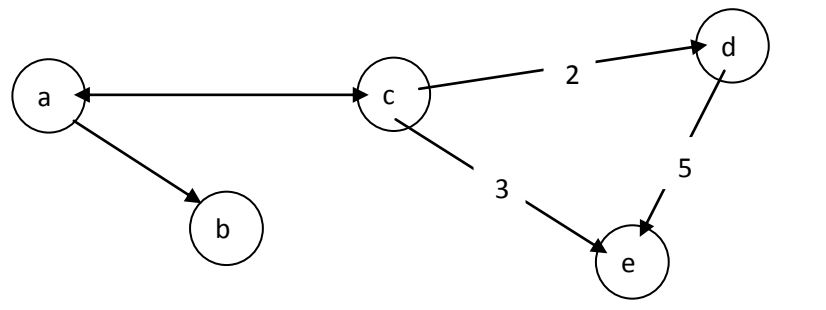

Figure 3: Graph *G(C)*

$$Adjacency\ matrix\ of\ graph\ C = \begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

How can resultant graphs be generated computationally? What sort of algorithm is required? The remainder of this paper looks at the mathematical process of joining graphs. We then consider how graphs may be split and finally look at why joining and splitting graphs is useful.

3. CONSTRUCTABILITY

The above examples of adjacency matrices may lead one to think that a graph and an its adjacency matrix convey the same information but is not true. Consider graph $G(D)$ and adjacency matrices $D_1, D_2\ and\ D_3$.

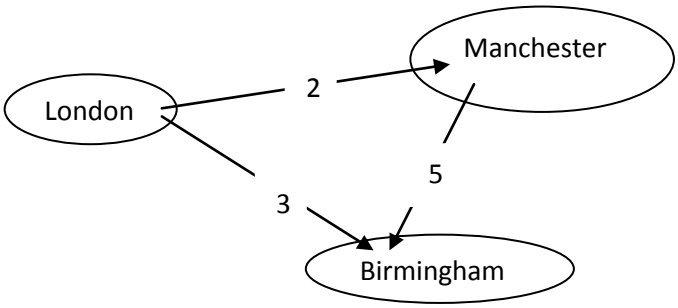

Figure 4: Graph *G(D)*

$$D_1 = \begin{bmatrix} 0 & 2 & 3 \\ 0 & 0 & 5 \\ 0 & 0 & 0 \end{bmatrix}, D_2 = \begin{bmatrix} 0 & 0 & 5 \\ 2 & 0 & 3 \\ 0 & 0 & 0 \end{bmatrix}, D_3 = \begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 2 \\ 5 & 0 & 0 \end{bmatrix}$$

Matrix D_1 is derived from graph $G(D)$ by letting the first row and first column refer to London, the second row and second column refer to Manchester and the third row and third column refer to Birmingham.

Matrix D_2 is derived from graph $G(D)$ by letting the first row and first column refer to Manchester, the second row and second column refer to London and the third row and third column refer to Birmingham.

Matrix D_3 is derived from graph $G(D)$ by letting the first row and first column refer to Birmingham, the second row and second column refer to London and the third row and third column refer to Manchester.

Thus one graph can generate multiple adjacency matrices, specifically if a graph $G(A)$ contains n nodes then there are $n!$ matrices that can be generated from $G(A)$ because the elements of $N(A)$ can be ordered $n!$ ways. $G(D)$ contains three nodes so 3!=6 adjacency matrices can be generated.

Now consider a specific adjacency matrix, say D_1. Can $G(D)$ be constructed from D_1? The structure of $G(D)$ is represented by D_1 but a graph generated by D_1 will have arbitrarily

named nodes. We cannot be sure that the two graphs will be equivalent since the name of the nodes in each graph are not guaranteed to be the same.

Since we can construct the set of all adjacency matrices through a graph but cannot generate the same graph through any of its adjacency matrices it follows that the graph contains more information than its adjacency matrix. The adjacency matrix needs an ordered set of nodes to provide enough information to construct a unique graph.

4. THE ADJACENCY SCRIPT

An adjacency matrix needs an ordered set of nodes to provide enough information to construct a graph. Denote the combination of an adjacency matrix and an ordered set of nodes the script of an adjacency matrix. Thus the adjacency script of matrix D_1 may be given by the following:

$$\mathcal{D}_1 = (D_1, N(D_1) = \{London, Manchester, Birmingham\})$$

The only problem here is that $\{London, Manchester, Birmingham\}$ is not ordered, since it is just a set. A well defined graph, defined below addresses this.

5. DEFINITION OF A WELL DEFINED GRAPH

A well defined graph, $G(\mathcal{A})$, consists of an adjacency script, $\mathcal{A}$,which is an object containing a matrix, A, along with an ordered set of nodes, $N(A)$, such that:

1. $N(A) = \{b_1, \dots, b_n\} : N(A)$ is a finite set of node names
2. $A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{bmatrix} :$ $a_{i,j}$ is the value of the path from the i^{th} to the j^{th} node in $N(A)$
3. There exists a bijective function $f_{\mathcal{A}}: N(A) \to \{1, \dots, n\} : n = |N(A)|$
$$f_{\mathcal{A}}(b_k) = k : k \in \{1, \dots, n\}$$

$N(A)$ needs to be ordered but there are no restrictions on how the ordering takes place. For example, if these nodes were the names of cities then alphabetical ordering may be appropriate.

