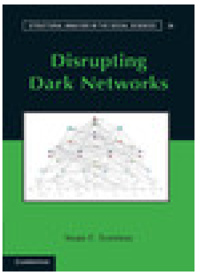


Disrupting Dark Networks

Sean F. Everton

Book DOI: http://dx.doi.org/10.1017/CBO9781139136877

Online ISBN: 9781139136877

Hardback ISBN: 9781107022591

Paperback ISBN: 9781107606685

Chapter

10 - Dynamic Analyses of Dark Networks pp. 319-342

Chapter DOI: http://dx.doi.org/10.1017/CBO9781139136877.015

Cambridge University Press

10

Dynamic Analyses of Dark Networks

10.1 Introduction

Networks are dynamic. They evolve and change over time as actors enter, leave, and move around. Capturing dynamics such as these and others can be difficult but it is possible. This chapter's purpose is to introduce readers to some of these approaches and demonstrate how to carry out relatively painless but potentially illuminating explorations of dynamic network data. We begin by examining approaches for exploring longitudinal networks both descriptively and statistically, although we do not consider in any great detail highly sophisticated approaches to the analysis of longitudinal network data, such as the actor-based models implemented by *SIENA* software[1] (see, however, Murphy, Everton, and Cunningham 2012). A comprehensive exploration of these models deserves its own book and requires software other than UCINET, Pajek, or ORA. Next we turn to the fusion of social network and geospatial data, which allows analysts to not only geospatially plot social network data but also to calculate geospatially weighted metrics; both features complement existing social network techniques in helpful ways.

10.2 The Longitudinal Analysis of Dark Networks

Historically, longitudinal network data have been difficult to come by and the methods for examining them have been underdeveloped. In recent years, this situation has begun to change. Longitudinal network data and their analyses are becoming more common. Many of these have been largely descriptive in nature, but they are increasingly becoming more sophisticated, employing model-based approaches that seek to identify the underlying mechanisms of network change (Breiger, Carley, and

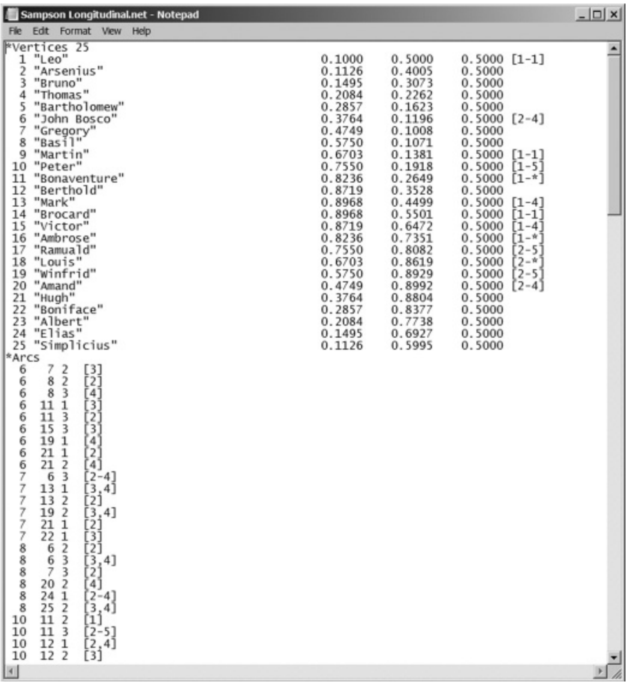
[1] See http://www.stats.ox.ac.uk/~snijders/siena/.

Pattison 2003; de Nooy 2011; Doreian and Stockman 1997; McCulloh and Carley 2011; Snijders 2005; Snijders, Bunt, and Steglich 2010; Steglich, Snijders, and Pearson 2010).

To date, the majority of longitudinal analyses have focused on "bright" or "light" networks. Only a handful of scholars have applied them to dark networks. One example is an analysis of the global Salafi jihad by Xu, Hu, and Chen (2009), which found that it not only evolved into a scale-free network but also appears to have passed through three distinct phases – emerging, maturing, and disintegrating. Another, which builds on the work done by Kossinets and Watts (2006), is the study of a co-offending network (Hu, Kaza, and Chen 2009), which discovered that acquaintances and shared vehicle affiliations served as key facilitators of tie formations, whereas age, race, and gender did not. More recently, McCulloh and Carley (2011) applied social network change detection (SNCD) methods to a number of longitudinal networks, including the Al Qaeda communication network from 1988 to 2004. Their analysis identified events that may have caused sudden changes in the networks they studied, including Al Qaeda.

Unfortunately, few would consider most of these newly developed modeling techniques easy to use; in fact, most of these approaches require specialized software, which is why descriptive approaches are still quite popular and why we feature them here. SNCD is an exception to this general rule. It is relatively easy to use and implemented in ORA, so we will explore it here, as well.

We begin with relatively simple longitudinal network data: the Sampson (1968) monastery data. If you recall, Sampson observed the social interactions of a group of monks and collected several sociometric rankings over a period of time. During his stay, a "crisis in the cloister" developed that resulted in the expulsion of four monks and the departure of several others; in the end, only four remained. Here, we will only focus on the positive measures of liking that Sampson recorded (adapted from de Nooy et al. 2005:93–95). These range from "3" (or "−3"), which indicates the highest or first choice; to "1" (or "−1"), which indicates the last choice; and "0," which indicates no choice. Sampson recorded measures of liking at five different points in time. At time one (T1), the group primarily consisted of novices who soon left to study elsewhere. Time two (T2) captures the period of time shortly after the arrival of several newcomers. Time three (T3) is the period during which one of the newcomers organized a meeting to discuss the monastery's situation. This meeting contributed to the polarization of the community, which ultimately caused the expulsion of four novices, an event that occurred one week after time four (T4), and which led several novices to leave the monastery shortly thereafter. At time five (T5), only seven of the eighteen novices still lived in the monastery.



