Visualisation of Social Networks using CAVALIER

Anthony Dekker
C3 Research Centre
Defence Science Technology Organisation (DSTO)
Fernhill Park, Department of Defence
Canberra ACT 2600, Australia
dekker@ACM.org

Abstract

Social Network Analysis is an approach to analysing organisations focusing on relationships as the most important aspect. In this paper we discuss visualisation techniques for Social Network Analysis, including spring-embedding and simulated annealing techniques. We introduce a visualisation technique based on Kohonen neural networks, and also introduce social flow diagrams for demonstrating the relationship between two forms of conceptual distance.

Keywords: Social network analysis, Kohonen neural networks.

1 Social Network Analysis: Introduction

Social Network Analysis (Wasserman and Faust 1994) is an approach to analysing organisations focusing on the *relationships* between people and/or groups as the most important aspect. Going back to the 1950's and before, it is characterised by adopting mathematical techniques especially from *graph theory* (Gibbons 1985, Krackhardt 1994). It has applications in organisational psychology, sociology and anthropology.

The first goal of Social Network Analysis is to *visualise* communication and other relationships between people and/or groups by means of diagrams. Visualisation of Social Networks has a long tradition, and an excellent historical survey is given in Freeman (2000). The importance of visualisation in this field lies in the complexity of organisational structure, and the need for good visual representations of how an organisation functions.

The second goal is to study the *factors which influence relationships* (for example the age, background, and training of the people involved) and to study the *correlations* between relationships. This can be done using traditional statistical techniques such as correlation, analysis of variance, and factor analysis (Cohen et al 1996), but also requires appropriate visualisation techniques.

The third goal of Social Network Analysis is to draw out *implications* of the relational data, including bottlenecks where multiple information flows funnel through one person or section (slowing down work processes) and situations where information flows does not match formal group structure.

The fourth and most important goal of Social Network Analysis is to make *recommendations* to improve communication and workflow in an organisation, and (in military terms) to speed up the observe-orient-decide-act (OODA) loop or decision cycle (Allard 1996). Visualisation is critically important in presenting these recommendations to clients.

In previous work, we have applied Social Network Analysis to military organisations (Dekker 2000). In this paper we present a number of visualisation techniques that we have developed in the course of this work. We have constructed a Java-based tool suite called CAVALIER (Communication and Activity VisuALIsation for the EnteRprise), to carry out Social Network Analysis, and the visualisation techniques that we describe are incorporated within that tool.

2 Spring Embedding

One of the most common techniques for visualising Social Networks is spring-embedding (Freeman 2000). A spring-embedding layout algorithm assumes that links between nodes behave physically like springs, with an ideal spring *length* (that corresponds to some kind of conceptual distance between the nodes), and a spring *strength* (best results are obtained when this decreases as the ideal spring length increases). The nodes can be assigned to points in two-dimensional or three-dimensional space by moving them in a way which minimises the total stress in the entire collection of strings, using straightforward physics.

The major case study we use in this paper (illustrated in figures 1 to 4) involved a military C3I-related organisation seeking scientifically based advice to a reorganisation process. Data was collected during March/April 2000 using electronic questionnaires which included questions on communication and areas of staff interest. We have also conducted similar studies in other organisations.

Figure 1 visualises communication between people in that organisation, using spring-embedding. The organisation consisted of six main sub-units (labelled "A" to "F" in the figure) as well as a number of special executive and liaison staff (labelled "G" in the figure). Extensive communication took place between all groups, but the strongest communication links were between the three groups "A," "E" and "F." Moderately strong links also existed between the four groups "C," "D," "E" and "F." Weaker, but still significant, links were "A–B," "A–C" and "A–D." Communication between people was coded pseudo-logarithmically as follows:

- 1.0 = three or more times per day
- 0.8 = once per day
- 0.6 = three or more times per week
- 0.4 = once per week
- 0.2 = once per fortnight
- 0.0 = less than once per fortnight

The 180-degree communication correlation (i.e. correlation between reported communication to and fro) was $r = 0.61$ ($r^2 = 0.37$), which is typical for surveys of this kind (since people have different recollections of the amount of communication between them).

We take the single-link distance between two people to be the reciprocal of the larger of the two numbers reporting communication between them (thus ranging from 1 to infinity). The conceptual distance between two arbitrary people is then the length of the shortest path between them (ranging from 1 to 7.25 in this case).