The function $f_{\mathcal{A}}(b_k) = k$ orders elements in $N(A)$. For example, the n^{th} element in $N(A)$ is given by $f_{\mathcal{A}}^{-1}(n)$.

6. ADJACENCY SCRIPT $\mathcal{D}_1$

$$\mathcal{D}_1 = (D_1, N(D_1) = \{London, Manchester, Birmingham\}):$$

$$\exists\, f_{\mathcal{D}_1}: N(D_1) \to \{1, \dots, n\} : n = |N(D_1)|$$

$$f_{\mathcal{D}_1}(London) = 1, f_{\mathcal{D}_1}(Manchester) = 2, f_{\mathcal{D}_1}(Birmingham) = 3$$

$\mathcal{D}_2$ and $\mathcal{D}_3$ are generated in a similar way.

In the remainder of the paper such function will be made implicit by the order in which the elements appear its node set. E.g. $A = \{London, Manchester, Birmingham, a, b, c\}$ implies the first element is London; $f_{\mathcal{A}}(London) = 1$, the second element is Manchester; $f_{\mathcal{A}}(Manchester) = 2$ and so on.

7. RELATIONSHIP TYPES BETWEEN GRAPH $G(\mathcal{A})$ AND $G(\mathcal{B})$

The relationship between any graph $G(\mathcal{A})$ and $G(\mathcal{B})$ falls into one of three categories:

- Type 1. $N(A) \bigcap N(B) = \emptyset$. The graphs have no nodes in common.
- Type 2. $N(A) \bigcap N(B) = C \neq \emptyset : A(c, c) = B(c, c) = 0 \; \forall \; c \in C$. Where $A(i, j)$ is the value of the path from node i to j in matrix A.The graphs have node(s) in common and there is no path among these node(s).
- Type 3.
  $N(A) \bigcap N(B) = C \neq \emptyset : A(c, c) = B(c, c) \neq 0 \; for \; atleast \; one \; pair \; (c, c) \in C \times C$.
  The graphs have common node(s) and there exists at least one path among these node(s)

8. EXAMPLES: JOINING TWO GRAPHS

1. Joining a pair of type 1 graphs $G(A)$ and $G(D)$ generates the resultant graph $G(\mathcal{X})$:

$$\mathcal{X} = \left(\begin{bmatrix} 0 & 2 & 3 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}, \{London, Manchester, Birmingham, a, b, c\} \right)$$

2. Joining a pair of type 2 graphs $G(A)$ and $G(B)$ generates the resultant graph $G(\mathcal{X})$:

$$\mathcal{X} = \left(\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \{a, b, c, d, e\} \right)$$

1. Joining a pair of type 3 graphs $G(A)$ and $G(E)$ generates the resultant graph $G(\mathcal{X})$:

$$\mathcal{X} = \left(\begin{bmatrix} 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 2 & 3 \\ 0 & 0 & 0 & 0 & 5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \{a, b, c, d, e\} \right)$$

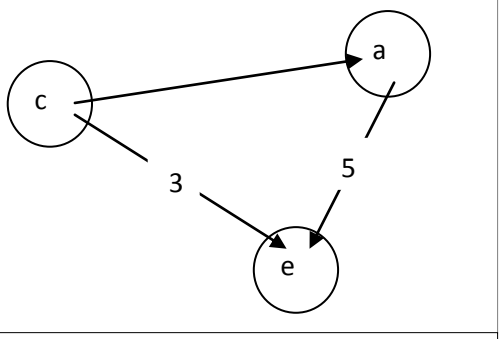

Figure 5: Graph *G(E)*

9. PROPERTIES OF THE RESULTANT GRAPH

Let $\mathcal{A} = (A, N(A))$ be an adjacency script of graph $G(B)$ joined with $G(C)$. The number of elements in $N(A)$ is given by the union of elements in $N(B)\ and\ N(C)$ thus:

$$N(A) = N(B) \bigcup N(C)$$

$$|N(A)| = |N(B) \bigcup N(C)|$$

Although $N(B)$ and $N(C)$ are ordered by functions $f_{\mathcal{B}}$ and $f_{\mathcal{C}}$, $N(A)$ is ordered by $f_{\mathcal{A}}$ which is independent of functions $f_{\mathcal{B}}$ and $f_{\mathcal{C}}$ thus the union of the above sets does not pose a problem with respect to ordering.

The adjacency matrix A contains $|N(A)|^2$ elements. The resultant's adjacency script $\mathcal{A}$ will have the following structure:

$$\mathcal{A} = \left(A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{bmatrix}, N(A) \right) : \ |N(A)| = n$$

Representing the adjacency script of the resultant graph in terms of features of its sub graphs:

$$\mathcal{A} = \left(A = T(B, C), N(A) = N(B) \bigcup N(C) \right)$$

Where $T(B, C)$ is a transformation (or algorithm, etc) of adjacency matrices $B\ and\ C$. Finding $T(B, C)$ is the key to generating the resultant's adjacency script computationally since $N(A)$ is known.