Sampson Longitudinal.net - Notepad

File Edit Format View Help

```
*Vertices 25
  1 "Leo"                  0.1000    0.5000    0.5000 [1-1]
  2 "Arsenius"             0.1126    0.4005    0.5000
  3 "Bruno"                0.1495    0.3073    0.5000
  4 "Thomas"               0.2084    0.2262    0.5000
  5 "Bartholomew"          0.2857    0.1623    0.5000
  6 "John Bosco"           0.3764    0.1196    0.5000 [2-4]
  7 "Gregory"              0.4749    0.1008    0.5000
  8 "Basil"                0.5750    0.1071    0.5000
  9 "Martin"               0.6703    0.1381    0.5000 [1-1]
 10 "Peter"                0.7550    0.1918    0.5000 [1-5]
 11 "Bonaventure"          0.8236    0.2649    0.5000 [1-*]
 12 "Berthold"             0.8719    0.3528    0.5000
 13 "Mark"                 0.8968    0.4499    0.5000 [1-4]
 14 "Brocard"              0.8968    0.5501    0.5000 [1-1]
 15 "Victor"               0.8719    0.6472    0.5000 [1-4]
 16 "Ambrose"              0.8236    0.7351    0.5000 [1-*]
 17 "Ramuald"              0.7550    0.8082    0.5000 [2-5]
 18 "Louis"                0.6703    0.8619    0.5000 [2-*]
 19 "Winfrid"              0.5750    0.8929    0.5000 [2-5]
 20 "Amand"                0.4749    0.8992    0.5000 [2-4]
 21 "Hugh"                 0.3764    0.8804    0.5000
 22 "Boniface"             0.2857    0.8377    0.5000
 23 "Albert"               0.2084    0.7738    0.5000
 24 "Elias"                0.1495    0.6927    0.5000
 25 "Simplicius"           0.1126    0.5995    0.5000
*Arcs
  6   7 2  [3]
  6   8 2  [2]
  6   8 3  [4]
  6  11 1  [3]
  6  11 3  [2]
  6  15 3  [3]
  6  19 1  [4]
  6  21 1  [2]
  6  21 2  [4]
  7   6 3  [2-4]
  7  13 1  [3,4]
  7  13 2  [2]
  7  19 2  [3,4]
  7  21 1  [2]
  7  22 1  [3]
  8   6 2  [2]
  8   6 3  [3,4]
  8   7 3  [2]
  8  20 2  [4]
  8  24 1  [2-4]
  8  25 2  [3,4]
 10  11 2  [1]
 10  11 3  [2-5]
 10  12 1  [2,4]
 10  12 2  [3]
```

Figure 10.1. Partial Listing of Sampson.net

Once we are comfortable working with a small dataset, we will move to a larger one (Noordin's alive operational network), examining it first in Pajek and then in ORA (UCINET does not have the capabilities that Pajek and ORA do for generating and examining networks over time).

Longitudinal Networks in Pajek (Sampson Data)

Pajek has implemented helpful functions for analyzing longitudinal network data. Figure 10.1 shows part of the network file `Sampson.net`. In Chapter 4 we noted that Pajek structures network data as edge lists, so here we will focus on the time indicators in square brackets that are added to each actor and arc (or edge).[2]

Note the time code to the right of Leo's name and his associated spatial coordinates (i.e., [1–1]). This indicates that he was at the monastery at time one but left before time two. For time codes associated with actors, Pajek assumes that a time indicator remains valid until it encounters a new one (e.g., John Bosco, who was at the monastery from time two to time four and left before time five). In other words, as the file is currently

[2] These can be entered in various ways: The time codes for the Noordin data were entered by hand after reading the *.net file into a text editor (e.g., Notepad).

coded, not only was Leo at the monastery at time one and left before time two, but the same is also true of Arsenius, Bruno, Thomas, and Bartholomew. Similarly, Gregory and Basil were at the monastery from time two to time four and left before time five, just like John Bosco. Note the "*" included in the time stamp for Bonaventure. This indicates that he was present at time one and stayed beyond time five. In other words, a "*" is synonymous with infinity.

Unlike time records for actors, there must be one for each arc (or edge). In other words, with arcs (and edges) Pajek does not assume that a time indicator remains valid until it encounters a new one. What do these time codes tell us? Well, we can see that at time two, John Bosco (node 6) indicated Gregory (node 7) as his second choice in terms of people whom he liked. Note that he does not choose Gregory at times three or four, however (remember he leaves before time five). Interestingly, Gregory has positive feelings toward Bosco during periods two through three, as indicated by the time record (i.e., [2–4]) to the right of the arc from node 7 to node 6 (note that Bosco is Gregory's first choice in terms of liking in all three periods).

[Pajek] File>Network>Read

Net>Transform>Generate in Time>All Only Different Interval

To see how to generate longitudinal networks in Pajek, first open the Sampson network file (`Sampson.net`) using Pajek's *File>Network>Read* command. Next, use the *Net>Transform>Generate in Time* command. This calls up a series of dialog boxes that offer several options for generating a series of networks. First, you can choose to generate a network for each period (option *All*), a network only if it differs from the previous one (option *Only Different*), or a slice of the network that spans a specific time interval (*Interval*). The second option is useful if a network does not change much over time. Whichever command you choose, you will next need to specify the first and last time point that you want to analyze, as well as the time interval (step) between successive networks. For this example, choose the *All* option, start at time one, stop at time five, and choose a step value of one (step values must be positive integers). Note that the number of actors changes in the generated networks when actors enter or leave the network. As a result, the partition corresponding to the original longitudinal network may not match the newly generated networks. However, if a corresponding partition appears in the first Partition drop-down menu at the time that you issue the *Generate in Time* command, Pajek automatically creates new partitions for each generated network.

Table 10.1 illustrates how one might compare a network over time. Because the number of choices individuals could make was fixed by Sampson, measures of density and average degree would not tell us too much in this case. However, the network's size and centralization level could capture some of the dynamics occurring in the monastery during Sampson's stay. As you can see, the network grew from thirteen novices at time

Table 10.1. *Comparison of various metrics over time*

	Time period				
	1	2	3	4	5
Size	13.00	18.00	18.00	18.00	7.00
Degree Centralization	14.77	19.48	18.75	9.19	23.33
Closeness Centralization	30.75	41.62	34.68	24.89	60.24

one, to eighteen novices at times two through four, and then dropped to seven novices at time five. The network's centralization measures vary dramatically from one time point to the next. In terms of both degree and closeness centralization, the network became more centralized from time one to time two, became somewhat less so at time three (at the time of the meeting to discuss what was occurring in the monastery – in other words, probably a time of increased polarization), and dropped dramatically at time four (before anyone was kicked out, but when the group was probably the most polarized). At time five the level of centralization had increased, which may reflect the fact that after the mass exodus, only like-minded people remained.

The networks can be visualized over time as illustrated by Figure 10.2. What these visualizations help us see is that it is not until the fourth time period that the network really begins to become divided. This, of course, is captured by the centralization measures in Table 10.1, but visually it is more dramatic. As one might guess, Pajek includes some very nice features for visualizing networks over time. After visualizing the network in the draw screen using the *Draw>Draw-Partition* command, you can switch from one time period to another with the draw screen commands *Previous* and *Next*, as long as the *Network* option is selected in the *Options>Previous/Next>Apply to* submenu. If you are visualizing a partition and/or a vector along with the network, then you can also select the *Partition* and/or *Vector* options in the *Options>Previous/Next>Apply to* submenu. Finally, if you select one of the energy options in the *Options>Previous/Next>Optimize Layouts* submenu, Pajek automatically energizes the network when you move to the next or previous network.