This definition of conceptual distance between people does not take into account the number of different paths between people, but it has a number of advantages:

- It can be computed efficiently.
- Distances do not change very much if some people fail to complete survey forms (a serious problem when survey participation is voluntary).
- It correlates extremely well with physical distance in spring-embedding, and hence is easily visualised.
- In simulation experiments, it correlates well with information propagation time (typical correlations are in the range 0.7 to 0.8, with r^2 in the range 0.5 to 0.7).

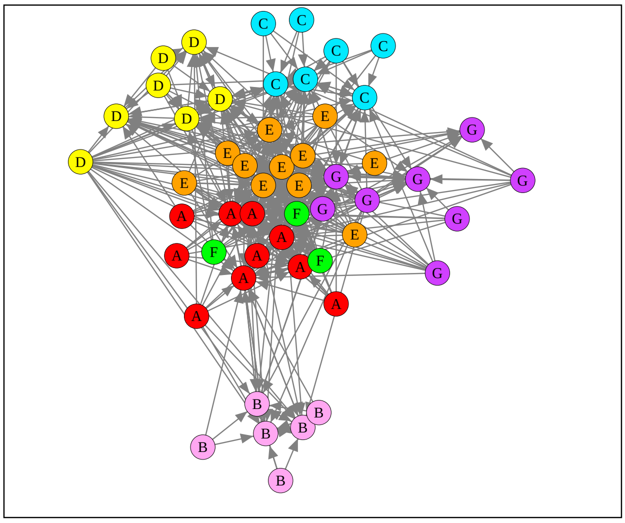

Figure 1. Social Network for a Military Organisation: Spring-Embedding Layout

In figure 1, spring-embedding is used with ideal spring lengths equal to the conceptual distance between people. The resulting correlation between physical distance in the diagram and conceptual distance is 0.85 ($r^2 = 0.72$).

Figure 2 is the result of spring-embedding in three-dimensional space. This results in a much more accurate depiction of conceptual distance. In particular, the correlation between physical distance in the three-dimensional diagram and conceptual distance is 0.93 ($r^2 = 0.86$). Such a three-dimensional diagram can be difficult to interpret, however, and many participants in our studies have reported difficulty in interpreting such diagrams. The inclusion of links in the diagram provides a valuable sense of perspective, but also obscures the nodes in the rear. We have had greater success with *interactive* three-dimensional diagrams using VRML (Virtual Reality Modelling Language) technology. The ability to manipulate the three-dimensional model increases understanding of its structure, and VRML is easily deployed using Web technologies. VRML also allows easy linking of explanatory text to nodes. However, VRML is somewhat difficult for first-time users, and installing VRML browsers is also difficult for the inexperienced computer user.

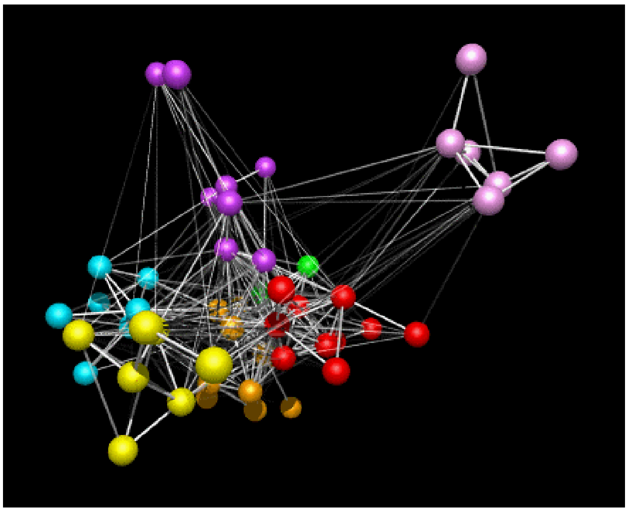

Figure 2. Social Network for a Military Organisation: Three-Dimensional Spring-Embedding Layout

3 Alternative Layout Algorithms

Both two-dimensional and three-dimensional spring-embedding layouts share some limitations for our purposes. In particular, they may place nodes too close together, which creates difficulties when the nodes must be labelled. Also, by reflecting the conceptual distance so accurately, they can paradoxically over-emphasise large conceptual distances. For example, the nodes labelled "B" at the bottom of figure 1 (also shown at the top right of figure 2) are separated significantly from the nodes representing the rest of the organisation, and this causes an immediate reaction when presented to case study participants. However, such an immediate reaction (which may go so far as to result in a restructuring of the organisation) may be inappropriate, since the large conceptual distances may reflect limitations in the data, or some non-obvious phenomenon.