10. EXPANSION

Suppose we wish to join graphs $G(\mathcal{A})$ and $G(\mathcal{D})$ to form the resultant graph $G(\mathcal{E})$. Expressing $\mathcal{A}$ and $\mathcal{D}$ with the same dimension as $\mathcal{E}$ opens up the possibility of using matrix operators such as multiplication and addition, which may be useful for finding $E = T(A, D)$.

Consider the graphs in figure 1 and figure 3. The adjacency script of these two graphs is as follows:

$$\mathcal{A} = \left(A = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \qquad N(A) = \{a, b, c\} \right)$$

$$\mathcal{D} = \left(D = \begin{bmatrix} 0 & 0 & 0 \\ 3 & 0 & 2 \\ 5 & 0 & 0 \end{bmatrix}, \qquad N(D) = \{Birmingham, London, Manchester\} \right)$$

The resultant adjacency script $\mathcal{E}$ is given by:

$$\mathcal{E} = \left(E = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 2 \\ 0 & 0 & 5 & 0 & 0 & 0 \end{bmatrix}, \qquad N(E) = \{a, b, Birmingham, c, London, Manchester\} \right)$$

Representing $\mathcal{A}$ in the same dimension as $\mathcal{E}$ gives $\mathcal{A}^*$:

$$\mathcal{A}^* = \left(A^* = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}, \qquad \{a, b, Birmingham, c, London, Manchester\} \right)$$

The graph of this script is shown below.

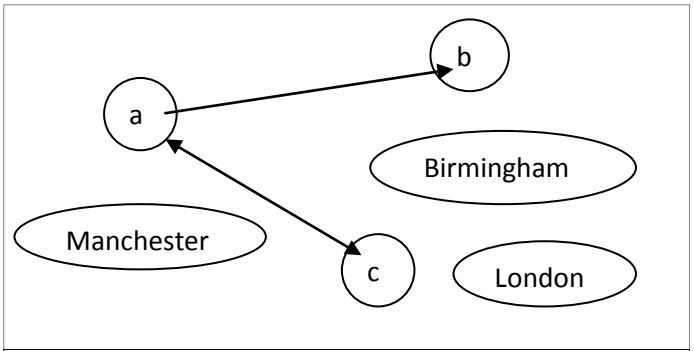

Figure 6: Graph G($\mathcal{A}^*$)

Therefore expressing $\mathcal{A}$ with the same dimension as $\mathcal{E}$ implies that only nodes are added to a graph (if the two graphs are distinct) without any paths. We shall call expressing a script in a higher dimension an *expansion.*

The expansion $\mathcal{D}^*$ is shown below.

$$\mathcal{D}^* = \left(D^* = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 2 \\ 0 & 0 & 5 & 0 & 0 & 0 \end{bmatrix}, \qquad \{a, b, Birmingham, c, London, Manchester\} \right)$$

From the above example notice:

$$E = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 2 \\ 0 & 0 & 5 & 0 & 0 & 0 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 & 2 \\ 0 & 0 & 5 & 0 & 0 & 0 \end{bmatrix}$$

Representing $E = T(A, D)$ gives $E = A^* + D^*$ so $\mathcal{E}$ is given by:

$$\mathcal{E} = \left(A^* + D^*, \;\; N(C) = N(A) \bigcup N(D) \right)$$

Is this a general result or just specific to the example chosen? It turns out this result is true for pairs of type 1 and type 2 graphs, proven below.

Generally speaking, if $G(C)$ is the resultant of graphs $G(\mathcal{A})$ and $G(\mathcal{B})$ then the expansion $\mathcal{A}^*$ is given by:

$$\mathcal{A}^* = \left(A^* : a^*_{i,j} = \begin{cases} 0 & : \; either \; f_{\mathcal{A}^*}^{-1}(i), f_{\mathcal{A}^*}^{-1}(j) \notin N(A) \\ a_{f_{\mathcal{A}}^{-1}\left(f_{\mathcal{A}^*}^{-1}(i)\right), f_{\mathcal{A}}^{-1}\left(f_{\mathcal{A}^*}^{-1}(j)\right)} & \in A \;\; otherwise, \end{cases} \;\; N(C) = N(A) \bigcup N(B) \right)$$

$$\mathcal{B}^* = \left(B^* : b^*_{i,j} = \begin{cases} 0 & : \; either \; f_{\mathcal{B}^*}^{-1}(i), f_{\mathcal{B}^*}^{-1}(j) \notin N(A) \\ b_{f_{\mathcal{B}}^{-1}\left(f_{\mathcal{B}^*}^{-1}(i)\right), f_{\mathcal{B}}^{-1}\left(f_{\mathcal{B}^*}^{-1}(j)\right)} & \in A \;\; otherwise, \end{cases} \;\; N(C) = N(A) \bigcup N(B) \right)$$