Draw>Draw

[Pajek-Draw Screen] Previous, Next

Options>Previous/ Next>Apply to>Network

Options> Previous/Next >Optimize Layouts

Longitudinal Networks in Pajek (Noordin Operational Network)

Now let us turn to Noordin's operational network (`Noordin Operational Network (Longitudinal).paj`) in which time codes have been entered that reflect when members entered and/or left. Of course, when someone enters and leaves a network is a matter of interpretation;

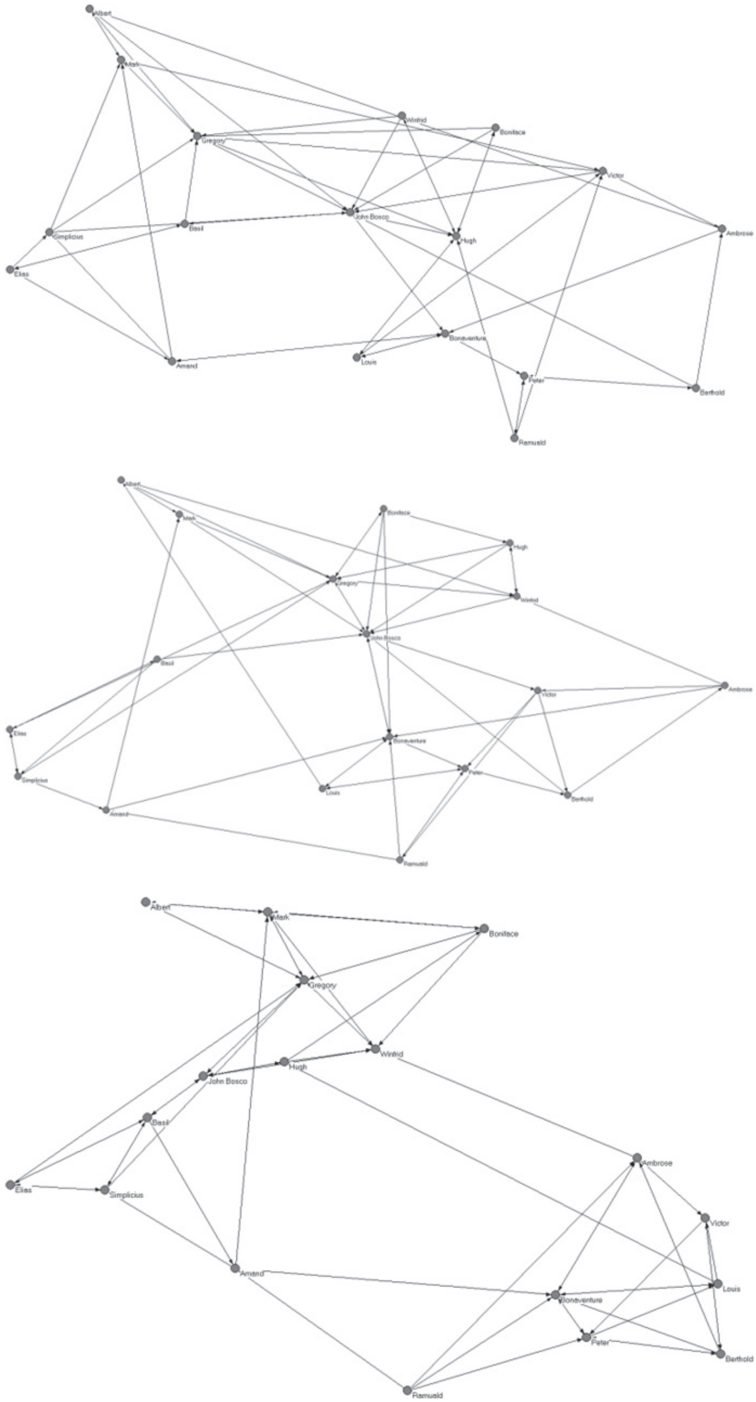
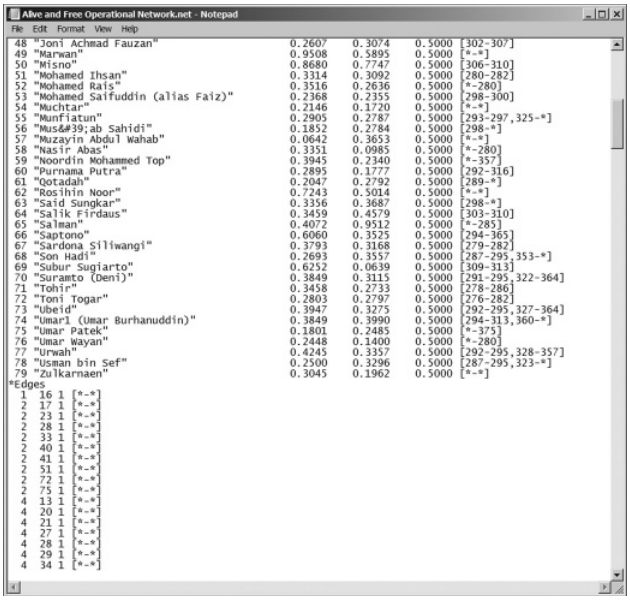


Figure 10.2. Sampson Liking Network at Times 2, 3, and 4 (Pajek)



```
Alive and Free Operational Network.net - Notepad
File  Edit  Format  View  Help
 48 "Joni Achmad Fauzan"                 0.2607   0.3074   0.5000 [302-307]
 49 "Marwan"                             0.9508   0.5895   0.5000 [*-*]
 50 "Misno"                              0.8680   0.7747   0.5000 [306-310]
 51 "Mohamed Ihsan"                      0.3314   0.3092   0.5000 [280-282]
 52 "Mohamed Rais"                       0.3516   0.2636   0.5000 [*-280]
 53 "Mohamed Saifuddin (alias Faiz)"     0.2368   0.2355   0.5000 [298-300]
 54 "Muchtar"                            0.2146   0.1720   0.5000 [*-*]
 55 "Munfiatun"                          0.2905   0.2787   0.5000 [293-297,325-*]
 56 "Mus&#39;ab Sahidi"                  0.1852   0.2784   0.5000 [298-*]
 57 "Muzayin Abdul Wahab"                0.0642   0.3653   0.5000 [*-*]
 58 "Nasir Abas"                         0.3351   0.0985   0.5000 [*-280]
 59 "Noordin Mohammed Top"               0.3945   0.2340   0.5000 [*-357]
 60 "Purnama Putra"                      0.2895   0.1777   0.5000 [292-316]
 61 "Qotadah"                            0.2047   0.2792   0.5000 [289-*]
 62 "Rosihin Noor"                       0.7243   0.5014   0.5000 [*-*]
 63 "Said Sungkar"                       0.3356   0.3687   0.5000 [298-*]
 64 "Salik Firdaus"                      0.3459   0.4579   0.5000 [303-310]
 65 "Salman"                             0.4072   0.9512   0.5000 [*-285]
 66 "Saptono"                            0.6060   0.3525   0.5000 [294-365]
 67 "Sardona Siliwangi"                  0.3793   0.3168   0.5000 [279-282]
 68 "Son Hadi"                           0.2693   0.3557   0.5000 [287-295,353-*]
 69 "Subur Sugiarto"                     0.6252   0.0639   0.5000 [309-313]
 70 "Suramto (Deni)"                     0.3849   0.3115   0.5000 [291-295,322-364]
 71 "Tohir"                              0.3458   0.2733   0.5000 [278-286]
 72 "Toni Togar"                         0.2803   0.2797   0.5000 [276-282]
 73 "Ubeid"                              0.3947   0.3275   0.5000 [292-295,327-364]
 74 "Umar1 (Umar Burhanuddin)"           0.3849   0.3990   0.5000 [294-313,360-*]
 75 "Umar Patek"                         0.1801   0.2485   0.5000 [*-375]
 76 "Umar Wayan"                         0.2448   0.1400   0.5000 [*-280]
 77 "Urwah"                              0.4245   0.3357   0.5000 [292-295,328-357]
 78 "Usman bin Sef"                      0.2500   0.3296   0.5000 [287-295,323-*]
 79 "Zulkarnaen"                         0.3045   0.1962   0.5000 [*-*]
*Edges
  1  16 1 [*-*]
  2  17 1 [*-*]
  2  23 1 [*-*]
  2  28 1 [*-*]
  2  33 1 [*-*]
  2  40 1 [*-*]
  2  41 1 [*-*]
  2  51 1 [*-*]
  2  72 1 [*-*]
  2  75 1 [*-*]
  4  13 1 [*-*]
  4  20 1 [*-*]
  4  21 1 [*-*]
  4  27 1 [*-*]
  4  28 1 [*-*]
  4  29 1 [*-*]
  4  34 1 [*-*]
```