One solution to these problems is to use spring-embedding based on a transformation of conceptual distance, such as the square root, or to use force-based layout techniques (Brandes 2001). However, we have also investigated a number of alternative layout techniques. In figure 3, staff are placed in a circle with senior staff central and junior staff towards the outside. *Simulated annealing* (Hecht-Nielsen 1990) was used to find a configuration which placed together staff who communicated most with each other. This is a proven technique for solving very difficult optimisation problems (or at least of finding near-perfect solutions). The origin of this technique lies in mathematical analysis of metals which are heated and very slowly cooled, resulting in a near-perfect crystal structure. Applications include modern chip-design software, which uses simulated annealing to minimise chip area and maximise clock speed.

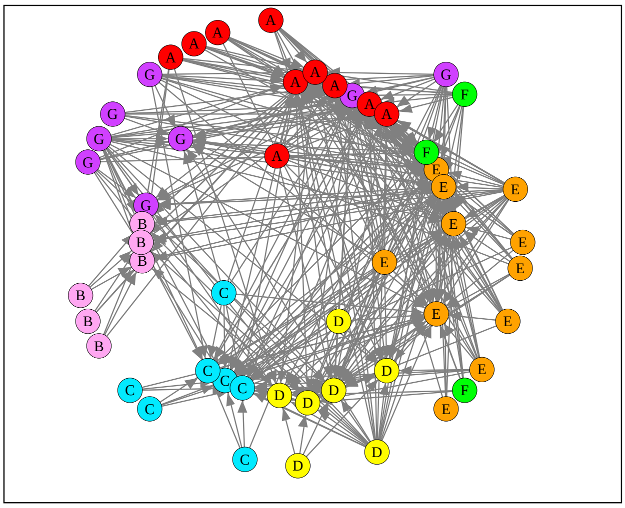

Figure 3. Social Network for a Military Organisation: Circle Layout

The simulated annealing algorithm we use (algorithm 1) is shown below. In producing figure 3, initial positions had radii based on seniority (small for senior staff and large for junior staff) and arbitrary evenly-spaced angles (with respect to the central point). The energy factor used in the algorithm can be the inverse of the correlation between physical and conceptual distance, or some similar quantity that decreases as quality increases. The algorithm attempts to minimise this energy by repeated swapping operations. In the case of figure 3, swapping means exchanging angles while leaving radii unchanged. The algorithm is also effective where initial positions are arranged in a regular grid, in which case swapping is interpreted literally.

Our version of simulated annealing converges fairly rapidly, in contrast to simulated annealing based on random positional changes, which converges much more slowly. However, our version suffers from the limitation that the range of possible node positions is limited *a priori* to a fairly small set. For the circle layout in figure 3, this is not a severe limitation, but for layouts on a regular grid the limitation is more serious.

input: set of nodes N of size n with distance function d
output: spatial layout of N
$T = T_{\max}$;
E = energy;
while $T > T_{\min}$
    randomly choose a pair of nodes p and q;
    $E_{\text{old}} = E$;
    swap positions of p and q;
    E = energy;
    if $E_{\text{old}} < E$ then
        with probability $1 - \exp((E_{\text{old}} - E)/T)$ undo swap;
    decrease T;
end while

Algorithm 1: Algorithm for Simulated Annealing Graph Layout

In figure 3, the correlation between physical distance in the diagram and conceptual distance is only 0.36 (r^2 = 0.13), but the placement of groups (groups “C,” “D,” “E,” and “F” together, “A” near “E” and “F,” and “B” near “A”) is similar to figure 1, i.e. topology is preserved even if exact distances are not. This diagram was found to be quite helpful in presenting study results to the client.

4 Layout using Kohonen Neural Networks

Figure 4 is produced using a self-organising (Kohonen) neural network. Kohonen neural networks (Kohonen 1989, Hecht-Nielsen 1990, Ritter *et al* 1992) are a form of self-organizing neural network which produce topological mappings. They have been applied to areas such as pattern recognition (Kohonen 1990), the learning of ballistic movements (Ritter and Schulten 1989), modelling aerodynamic flow (Hecht-Nielsen 1988), and image colour quantization (Dekker 1994).