Consider $a^*_{i,j}$ corresponding to node $f_{\mathcal{A}^*}^{-1}(i)$ to $f_{\mathcal{A}^*}^{-1}(j)$, if either of these nodes are not in $N(A)$ then they are the result of adding nodes without any paths thus $a^*_{i,j} = 0$. On the other hand if these nodes are in $N(A)$ then they correspond to the element indexed by $a_{f_{\mathcal{A}}^{-1}\left(f_{\mathcal{A}^*}^{-1}(i)\right), f_{\mathcal{A}}^{-1}\left(f_{\mathcal{A}^*}^{-1}(j)\right)}$ since $f_{\mathcal{A}}^{-1}\left(f_{\mathcal{A}^*}^{-1}(k)\right)$ converts a node into its index value in A.

Also note $f_{\mathcal{C}}$ is equivalent to $f_{\mathcal{A}^*}$ and $f_{\mathcal{B}^*}$.

11. JOINING A PAIR OF GRAPHS

Theorem 1. Given a pair of type 1 well defined graphs $G(\mathcal{A})$ and $G(\mathcal{B})$ the resultant graph $G(\mathcal{C})$ is given by:

$$G(\mathcal{C}):\ \mathcal{C} = \left(C = A^* + B^*, N(C) = N(A)\bigcup N(B)\right)$$

Proof. Suppose the resultant graph is given by $G(\mathcal{D}):\ \mathcal{D} = (D, N(D) = N(A) \cup N(B))$. Since $G(\mathcal{D})$ is a well defined graph there exists a bijective function

$$f_{\mathcal{D}}: N(D) \to \{1, \dots, n\}\ where\ n = |N(D)|\ such\ that\ f_{\mathcal{D}}(b_k) = k : k \in \{1, \dots, n\}$$

Then ${f_{\mathcal{D}}}^{-1}(k) = b_k : k \in \{1, \dots, n\}$. For some value(s) of k, ${f_{\mathcal{D}}}^{-1}(k) = b_k \in N(A)$. Denote this set of integers Z_a. For other values of $\tilde{k}$, ${f_{\mathcal{D}}}^{-1}(\tilde{k}) = b_{\tilde{k}} \in N(B)$. Denote this set of integer values by Z_b. Thus there exist bijective functions ${f_{\mathcal{D}}}^{-1}: Z_a \to N(A)$ and ${f_{\mathcal{D}}}^{-1}: Z_b \to N(B)$. Let $p \in Z_a, q \in Z_b$ then ${a^*}_{p,p} \in \{x, 0\}, {b^*}_{q,q} \in \{y, 0\}$ where x refers to non-zero paths in graph A^* and y refers to non-zero paths in graph B^*. By definition of a pair of type 1 graphs ${a^*}_{p,q} = \left({a^*}_{q,p}\right) = {b^*}_{p,q} = \left({b^*}_{q,p}\right) = 0$. By construction of the expansion matrices ${a^*}_{q,q} = {b^*}_{p,p} = 0$.

Consider the following cases:

$$d_{i,j} \in D: d_{i=p,j=p} = x\ \left(\Longrightarrow {a^*}_{p,p} = x, {b^*}_{p,p} = 0\right) = {a^*}_{p,p} + {b^*}_{p,p}$$

$$d_{i=p,j=p} = 0\ \left(\Longrightarrow {a^*}_{p,p} = 0, {b^*}_{p,p} = 0\right) = {a^*}_{p,p} + {b^*}_{p,p}$$

$$d_{i=p,j=q} = 0\ \left({a^*}_{p,q} = 0\ by\ definition, {b^*}_{p,q} = 0\ by\ definition\right) = {a^*}_{p,q} + {b^*}_{p,q}$$

$$d_{i=q,j=p} = 0\ \left({a^*}_{q,p} = 0\ by\ definition\ , {b^*}_{q,p} = 0\ by\ definition\right) = {a^*}_{q,p} + {b^*}_{q,p}$$

$$d_{i=q,j=q} = y\ \left({a^*}_{q,q} = 0\ , \Longrightarrow {b^*}_{q,q} = y\ \right) = {a^*}_{q,p} + {b^*}_{q,p}$$

$$d_{i=q,j=q} = 0\ \left({a^*}_{q,q} = 0\ , \Longrightarrow {b^*}_{q,q} = 0\ \right) = {a^*}_{q,q} + {b^*}_{q,q}$$

These cases constitute the entire set of elements contained in adjacency matrix D so it is true for all $i, j \in \{1, \dots, n\}$ that $d_{i,j} = {a^*}_{i,j} + {b^*}_{i,j}$ but $C = A^* + \mathcal{B}^* \therefore c_{i,j} = {a^*}_{i,j} + {b^*}_{i,j} \ \therefore c_{i,j} = d_{i,j} \Longrightarrow C = D \therefore\ G(\mathcal{C}) = G(\mathcal{D})$ and since $G(\mathcal{D})$ is the resultant $G(\mathcal{C}):\ \mathcal{C} = (C = A^* + \mathcal{B}^*, N(C) = N(A) \cup N(B))$ is one such construction that represents the resultant graph of a pair of type 1 graphs.