Figure 10.3. Partial Listing of Noordin's Alive and Free Operational Network

consequently, two instances have been created: one where members are considered to have left only if they die or defect (i.e., an alive network) and one where they are considered to have left if they die, are arrested, or defect (i.e., an alive and free network). In both it is assumed that individuals were members of the network unless sources specifically stated that they did not come in contact with the group until a certain time (e.g., when an individual was recruited to be a suicide bomber or provide sanctuary to Noordin).

Figure 10.3 shows a partial listing of the alive and free operational network. It shows the lower half of the list of actors (i.e., vertices) and the upper half of the associated edge list. The actors' time is coded in terms of months with January 1980 equaling time one, but it could have just as easily been coded in terms of years (or some other time period). As you can see, some of the actors are considered to be in the network at all times (i.e., [*-*]). This, of course, is not possible, but without evidence of when they entered or left, they are treated as if they have been members forever. In practical terms, this means that they are treated as members of the network for the duration of whatever period of time is under investigation. Other actors (e.g., Urwah) entered and left the network more than once and are time coded as such (i.e., [292–295, 328–357]). Note also that unlike the time codes assigned to each edge (i.e., tie) in the Sampson data, most are coded as if they have existed forever (i.e., [*-*]). Again, although this is an unreasonable assumption, the practical effect is that the ties are treated as forming once both actors are present in the network and dissolving if either one leaves the network. If you scroll through the network file (e.g., in a text editor), you will note that some of the edges do have specific time records associated with them. Time stamps were only assigned to particular ties if such information could be gleaned from the sources.

File>Pajek Project File>Read

We will generate the new networks in exactly the same way as we did with the Sampson data. First, load the time-coded Noordin operational network (`Noordin Operational Network (Longitudinal).paj`) into Pajek. After ensuring that the alive and free operational network is showing in the first Network drop-down menu, use Pajek's *Net>Transform>Generate in Time>All* command. This time, however, choose 253 (i.e., January 2001) as the first time point and 361 (January 2010) as the last time point with a step of twelve (12). This will generate ten networks, ranging from January 2001 to January 2010. We can visualize these as we did before in Pajek's Draw screen.

Net>Transform >Generate in Time>All

Options> Previous/ Next>Apply to>Network

We can also examine how the network's topography changes over the ten-year period. To do this, we highlight each network separately in the first Network drop-down menu, and then issue the *Info>Network>General* command, which produces a report on the density and average degree for each network. Of course, we may not be interested in average degree and density; we may be interested in centralization, clustering, and so on. Each of these measures can be estimated for each network and compared, as well. Although doing this in Pajek is possible, it is much easier in ORA. Before moving to ORA, you will want to save each of the ten newly generated network files (making sure that each is separately showing in the first Network drop-down menu), using Pajek's *File>Network>Save* command.

Info>Network >General

File>Network >Save

Longitudinal Networks in ORA (Noordin Operational Network)

Begin by loading the ten time-coded trust network files into ORA, using its *File>Data Import Wizard* command, which brings up the following dialog box (Figure 10.4). Click on "Next" and at the next dialog box, browse for the newly created Pajek network files. When you import a Pajek file containing time records into ORA, ORA attempts to account for the time records and typically creates numerous networks, one for each year represented by the time records. Thus, for January 2003 data, you will want to delete all of the networks except the one ending "–277," and then repeat this for each of the nine remaining years. If ORA consistently read the Pajek data and corresponding time records correctly, then we could simply import the original time-coded data and keep the years that

File>Data Import Wizard

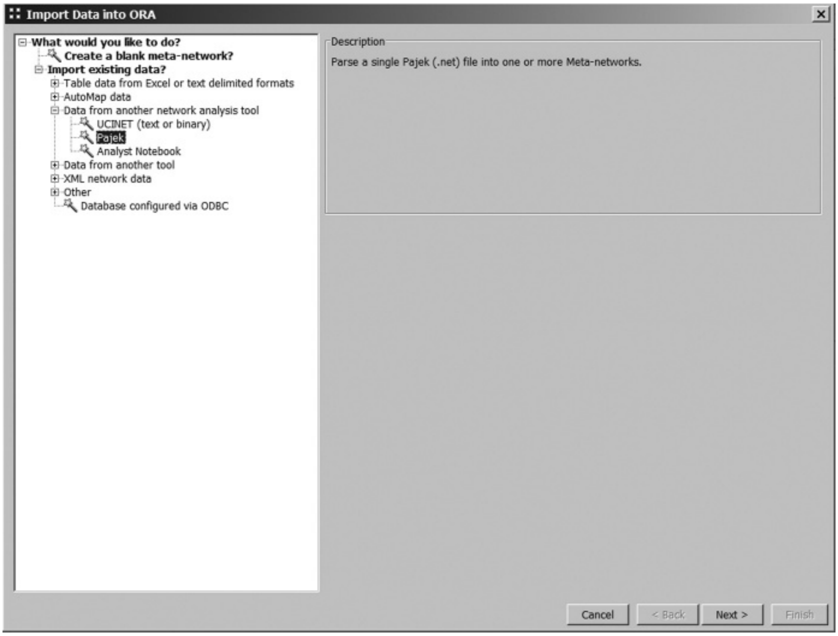


Figure 10.4. ORA's Data Import Wizard (Pajek Option Selected)

we wanted to analyze. Unfortunately, at this writing, the networks do not always "show up" in ORA with the correct number of nodes (although I expect this will eventually be corrected).

An alternative intermediary step is to first import the Pajek networks into UCINET, using UCINET's *Data>Import text file>Pajek* command, which calls up the following dialog box (Figure 10.5). Clicking "OK" creates two sets of files: (1) a UCINET network file and (2) a UCINET coordinate file. The latter file can be used for visualizing networks in NetDraw, but it is the former that interests us here. If we first import all of the Pajek networks into UCINET and then import the resulting UCINET networks into ORA using the same aforementioned commands (except selecting the UCINET option instead of the Pajek option), we will not encounter any time code issues.