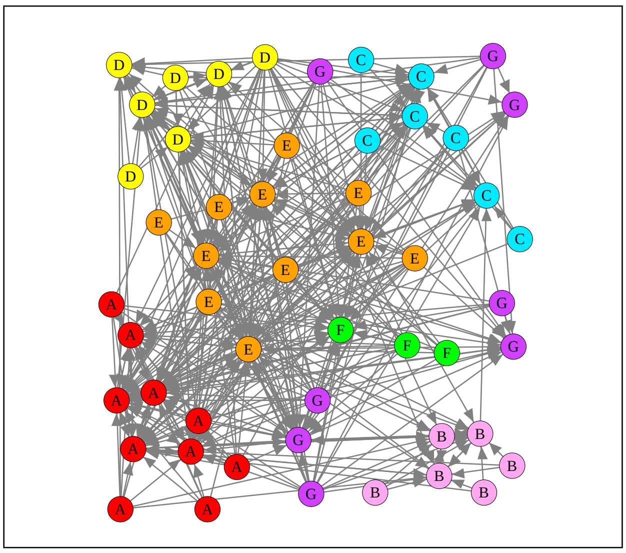

Figure 4. Social Network for a Military Organisation: Kohonen Neural Network Layout

We use an adaptation of the basic Kohonen neural network presented in Dekker (1994). In that work, one-dimensional networks are used, but the extension to arbitrary graphs is easy.

The basic algorithm (algorithm 2) is shown below. The algorithm is controlled by two parameters: a factor α in the range 0…1, and a radius r, both of which decrease with time. We have found that the algorithm works well if the main loop is repeated 1,000,000 times. The algorithm begins with each node assigned to a random position. At each step of the algorithm, we choose a random point within the region that we want the network to cover (in the case of figure 4, a rectangle), and find the closest node (in terms of *Euclidean distance*) to that point. We then move that node towards the random point by the fraction α of the distance. We also move nearby nodes (those with *conceptual distance* within the radius r) by a lesser amount.

input: set of nodes N of size n with distance function d
output: spatial layout of N
$r = 12$;
$\alpha = 1$;
repeat many times
  choose random (x,y);
  i = index of closest node;
  move node i towards (x,y) by α;
  move nodes with $d<r$ towards (x,y) by $\alpha\times(1-d^2/r^2)$;
  decrease α and r;
end repeat

Algorithm 2: Basic Algorithm for Kohonen Neural Network Layout

Figure 5 illustrates this procedure at a point where α = 0.6666 and the radius r = 4. Dark circles show the network before update, and lighter circles show the updated network. The random point is shown by a square. Numbers show conceptual distances between nodes.

Kohonen (1989) suggests decreasing α and r linearly over time, but we have found that results are improved and the required iteration time reduced if both are decreased *exponentially*, by multiplying by 0.98 at regular intervals during the main loop, so that the final values are $\alpha = 0.02$ and $r = 1$.

The improved algorithm (algorithm 3) incorporates a modification due to Desieno (Hecht-Nielsen 1990, p 69) which significantly improves performance. A bias factor $b[j]$ is subtracted from the distances to the random point at each step, based on an estimate of the frequency with which nodes have been moved in the past ($f[j]$ in the algorithm).

A graph layout algorithm similar to our algorithm 2 was introduced by Meyer (1998). However it is subject to "clashes" which the Desieno modification helps avoid.

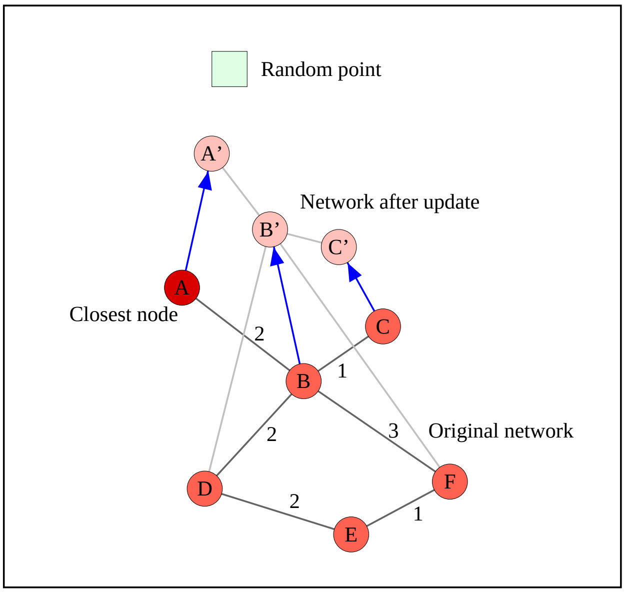

Figure 5. Update Step for Kohonen Neural Network Layout Algorithm

The final result of the algorithm is a positioning of nodes which balances the placement of conceptually close nodes together with an approximately evenly-based distribution of nodes within the specified region. The latter property is of benefit when the nodes are labelled with names of people. It is also possible to conduct the learning process with random points drawn from an arbitrary non-rectangular shape, and thus overlay a social network on e.g. an office building layout.