■

Theorem 2. Given a pair of type 2 well defined graphs $G(\mathcal{A})$ and $G(\mathcal{B})$ the resultant graph $G(\mathcal{C})$ is given by:

$$G(\mathcal{C}):\ \mathcal{C}=\left(C=A^{*}+B^{*},N(C)=N(A)\bigcup N(B)\right)$$

Proof. Suppose the resultant graph is given by $G(\mathcal{D}):\ \mathcal{D}=(D,N(D)=N(A)\cup N(B))$. Since $G(\mathcal{D})$ is a well defined graph there exists a bijective function

$$f_{\mathcal{D}}\colon N(D)\to\{1,\dots,n\}\ where\ n=|N(D)|\ such\ that\ f_{\mathcal{D}}(b_k)=k:k\in\{1,\dots,n\}$$

Then ${f_{\mathcal{D}}}^{-1}(k)=b_k:k\in\{1,\dots,n\}$. For some value(s) of k, ${f_{\mathcal{D}}}^{-1}(k)=b_k\in N(A)$. Denote this set of integers Z_a. For other values of $\tilde{k}$, ${f_{\mathcal{D}}}^{-1}(\tilde{k})=b_{\tilde{k}}\in N(B)$. Denote this set of integer values by Z_b. Thus there exist bijective functions ${f_{\mathcal{D}}}^{-1}\colon Z_a\to N(A)$ and ${f_{\mathcal{D}}}^{-1}\colon Z_b\to N(B)$. Let $p\in Z_a,q\in Z_b,\overline{p}\in(Z_a-(Z_a\cap Z_b)),\overline{q}\in(Z_b-(Z_b\cap Z_a)),m\in(Z_b\cap Z_a)$ and $T=\{p,q,\overline{p},\overline{q},m\}$ then D is given by $d_{i,j}\ \forall\ i,j\in T$.

Consider ${a^{*}}_{\overline{p},t}\in\{x,0\},{b^{*}}_{\overline{p},t}=0\ by\ definition$ and $a_{\overline{q},t}=0\ by\ definition,b_{\overline{q},t}=0:t\in T$ which is consistent with $d_{i,j}={a^{*}}_{i,j}+{b^{*}}_{i,j}=c_{i,j}$ thus we only need to consider $T'=\{p,q,m\}$ which is equivalent to looking at paths from node r to s (r,s):
$(p,p),(p,q),(p,m),(q,q),(q,p),(q,m),(m,m)$

Note that considering (r,s) also considers (s,r) thus there is no loss of generality.

${a^{*}}_{m,m}+{b^{*}}_{m,m}=0$ otherwise there would be common paths, which is a feature of type 3 graphs not type 2.

${a^{*}}_{p,p}\in\{x,0\},{b^{*}}_{p,p}=0$ since if ${b^{*}}_{p,p}=y$ this would imply multiple paths and if ${b^{*}}_{p,p}=x$ this shows there are common paths, a contradiction to the type 2 construction.

Analogously ${a^{*}}_{q,q}=0,{b^{*}}_{q,q}\in\{y,0\}$. (p,m) and (q,m) are subsets of (p,p) and (q,q) respectively thus these have already been considered. Finally consider (p,q), but $p\in\{\overline{p},m\}$ so (p,q) represents cases $(\overline{p},q)$ and (m,q) but both of these cases has already been looked at.

Thus for all $i,j\in T,d_{i,j}={a^{*}}_{i,j}+{b^{*}}_{i,j}=\ c_{i,j}\ \therefore C=D\ \therefore\ G(\mathcal{C})=G(\mathcal{D})$

■

The construction $G(\mathcal{C}):\ \mathcal{C}=(C=A^{*}+\mathcal{B}^{*},N(C)=N(A)\cup N(B))$ does not represent the resultant graph if sub-graphs $G(\mathcal{A})$ and $G(\mathcal{B})$ are type 3, that is, they have common paths. Suppose for some pair of intergers (α,β) ${a^{*}}_{\alpha,\beta}=\gamma$, ${b^{*}}_{\alpha,\beta}=\gamma$ then $d_{\alpha,\beta}=\gamma$ but $c_{\alpha,\beta}=2\gamma$. Thus this construction is not a universal, it does not hold for all types of graphs.