UCINET's Data>Import text file>Pajek

ORA File>Data Import Wizard

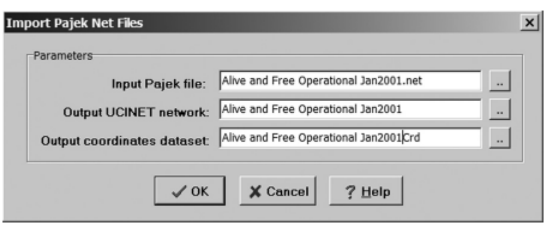


Figure 10.5. UCINET Pajek Import Dialog Box

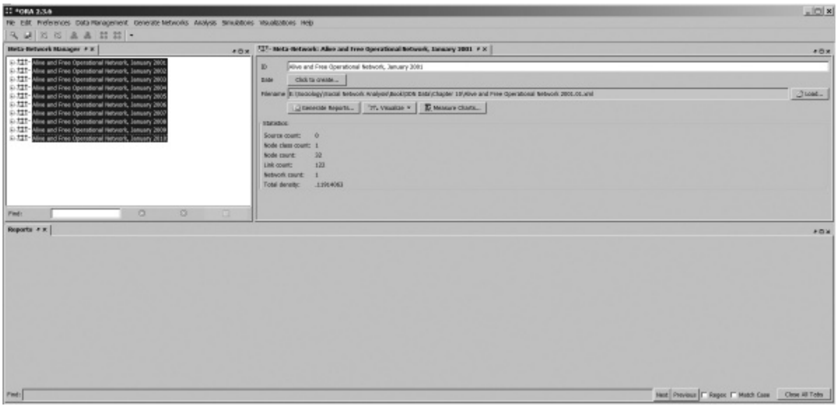


Figure 10.6. ORA's Main Screen with Networks Highlighted

Visualizations >View Networks Over Time

Once the import process is complete, you should have the ten networks loaded into ORA as separate meta-networks. You can obtain topographical metrics for each network following the procedures outlined in Chapter 5, and we can visualize the networks over time by highlighting all ten networks (Figure 10.6) and then selecting the *Visualizations>View Networks Over Time* command. This will produce a dialog box (not shown) that indicates that the networks lack an assigned date. This is not an issue for us here. Click "OK," and ORA will call up its *Visualizer* screen along with a "Networks Over Time" dialog/control box (not shown) that offers options for visualizing the networks as well as "playing" them. If you click "Play," ORA will walk through each of the ten networks (not shown).

Visualizations >View Measures Over Time

Although ORA's capabilities for visualizing longitudinal networks are quite good, here we are going to focus on ORA's capabilities of examining network measures over time. To do this, highlight all ten networks (Figure 10.6) and then select the *Visualizations>View Measures Over Time* command. This will generate a new dialog box (not shown) that asks you what measures you want to view. For now, select ORA's default "All Measures" and click "Compute." This will bring up ORA's "Measures Over Time" function (Figure 10.7), which allows you to select various measures and see (graphically) how they change over time.

Here, fragmentation (top line) and degree centralization (bottom line) have been selected, but as you can see, a number of measures are available for longitudinal analysis. Interestingly, the network became noticeably less fragmented (i.e., more cohesive) between January 2004 and January 2005, which reflects the period of time after the Australian Embassy bombing (July 2004) and before the second Bali bombings (October 2005). After this the network appears to become slowly but increasingly more fragmented, although it may have begun to stabilize in 2007.

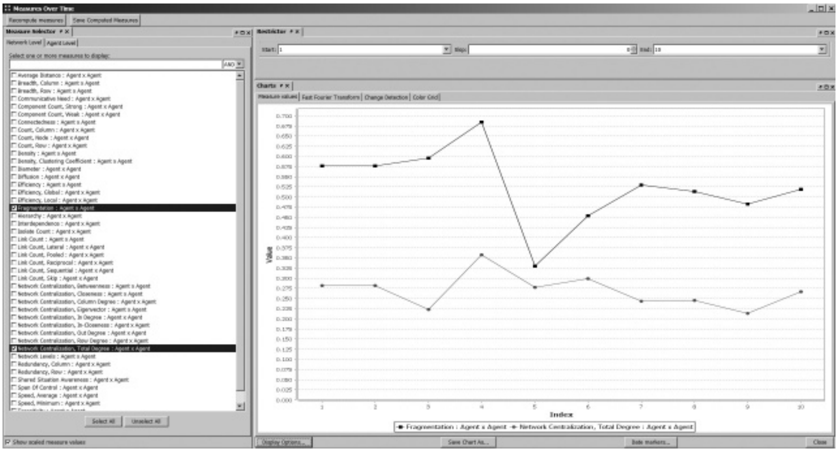


Figure 10.7. Measures over Time of Noordin's Alive and Free Operational Network

In terms of centralization, the network became more centralized between 2003 and 2004 and then slowly returned to its earlier levels. In Chapter 5 we saw that, overall (i.e., not only the alive and free subset), Noordin's operational network was relatively centralized, which suggests that we may want to examine other aspects of the operational network, such as the alive subset. We will do this in conjunction with exploring ORA's change detection functionality.

Before turning to that, however, it is worth noting that if you click on the Source Node Level tab, you will discover that you can see how the position of the actors in the network changes over time as well (e.g., how their degree centrality varies over the period of time under examination). Also, you can save the analysis in various ways by clicking on "Save Chart As . . . " found below the chart. If you choose to save it as a *.png or *.jpeg, ORA saves the graph itself; if you choose to save the chart as a *.csv, ORA saves the computed metrics in a format that can be imported into and analyzed in other statistical programs such as Excel, SPSS, Stata, and R.

Social Network Change Detection in ORA (Noordin Operational Network)

Social network change detection (SNCD) is a method that allows for real-time analysis and can alert analysts as to whether and when a sudden change occurs in a network. This, in turn, can help them to identify potential causes for the change and may help prevent unwanted future events (McCulloh and Carley 2011). For example, "terrorist organizations will begin planning their attacks, long before they are actually carried out.

Rapid change detection could alert military intelligence analysts to the shift in planning activities prior to the attack occurring" (McCulloh and Carley 2011:5).