input: set of nodes N of size n with distance function d
output: spatial layout of N
$\beta = 0.001$;
$\gamma = 2000$;
$b[1\ldots n] = 0$;
$f[1\ldots n] = 1/n$;
invariant: for $j \in 1\ldots n$, $b[j] = \gamma \times ((1/n) - f[j])$
$r = 12$;
$\alpha = 1$;
repeat many times
  choose random (x,y);
  i = index of node with minimum of distance $- b[i]$;
  for $j \in 1\ldots n$, $b[j] = b[j] + \beta\times\gamma\times f[j]$;
  for $j \in 1\ldots n$, $f[j] = f[j] - \beta\times f[j]$;
  move node i towards (x,y) by α;
  move nodes with $d<r$ towards (x,y) by $\alpha\times(1-d^2/r^2)$;
  $b[i] = b[i] - \beta\times\gamma$;
  $f[i] = f[i] + \beta$;
  decrease α and r;
end repeat

Algorithm 3: Improved Algorithm for Kohonen Neural Network Layout

In figure 4, the correlation between physical distance in the diagram and conceptual distance is 0.50 (r^2 = 0.25), which is better than the circle layout in figure 3. The placement of groups is similar to figures 1 and 3. Comparing figures 1 and 4 shows that figure 4 is very close to a topological (continuous) distortion of figure 1. This is due to the fact that Kohonen neural networks create topological mappings.

We have found this kind of diagram extremely useful, particularly when documenting the results of a Social Network analysis survey.

The decrease within the algorithm of the radius over time means that the algorithm effectively moves progressively smaller "chunks" of the network together, i.e. it is inherently *hierarchical*, but without explicitly specifying clustering as in Brockenauer and Cornelsen (2001). This makes the algorithm highly effective for visualising *tree structures*. Figure 6 shows a tree structure (an organisational chart) laid out using the Kohonen algorithm.

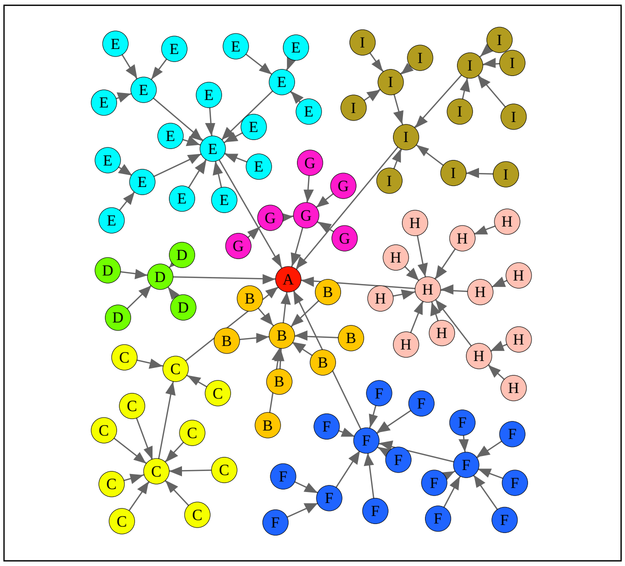

Figure 6. Kohonen Neural Network Layout for a Tree Structure

5 Conceptual Spaces

The electronic questionnaire in this case study included information about 15 topics of interest, many of which were found to be highly correlated with each other. Factor analysis (principal component analysis) on these topics found that 4 main factors (principal components) explained 73 percent of the variation in data about interests. The first of these factors simply represented people's general level of interest in everything. Figure 7 illustrates the other three principal components — people are located in a three-dimensional conceptual space according to the numerical values of these three principal components. In other words, the diagram places people with similar interest together, and in most cases, these are people in the same group.