Theorem 3. $G(\mathcal{C})$ is a universal resultant graph for any two sub- graphs $G(\mathcal{A})$ and $G(\mathcal{B})$:

$$\mathcal{C}=\left(C\colon c_{i,j}=\begin{cases}x\in\{{a^{*}}_{i,j},{b^{*}}_{i,j}\} & :\left|\{{a^{*}}_{i,j},{b^{*}}_{i,j}\}\right|=1\\ x\in\left(\{{a^{*}}_{i,j},{b^{*}}_{i,j}\}-\{0\}\right) & :\left|\{{a^{*}}_{i,j},{b^{*}}_{i,j}\}\right|=2\end{cases},N(C)=N(A)\bigcup N(B)\right)$$

Proof. Suppose the resultant graph is given by $G(\mathcal{D}) : \mathcal{D} = (D, N(D) = N(A) \cup N(B))$ then $d_{i,j} \in \{u, v, z, 0\}$ where u is a non-zero path derived from graph $G(\mathcal{A}), u$ is a non-zero path derived from graph $G(\mathcal{B}), z$ is a non-zero path common to both $G(\mathcal{A})$ and $G(\mathcal{B})$ and 0 represents no path. Then consider the following cases:

$$d_{i,j} = u \Longrightarrow \left(a^*_{i,j} = u, b^*_{i,j} = 0\right) \therefore |\{u, 0\}| = 2 \therefore c_{i,j} \in (\{u, 0\} - \{0\}) \therefore c_{i,j} = u$$

$$d_{i,j} = v \Longrightarrow \left(a^*_{i,j} = 0, b^*_{i,j} = v\right) \therefore |\{0, v\}| = 2 \therefore c_{i,j} \in (\{u, 0\} - \{0\}) \therefore c_{i,j} = v$$

$$d_{i,j} = z \Longrightarrow \left(a^*_{i,j} = z, b^*_{i,j} = z\right) \therefore |\{z\}| = 1 \therefore c_{i,j} \in \{z\} \therefore c_{i,j} = z$$

$$d_{i,j} = 0 \Longrightarrow \left(a^*_{i,j} = 0, b^*_{i,j} = 0\right) \therefore |\{0\}| = 1 \therefore c_{i,j} \in \{z\} \therefore c_{i,j} = 0$$

Thus $c_{i,j} = d_{i,j}\ \forall(i,j)$ so $G(\mathcal{C}) = G(\mathcal{D})$ is a universal resultant graph for two sub-graphs.

■

So far we have only considered the case of two sub-graphs. Now we will look at generalising these results to finitely many sub-graphs.

12. JOINING FINITELY MANY GRAPHS

Given k subgraphs $\{G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)\}$ the expansion of the l^{th} script is given by:

$$\mathcal{A}^*_l = \left(A^*_l : a^l_{i,j} = \begin{cases} 0 & : f^{-1}_{\mathcal{A}_l}(i), f^{-1}_{\mathcal{A}_l}(j) \notin N(A_l) \\ a_{f^{-1}_{\mathcal{A}^*_l}\left(f^{-1}_{\mathcal{A}_l}(i)\right), f^{-1}_{\mathcal{A}^*_l}\left(f^{-1}_{\mathcal{A}_l}(j)\right)} & otherwise \end{cases}, \bigcup_{s=1}^{k} N(A_s) \right)$$

Theorem 4. Let $S = \{a^1_{i,j}, \dots, a^k_{i,j}\}$ then $G(\mathcal{C})$ is a universal resultant graph:

$$\mathcal{C} = \left(C : c_{i,j} = \begin{cases} x \in S & : |S| = 1 \\ x \in (S - \{0\}) & : |S| = 2 \end{cases}, \bigcup_{s=1}^{k} N(A_s) \right) \quad (1)$$

Proof. Let $G(\mathcal{D})$ be the resultant graph.

There are two cases to assess; $|S| = 1$ and $|S| = 2$. Note $|S| < 1 \Longrightarrow S = \emptyset$ which cannot occur and $|S| > 2 \Longrightarrow$ there are multiple paths from one node to another and this case has been restricted.

$|S| = 1 \Longrightarrow \left(a^1_{i,j} = \cdots = a^k_{i,j}\right) \therefore \left(d_{i,j} \in S\right) = (c_{i,j} \in S)$

$|S| = 2 \Longrightarrow S = \{P, 0\}$ where P is a non-zero path. This is true since the only alternative is $S = \{P, \bar{P}\}$ where $\bar{P}$ is a non-zero path distinct from P but then there would be multiple paths from one node to another, a case that has been restricted. Thus $d_{i,j} \in S - \{0\}$ and $c_{i,j} \in S - \{0\}$

Since in both cases $c_{i,j} = d_{i,j}\ \forall(i,j).\ G(\mathcal{C})$ is a universal resultant graph of the set of k sub-graphs $\{G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)\}$

■

13. MULTIPLE PATHS

If there are multiple paths from at least one node to another in sub-graphs $\{G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)\}$ then $|S| = \left|\left\{a^1_{\vec{\imath},\vec{\jmath}}, \dots, a^k_{\vec{\imath},\vec{\jmath}}\right\}\right| > 2$ for at least one pair of nodes $\vec{\imath}, \vec{\jmath}$.