File>Open Meta-Network

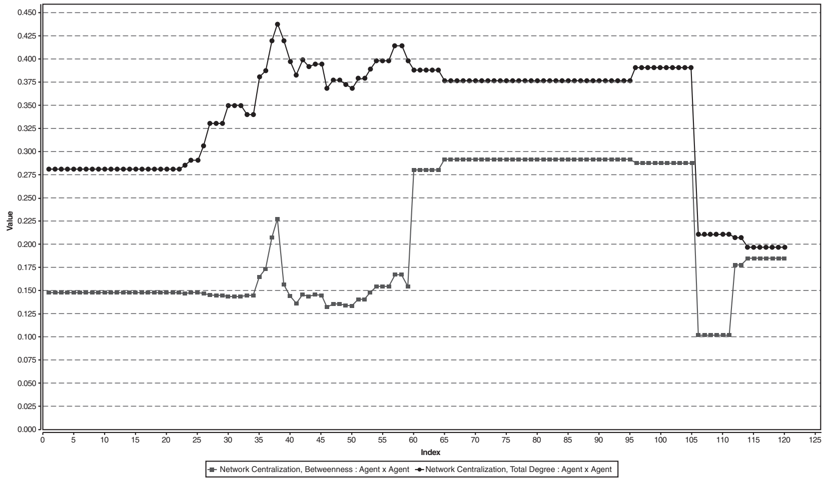
Visualizations> View Measures Over Time

To see how ORA implements SNCD, first read all alive operational networks (these are monthly, not yearly, networks) into ORA using its *File>Open Meta-Network* command.[3] Highlight all 120 networks and select the *Visualizations>View Measures Over Time* command. In the resulting dialog box (not shown), accept ORA's defaults, click "Compute," and ORA's "Measures Over Time" function should be ready for analysis. Building upon and following our previous logic, we analyze both centralization (degree and betweenness) and fragmentation (fragmentation and clustering coefficient) – first descriptively, and then turning to SNCD. In the upper panel of Figure 10.8 the black and gray lines map monthly degree and betweenness centralization, respectively, from 2001 to 2010; in the lower panel, they map monthly fragmentation and clustering coefficient over the same period of time. These graphs indicate that Noordin's alive operational network began to become more centralized (and less fragmented) in 2003 (beginning with time period 24), a process that continued until 2006 (beginning with time period 60). Then, in 2009 when Noordin and a few of the network's key members were killed, the network became significantly less centralized and more fragmented. It was at this time that the network essentially fell apart although remnants did try later to reconstitute the network (International Crisis Group 2010). This is consistent with Bakker, Raab, and Milward's (2011) contention that centralized dark networks tend to be less resilient than decentralized ones, and that they are more likely to collapse if they suffer a shock to the system, much like Noordin's network did. That the network collapsed after the removal of key operatives also appears to lend support for kinetic strategies that remove key members. While this may be true, it is worth noting that initially, Noordin's network was less centralized and may have been able to withstand the removal of key actors. Indeed, it is arguable that the network became more centralized over time because of the kinetic and nonkinetic strategies pursued by Indonesian authorities, suggesting that these earlier strategies are what made Noordin's removal at a later date effective.

To see what SNCD can contribute to this analysis, select the Change Detection tab in the "Measures Over Time" screen, rather than accessing it from ORA's main screen.[4] To simplify our analysis, we will illustrate SNCD using only degree centralization. As with many statistical

[3] The January 2001 network is named `Alive Operational Network 2001.01.xml`, the February 2001 is named Alive Operational Network 2001.02.xml, and so on through December 2010 (Alive Operational Network 2010.12.xml).

[4] While SNCD is available from the latter, it is more limited than it is through the "Measures Over Time" screen.



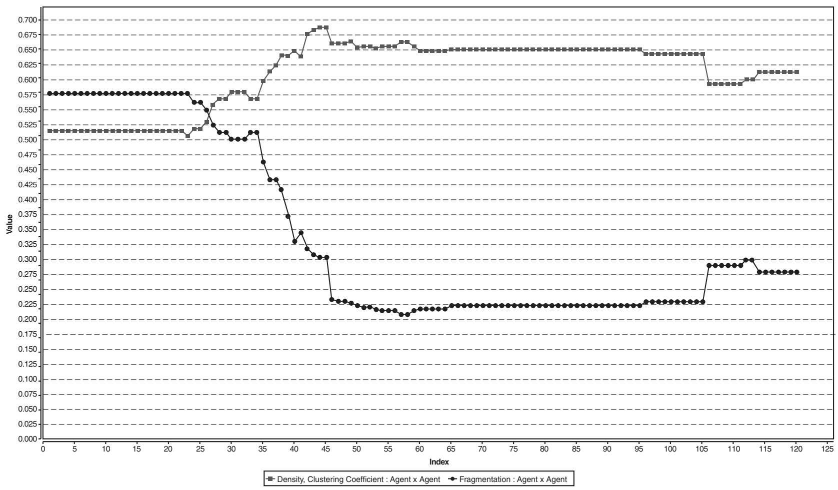


Figure 10.8. Measures over Time of Noordin's Alive Operational Network

approaches, we have to choose between numerous options. We first need to select the process by which to monitor change. One option, the Shewhart x-bar chart method, was originally a quality control technique for monitoring change in a business or industrial process when samples were collected at regular intervals (Shewhart 1927). The cumulative sum (CUSUM) control chart (Page 1961) is generally seen as an improvement over the Shewhart x-bar because of its use of sequential probability ratio testing (McCulloh and Carley 2011:7). The final option, the exponentially weighted moving average (EWMA) control chart (Roberts 1959), has

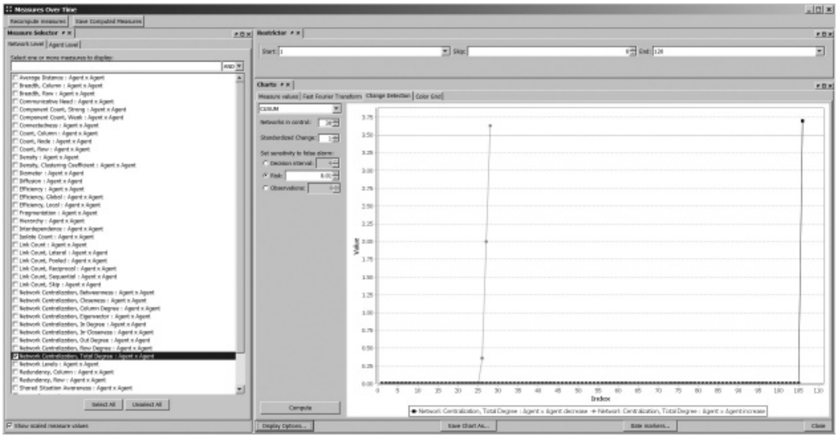


Figure 10.9. Change Detection Noordin's Alive Operational Network

been shown to perform similarly to CUSUM. Here, we follow McCulloh and Carley (2011:7) and use CUSUM, which they recommend for SNCD with longitudinal data. Next, we need to select the number of networks needed in order to form a baseline to which to compare changes. We have selected thirty networks (i.e., a quarter of the total networks), but there is no hard and fast rule. Finally, we have accepted ORA's default risk/decision threshold for a false alarm (i.e., 0.01/3.50) although McCulloh and Carley (2011:15) used 0.017/3.00 in their analysis. This threshold is denoted by the thin, solid line running horizontal across the graph at the 3.50 mark in Figure 10.9. If a change in the network crosses this line, then a significant change is considered to have occurred. Keep in mind that if the threshold is set too high, then analysts may miss an important change in a network under examination. If it is set too low, then analysts may mistake normal change for sudden change and potentially take unnecessary actions. Thus, analysts need to be careful in determining what constitutes a significant change, drawing heavily on experience and being willing to adjust expectations in light of new information.