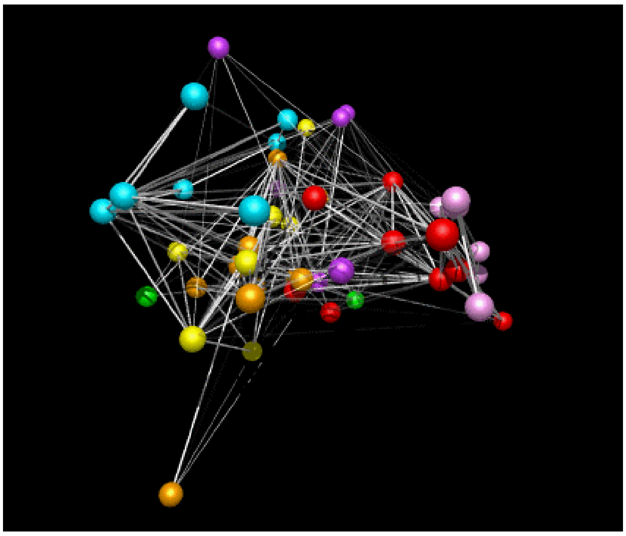

Figure 7. Social Network for a Military Organisation: Three-Dimensional Factor Layout

This kind of diagram was found to be quite deceptive if only two factors were presented (in a two-dimensional diagram), since that could place people with quite different interests together. Two-dimensional diagrams of this kind were frequently misunderstood by clients. Three-dimensional diagrams like figure 7 were more successful (provided that there were only three major factors), but suffered from the same problems noted for figure 2 above.

6 Social Flow Diagrams

Figure 8 shows an example social network diagram produced by the CAVALIER tool, based on the ground force structure during the Gulf War (Clancy and Franks 1999, Khaled and Seale 1995). Boxes represent division-level units from participating countries (in the case of Saudi, Kuwaiti, and Gulf state units, these are notional), while circles represent commanders. Units on the right were under the control of American General Norman Schwarzkopf, while those on the left were under the control of Saudi Prince Khaled bin Sultan.

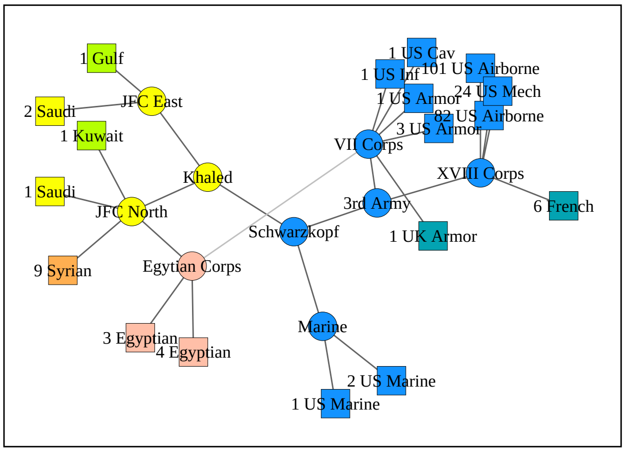

Figure 8. Social Network for Gulf War Ground Forces: Spring-Embedding Layout

Figure 8 is produced by a spring-embedding layout algorithm which attempts to balance two forms of distance: cultural distance and command distance. Cultural distances range from $^1/_8$ for units from the same country and service to 6 for the less than friendly relationship between the US and Syria. Cultural differences between the US Army and Marines are reflected by a distance of $^1/_2$. Dark grey lines in the figure show formal command relationships, and command distance is measured by counting the minimum number of these links. The light grey line between the US VII Corps commander and the Egyptian Corps commander represents an informal working relationship, which we represent using a command distance of 2.

We are particularly interested in comparisons between two or more distance relationships, such as occur in this example. When all pairs of division-level units in figure 8 are considered, there is a statistical correlation of 0.58 (r^2 = 0.33) between the cultural distance and command distance. This indicates that the organisational structure negotiated between the US and Saudi Arabia was fairly successful in separating culturally different units.