Theorem 5. $G(\mathcal{C})$ is a universal resultant graph for sub-graphs $\{G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)\}$ that may contain multiple paths:

$$\mathcal{C} = \left(C: c_{i,j} = \begin{cases} x \in S^* & : |S^*| = 1 \\ x \in (S^* - \{0\}) : (|S^*| = 2) \bigcap (0 \in S^*) & \\ X = S^* : (|S^*| \geq 2) \bigcap (0 \notin S^*) & , \bigcup_{s=1}^{k} N(A_s) \\ X = S^* - \{0\} : (|S^*| > 2) \bigcap (0 \in S^*) & \end{cases} \right) \quad (2)$$

Where S^* removes all but the two outside braces from S. Thus S^* converts S into a set since $S = \{a^1_{i,j}, \dots, a^k_{i,j}\}$ but any element $a^n_{i,j} = \{h_1, \dots, h_m\}$ so $S = \{a^1_{i,j}, \dots, \{h_1, \dots, h_m\}, \dots, a^k_{i,j}\}$ and removing all but the outside braces $S^* = \{a^1_{i,j}, \dots, h_1, \dots, h_m, \dots, a^k_{i,j}\}$ leaves a list of elements $\therefore\ S^*$ is a set.

Proof. (2) reduces to (1) given $|S^*| = 1, (|S^*| = 2) \cap (0 \in S^*)$ so these cases have been proven. Cases $(|S^*| \geq 2) \cap (0 \notin S^*)$ and $(|S^*| > 2) \cap (0 \in S^*)$ must be proved. If $(|S^*| = 2) \cap (0 \notin S^*)$ then $S^* = \{P, \bar{P}\}$ contains two unique paths thus $d_{i,j} = S^* = c_{i,j}$. Given $(|S^*| > 2) \cap (0 \notin S^*)$ then $S^* = \{P_1, \dots, P_V\}$ contains V unique paths thus $d_{i,j} = S^* = c_{i,j}$. Given $(|S^*| > 2) \cap (0 \in S^*)$ then $S^* = \{P_1, \dots, P_V, 0\}$ contains V non-zero paths hence $d_{i,j} = (S^* - \{0\}) = c_{i,j}$. Thus $d_{i,j} = c_{i,j}$ in all cases so $G(\mathcal{C})$ is a universal resultant graph for finitely many sub-graphs $\{G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)\}$.

■

14. SPLITTING A GRAPH

Theorem 6. Given a graph $G(\mathcal{C}): \mathcal{C} = (C, N(C))$ we can generate k sub-graphs $\{G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)\}: \mathcal{A}_n = (A_n, N(A_n))$ with $N(C) = \bigcup_{h=1}^{k} N(A_h)$. Then an algorithm generating the n^{th} script is:

$$\mathcal{A}_n = \left(A_n: a_{i,j} = c_{f_c\left(f^{-1}_{\mathcal{A}_n}(i)\right), f_c\left(f^{-1}_{\mathcal{A}_n}(j)\right)}, N(A_n) \right) \quad (5)$$

Proof. $G(\mathcal{A}_n)$ is a well defined graph thus there exists a bijective function $f_{\mathcal{A}_n}: N(A_n) \to \{1, \dots, v\} : v = |N(A)|, f_{\mathcal{A}_n}(b_u) = u : u \in \{1, \dots, v\}$ with inverse $f_{\mathcal{A}_n}^{-1}(u) = b_u$. Consider $a_{i,j}$ corresponding to nodes $f_{\mathcal{A}_n}^{-1}(i)$ and $f_{\mathcal{A}_n}^{-1}(j)$ which correspond to the index $f_c\left(f_{\mathcal{A}_n}^{-1}(i)\right), f_c\left(f_{\mathcal{A}_n}^{-1}(j)\right)$ in C thus $a_{i,j} = c_{f_c\left(f_{\mathcal{A}_n}^{-1}(i)\right), f_c\left(f_{\mathcal{A}_n}^{-1}(j)\right)}$.

■

15. JOINING AND SPLITTING GRAPHS

There are various methods, as shown above, of how a given sub-graph specification, $G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)$, may be joined depending on its relationship category. For example if the sub-graph specification $S = G(\mathcal{A}_1), \dots, G(\mathcal{A}_K)$ is type 1 then methods in theorems 1 to 5 can be used to join such specification. Let $\mathcal{J}^* = \{\mathcal{J}_1, \mathcal{J}_2, \dots\}$ denote the set of all methods for joining sub-graphs. Only some of these methods can be applied to S, denote this set $\mathcal{J}_S$. Thus $\mathcal{J}_S \subset \mathcal{J}^*$. Analogously let $\mathcal{S}^* = \{\mathcal{S}_1, \mathcal{S}_2, \dots\}$ denote all methods of splitting a resultant graph. Then only some of these methods can be used to split a resultant into a particular sub-graph specification, denote this set $\mathcal{S}_S$. Thus $\mathcal{S}_S \subset \mathcal{S}^*$

Let $I(G(\mathcal{B}))$ denote the information conveyed in graph $G(\mathcal{B})$. That is, the nodes and the connections between them. Suppose $G(\mathcal{B})$ has a sub-graph specification $S = (G(\mathcal{A}_1), G(\mathcal{A}_2), G(\mathcal{A}_3))$ then $G(\mathcal{A}_1)\ and\ G(\mathcal{A}_2)$ can be joined to form a new graph, $G(\mathcal{A}_4)$, but this represents a new sub-graph specification $S' = (G(\mathcal{A}_4), G(\mathcal{A}_3))$. New sub-graph specifications can also be generated by restructuring information as long as the information the new sub-graph specification portrays is equivalent to $I(S)$. Let X be the set of all sub-graph specifications for graph $G(\mathcal{B})$ then $X = \{S_1, \dots, S_\lambda\}: I(S_i) = I\left(S_j\right), \forall i, j \in \{1, \dots, \lambda\}$. Denote S_0 the resultant graph for convenience.