Looking at Figure 10.9, two significant changes appear to have occurred. One, denoted by the gray (or lighter) line, indicates whether a significant increase has occurred in terms of network centralization; the other, denoted by the black (or darker) line, indicates whether a significant decrease in network centralization occurred. Both cross the decision threshold (i.e., 3.50), indicating that both a significant increase and decrease in degree centralization occurred in the network. When did these occur? Most likely at the point that the lines leave the baseline and begin their climb upward. That the black line begins its climb on September 2009 (the 105th month) is relatively uninteresting because that is when Noordin was killed. However, the fact that the gray line leaves the baseline

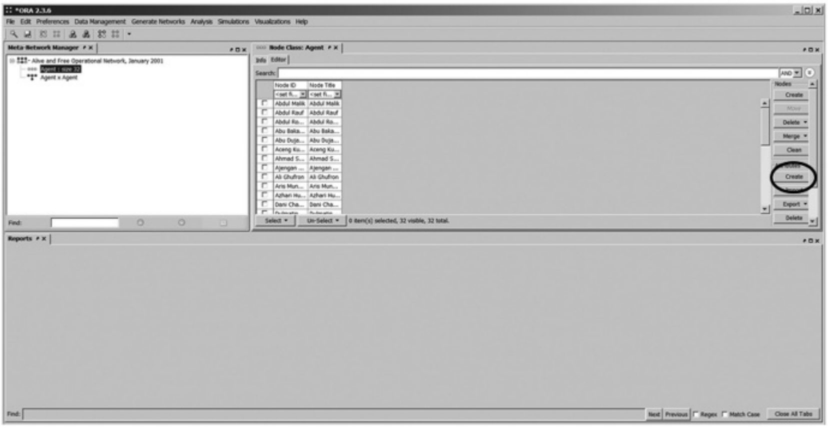


Figure 10.10. Node Class Editor

at February 2003 (the 25th month) is interesting because it occurs at about the time Noordin began to distance himself from Jemaah Islamiyah (JI). Indeed, it occurs only two months after Noordin acquired the explosives left over from the Christmas Eve bombings carried out in 2000 by JI, which implies that the acquisition of the explosives may have been the catalyst for Noordin to strike out on his own (International Crisis Group 2006:3). Thus, the increase in centralization may not have been a response to exogenous pressures, such as the efforts of Indonesian authorities to shut down Noordin's network, but rather to something inherent in the network itself, such as Noordin's leadership style. Of course, it could be the result of both factors.

10.3 Fusing Geospatial and Social Network Data

We now turn to an examination of how ORA fuses geospatial and relational data. We begin by walking through the steps of preparing social network data for fusion with geospatial data, in particular how to enter geospatial coordinates into ORA. Next, we examine how to fuse social network data with corresponding geospatial data in terms of both metrics and visualization.

Preparing Social Network and Geospatial Data in ORA

First begin by loading the January 2001 alive and free operational meta-network (`Alive and Free Operational Network January 2001.xml`) in ORA using ORA's *File>Open Meta-Network* command. Highlight the agent node class of the meta-network and then click on *[ORA-Main Screen] File>Open Meta-Network*

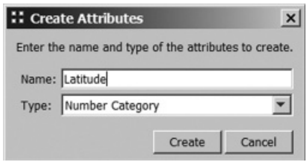


Figure 10.11. Create Attributes Dialog Box

the Editor tab in the Information/Editor panel of ORA's main interface (Figure 10.10).

In ORA we enter geospatial data as actor attributes. ORA allows users to enter geospatial coordinates in a number of different formats: (1) Latitude and Longitude,[5] (2) Military Grid Reference System (MGRS),[6] (3) Universal Transverse Mercator (UTM),[7] and (4) Cartesian.[8] To enter attribute data directly into ORA, click "Create," which is on the right side of the Information/Editor panel (circled in Figure 10.10); this calls up a dialog box (Figure 10.11) where we enter the name of the attribute (e.g., Latitude) as well as attribute type (note that "Number Category" has been selected as the attribute type because geospatial coordinates are nominal, rather than ordered, data).

If we repeat this process for creating longitudinal and MGRS attributes, we should end up with three empty columns in which we can enter geospatial data either by hand or by copying and pasting from a spreadsheet. Included with the data for this chapter is a *.csv file (`LatLonMGRS2001.csv`) that contains latitude, longitude, and MGRS data that can be copied and pasted into the columns we just created (see Figure 10.12).[9]

Visualizing Social Network and Geospatial Data in ORA

Visualizations> Geospatial Networks

One function implemented in ORA that fuses geospatial and social network data is its ability to visualize social network data geospatially. To do this we use ORA's *Visualizations>Geospatial Networks* command, which calls up a dialog box (not shown), indicating that ORA has not detected any location information. Click "OK." This will bring up a dialog box similar to Figure 10.13, which we use to tell ORA where it can find the geospatial information it needs. First, highlight the agent node class in the left-hand portion of the dialog box. Then, click "Add GIS

[5] See http://en.wikipedia.org/wiki/Lat_lon.
[6] See http://en.wikipedia.org/wiki/Military_grid_reference_system.
[7] See http://en.wikipedia.org/wiki/Universal_Transverse_Mercator.
[8] See http://en.wikipedia.org/wiki/Cartesian_coordinate_system.
[9] Although the data are for locations where Noordin network was operative (e.g., Indonesia, Philippines), any correspondence between these and actual locations of members of Noordin's network at a particular time is purely coincidental.

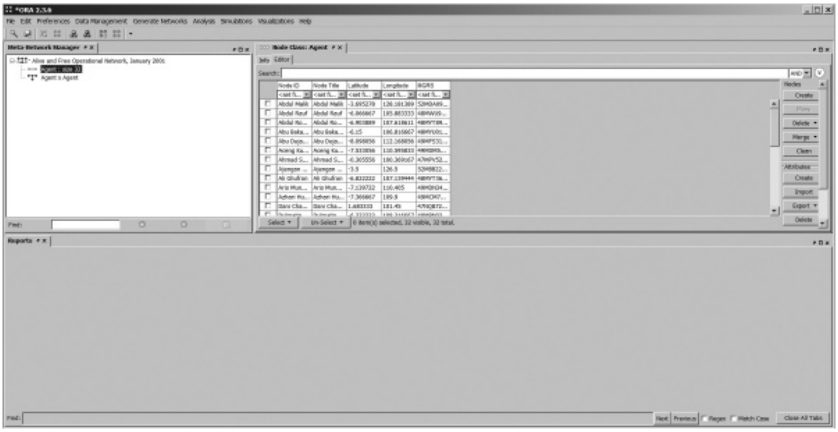


Figure 10.12. Node Class Editor with Latitude, Longitude, and MGRS Attributes

Attribute" toward the bottom of the dialog box. This will bring up a subdialog box (also shown in Figure 10.13), which asks you to indicate the type of GIS coordinate system you intend to use; we have chosen MGRS, but we could have just as easily chosen to use latitude and longitude coordinates. Next, select the node attribute where the MGRS coordinate is found, click "Finish," and the subdialog box will disappear. Click "Next" at the bottom of the dialog box, and (not surprisingly) another dialog box will appear where you will want to click "Finish" one more time.