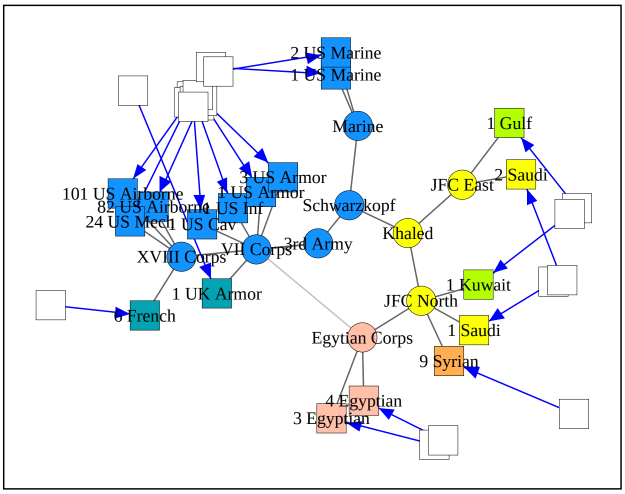

Figure 9. Social Flow Diagram for Gulf War Ground Forces

The relationship between these two distance concepts is visualised in figure 9, which we call a *social flow diagram*. In this figure, each division-level unit is represented by a pair of boxes (one white, one coloured) linked by an arrow. As a result of the spring-embedding layout algorithm, the physical distance between white boxes closely indicates cultural distance (physical distance has a 0.97 correlation with cultural distance), while the physical distance between coloured boxes indicates command distance (somewhat less closely, with a correlation of 0.86). The arrows indicate how culturally similar units have been separated in some cases, and culturally dissimilar units have been combined in others. For example, in the lower left of the figure, the French division, which was initially strongly opposed to being under US control, has in fact been placed within the US command structure. In the upper left, the US Marines have separated themselves from their Army colleagues. On the other hand, in spite of a somewhat deceptive long arrow in the diagram, figure 9 shows that the UK forces remained closely tied to the US Army.

Figure 10 uses spring-embedding to visualise a slightly different definition of cultural distance between selected countries. The definition of cultural distance used here combines differences in religion, language, economics, and military alliances such as NATO (the correlation with physical distance is 0.85).

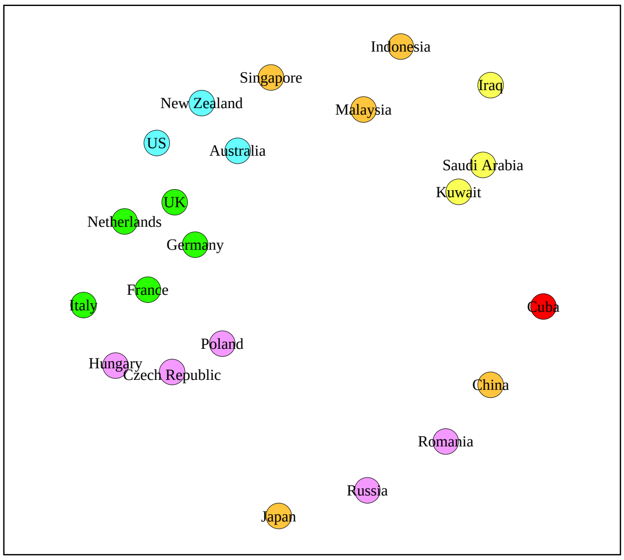

Figure 10. Cultural Distances for Selected Countries

Figure 11 is a social flow diagram which visualises the change in this definition of cultural distance after the end of the Cold War. White boxes represent the situation during the Cold War, while coloured circles represent the present situation (also shown in figure 10). The correlation with physical distance is 0.85 for the Cold War situation, and 0.81 for the present. The top left of the diagram shows how some former Communist countries have moved closer to Western Europe, while others have not.

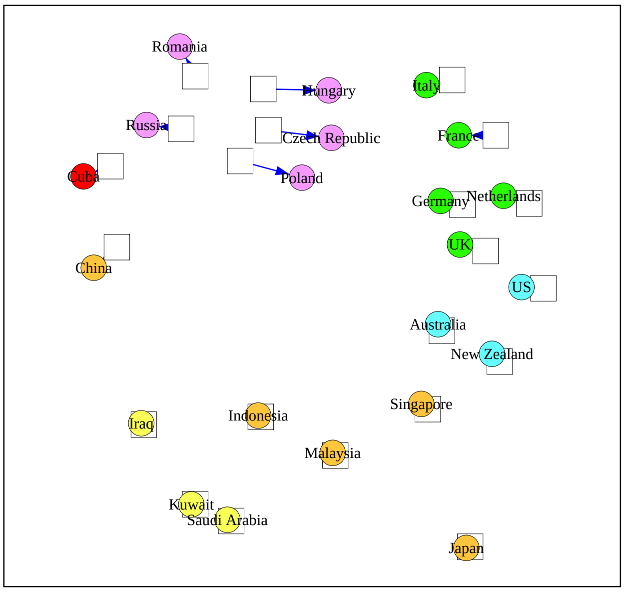

Figure 11. Social Flow Diagram for End of Cold War

We produce social flow diagrams using spring-embedding, giving the introduced arrows a very short desired *length*, but a very low *strength*, so that a wide variation in arrow lengths is tolerated by the spring-embedding algorithm.

We have found social flow diagrams to be very effective in visualising how conceptual distance is affected by factors such as similarity of tasking, both in relationships between groups (as in figures 9 and 11) and in relationships between individual people.