For any sub-graph specification, S_i, there is an associated set of joining methods, denoted $\mathcal{J}_{S_i}^*$ and a set of splitting methods denoted $\mathcal{S}_{S_i}^*$ where $\mathcal{J}_{S_i}^* \subset \mathcal{J}^*$ and $\mathcal{S}_{S_i}^* \subset \mathcal{S}^*$. Thus given set $X = \{S_1, \dots, S_\lambda\}$ there exists $\mathcal{J} = \{\mathcal{J}_{S_1}^*, \dots, \mathcal{J}_{S_\lambda}^*\}$ and $\mathcal{S} = \{\mathcal{S}_{S_1}^*, \dots, \mathcal{S}_{S_\lambda}^*\}$.

Consider the following processes:

$$\mathcal{J}: X \to S_0: \mathcal{J}_{S_i}^*(S_i) = S_0$$

$$\mathcal{S}: S_0 \to X: \mathcal{S}_{S_i}^*(S_0) = S_i$$

Although these appear to be mappings that define a function, they are not, however they are analogous. They define the procedure of turning a sub-graph into the resultant and back.

Consider $\mathcal{J}_{S_i}^*\left(\mathcal{S}_{S_i}^*(S_0)\right) = S_0$ and $\mathcal{S}_{S_i}^*\left(\mathcal{J}_{S_i}^*(S_i)\right) = S_i$ then certain joining and splitting methods can be thought of as being inverse mechanisms.

Given that we can transition from a resultant graph, S_0, to a certain sub-graph specification, S_i, using joining and splitting methods we can choose a preferred specification to represent our information since all such specifications represent the same information by definition, only the way the information is structured differs. The question now becomes which specification $S_k : k \in \{0, \dots, \lambda\}$ contains the best structure?

16. DIMENSIONAL EFFICIENCY

One consideration would be the efficiency of a specification where the term efficiency corresponds to the number of elements needed to convey the required information.

Making specification S_k explicit: $S_k = \left(G_k(\mathcal{A}_1), \dots, G_k\left(\mathcal{A}_{\Omega(k)}\right)\right)$ where $\Omega(k) \in \mathbb{N}$ is the number of sub-graphs in specification S_k and $G_k(\mathcal{A}_i) = \left(A_{i,k}, N\left(A_{i,k}\right)\right)$ where $A_{i,k}$ corresponds to the adjacency matrix in the i^{th} graph in specification S_k and $N\left(A_{i,k}\right)$ its nodes.

The dimension of the i^{th} graph in specification S_k is $Dim[G_k(\mathcal{A}_i)] = |N\left(A_{i,k}\right)|^2 + |N\left(A_{i,k}\right)|$ which is the sum of the number of elements in its adjacency matrix and the number of elements in the node set. The dimension of specification S_k is given by:

$$Dim[S_k] = Dim\left[\left(G_k(\mathcal{A}_1), \dots, G_k\left(\mathcal{A}_{\Omega(k)}\right)\right)\right] = Dim[G_k(\mathcal{A}_1)] + \cdots + Dim\left[G_k\left(\mathcal{A}_{\Omega(k)}\right)\right]$$

$$Dim[S_k] = \sum_{i=1}^{\Omega(k)} Dim[G_k(\mathcal{A}_i)]$$

Let $x_{i,k} = |N\left(A_{i,k}\right)|$ then $Dim[G_k(\mathcal{A}_i)] = (x_{i,k})^2 + x_{i,k} = x_{i,k}\left(x_{i,k} + 1\right)$ thus

$$Dim[S_k] = \sum_{i=1}^{\Omega(k)} x_{i,k}\left(x_{i,k} + 1\right)$$

Finding the specification with the smallest dimension:

$$MIN_{(k)}\left[\sum_{i=1}^{\Omega(k)} x_{i,k}\left(x_{i,k} + 1\right)\right] = L$$

Specification $S_{k^*}$is optimal:

$$\sum_{i=1}^{\Omega(k^*)} x_{i,k^*}\left(x_{i,k^*} + 1\right) = L$$

There may be numerous specifications that are optimal. Let $k^* \in K \subset \mathbb{Z}$ be the set of numbers corresponding to the index of all optimal specifications.

E.g. If $\sum_{i=1}^{\Omega(3)} x_{i,3}\left(x_{i,3}+1\right) = \sum_{i=1}^{\Omega(5)} x_{i,5}\left(x_{i,5}+1\right) = L$ then specifications S_3 and S_5 are optimal thus $k^* \in \{3,5\} = K$.