7 Conclusion

We have discussed the use of visualisation for Social Network Analysis within the CAVALIER (Communication and Activity VisuALIsation for the EnteRprise) tool suite, including spring-embedding and simulated annealing techniques. We have introduced a visualisation technique based on Kohonen neural networks, and we have also introduced *social flow diagrams* for demonstrating the relationship between two forms of conceptual distance.

8 Acknowledgements

Dawn Hayter and Jon Rigter provided valuable suggestions on this research project. The CAVALIER software included the JAMA linear algebra module from the US National Institute of Standards and Technology (NIST) which has been released to the public domain. Figures 2 and 7 were produced using the *Persistence of Vision*™ (*POV-Ray*™ version 3.1) ray-tracing package. Thanks also to an anonymous referee who provided several useful comments, including pointing out the work of Meyer (1998).

9 References

ALLARD, K. (1996): *Command, Control, and the Common Defense, revised edition*, National Defense University Press, Washington, D.C.

BRANDES, U. (2001): Drawing on Physical Analogies. In *Drawing Graphs: Methods and Models*, 71–86, KAUFMANN, M. and WAGNER, D. (eds), Springer Verlag LNCS **2025**.

BROCKENAUER, R. and CORNELSEN, S. (2001): Drawing Clusters and Hierarchies. In *Drawing Graphs: Methods and Models*, 193–227, KAUFMANN, M. and WAGNER, D. (eds), Springer Verlag LNCS **2025**.

CLANCY, T. and FRANKS, F. (1999): *Into the Storm: A Study in Command*, Sedgwick and Jackson.

COHEN, R. J., SWERDLIK, M. E. and PHILLIPS, S. M. (1996): *Psychological Testing and Assessment, 3rd edition*, Mayfield.

DEKKER, A. H. (1994): Kohonen Neural Networks for Optimal Colour Quantization. *Network: Computation in Neural Systems* **5**: 351–367, Institute of Physics Publishing. See also the associated web site at http://www.acm.org/~dekker/NEUQUANT.HTML

DEKKER, A. H. (2000): Social Network Analysis in Military Headquarters using CAVALIER. *Proc. 5th International Command and Control Research and Technology Symposium*, Canberra. Available at http://www.dodccrp.org/2000ICCRTS/cd/papers/Track6/039.pdf

FREEMAN, L. C. (2000): Visualizing Social Networks. *Journal of Social Structure* **1**(1). Available at http://www.heinz.cmu.edu/project/INSNA/joss/vsn.html

GIBBONS, A. (1985): *Algorithmic Graph Theory*, Cambridge University Press.

HECHT-NIELSEN, R. (1988): Neurocomputing: picking the human brain. *IEEE Spectrum* **25** (3): 36–41.

HECHT-NIELSEN, R. (1990): *Neurocomputing*, Addison Wesley.

KHALED BIN SULTAN and SEALE, P. (1995): *Desert Warrior: A Personal View of the Gulf War by the Joint Forces Commander*, Harper Collins.

KOHONEN, T. (1989): *Self-Organization and Associative Memory, 3rd edition*, Springer-Verlag, Berlin.

KOHONEN, T. (1990): Pattern Recognition by the Self-Organizing Map. *Parallel Architectures and Neural Networks: Third Italian Workshop*, Vietri sul Mare, Salerno, 13–18, CAIANIELLO, E. R. (ed), World Scientific, Singapore.

KRACKHARDT, D. (1994): Graph Theoretical Dimensions of Informal Organizations. In *Computational Organization Theory*. 89–111. CARLEY, K. M. and PRIETULA, M. J. (eds). Lawrence Erlbaum Associates, Hillsdale, NJ.

MEYER, B (1998): Self-Organizing Graphs: A Neural Network Perspective of Graph Layout. *Proc. Graph Drawing '98*, Montreal, Canada, 246–262, Springer Verlag LNCS **1547**. See also the associated web site at http://www.csse.monash.edu.au/~berndm/ISOM/index.html

RITTER, H. and SCHULTEN, K. (1989): Extending Kohonen's Self-Organizing Mapping Algorithm to Learn Ballistic Movements. In *Neural Computers*, 393–406, ECKMILLER, R. and v.d. MALSBURG, C. (eds), Springer-Verlag, Berlin.

RITTER, H., MARTINETZ, T. and SCHULTEN, K. (1992): *Neural Computation and Self-Organizing Maps: An Introduction*, Addison-Wesley.

WASSERMAN, S. and FAUST, K. (1994): *Social Network Analysis: Methods and Applications*, Cambridge University Press.