When you do this, ORA's GIS visualizer will appear with the 2001 alive and free operational network mapped geospatially (probably in the lower left of the map). In Figure 10.14 we centered the map by clicking on

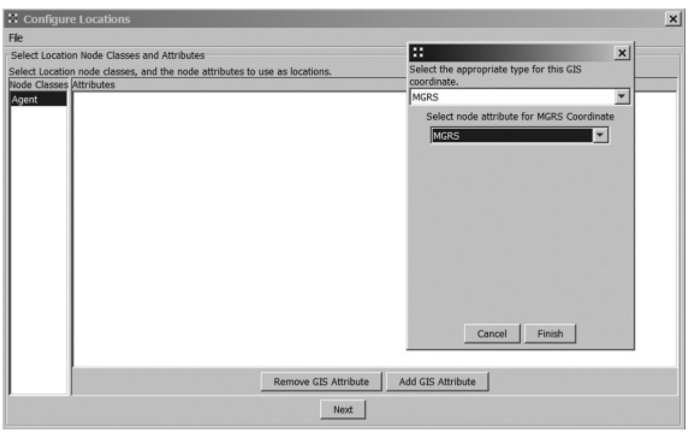


Figure 10.13. Configure Locations Dialog Box

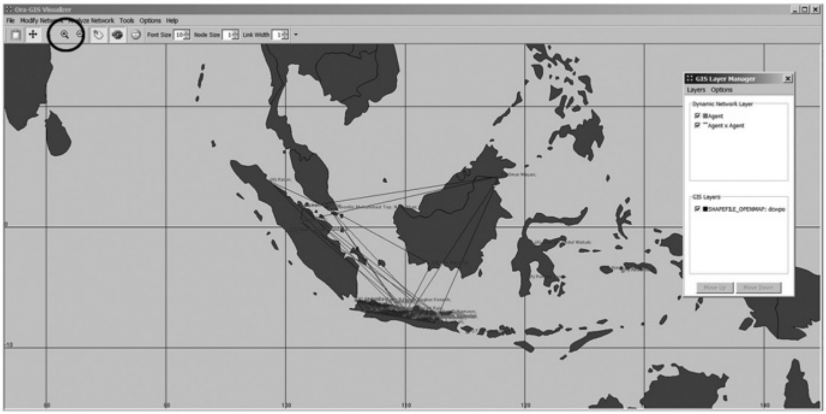


Figure 10.14. ORA's GIS Visualization of Noordin Alive Operational Network

the portion of the map we wanted to center and then zoomed in by first clicking on the "zoom" tool (circled in Figure 10.14) and then clicking on the map.

File>Save Map>Save Map to KML, Save Map to SHP

Although ORA's mapping capabilities are somewhat limited (although you may want to explore ORA's three-dimensional option using the *Options>Use 3D Visualization (NASA WorldWind)* command found in ORA's GIS Visualizer menu), it does allow you to export the network in Google Earth and ArcGIS formats with its *File>Save Map>Save Map to KML* and *File>Save Map>Save Map to SHP* commands, respectively. Picking up on the theme of the previous section, analysts could geospatially map dark networks over time in order to see whether (and how) they vary and what such variances may indicate about the network. An example of this is presented in Figure 10.15, which pictures Noordin's alive and free operational network in 2001, 2004, 2007, and 2010 (the panels run left to right, top to bottom). They indicate that the network was anything but static, and was instead constantly on the move.[10] For example, in 2001 (upper-left panel) the members of Noordin's network were located solely in Indonesia and Malaysia, but by 2004 (upper-right panel) a few members had relocated to the Philippines. In 2007 (lower-left panel) there appears to have been a movement of some members to the island of Java, but in 2010 (lower-right panel), shortly after Noordin's death, the network appears to have dispersed across Indonesia.

[10] A series of networks from 2001–2010 have been created that include geospatial information (e.g., `Alive and Free Operational Network January 2001 (geo).xml`). As previously noted, any correspondence between these and actual locations of members of Noordin's network at a particular time is purely coincidental; thus, the following analysis is purely hypothetical.

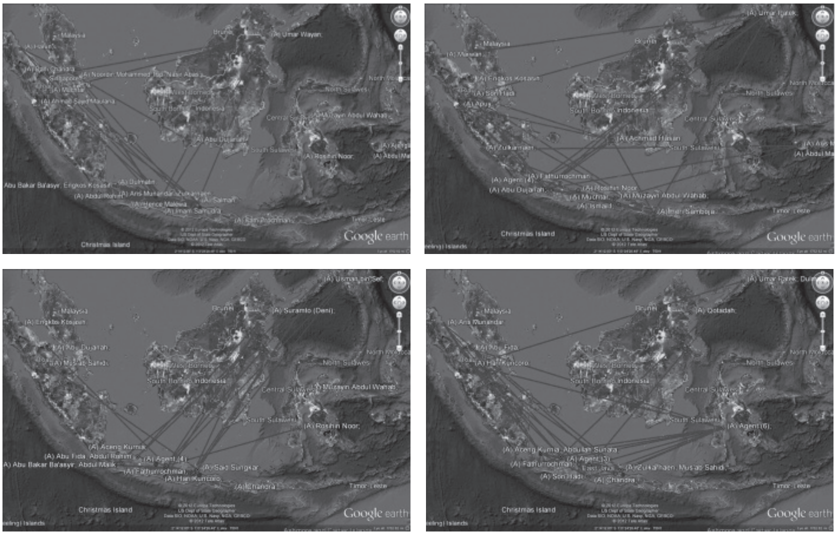


Figure 10.15. Google Earth Maps of Alive and Free Operational Network

The geospatial mapping of social networks suggests something else, as well. In particular, it seems likely that actors who are peripheral to a network in terms of standard measures of centrality may be geospatially located in such a way that they are more central to the network's operations than otherwise believed. Thus, it makes sense to estimate social network metrics that take into account the geospatial location of actors, although given the ease by which persons and resources move through the modern world, they should not be seen as a substitute for, but rather as a compliment to, standard social network metrics. The availability of such metrics is currently limited, but ORA does estimate a few, and it is to that functionality that we now turn.

Geospatially Weighted Social Network Metrics in ORA

When exploring ORA geospatial visualization capabilities, we encountered a dialog box that indicated that it could not detect any location information. That is because we entered the geospatial data as attributes of the actors in the network rather than create a separate location node class. This is fine if all we want to do is visualize social networks geospatially and nothing else. However, if we want to take advantage of ORA's ability to estimate geospatially weighted centrality metrics (Olson and Carley 2009), then we need to create a separate location node class. To do this, right-click on the alive and free operational meta-network

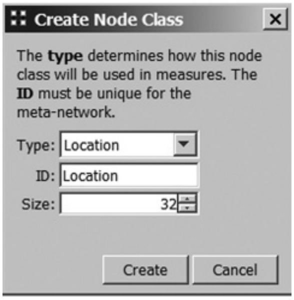


Figure 10.16. ORA's Create Node Class Dialog Box

(`Alive and Free Operational Network January 2001.xml`); this brings up a dialog "bubble" box (not shown) that provides users with several options. Choose the "Add New Node Class" option. This will bring up ORA "Create Node Class" dialog box (Figure 10.16). Using the drop-down menu, select *Location*, and indicate that it should have the same number of nodes (i.e., size) as the number of agents in the network (in this case, thirty-two, but be aware that the size will vary from network to network) because we will assign each actor to a unique location. When done, click "Create."

Next, we need to add an agent-by-location network. Because each agent (actor) will be assigned its own unique location, we need to create a square network (i.e., 32 × 32). To do this, once again right-click on the meta-network, which will call up the dialog bubble box we previously saw. This time, choose the "Add Blank Network" option, which will bring up ORA's "Create Network" dialog box (Figure 10.17). For the source node class, choose *Agent*, for the Target node class, choose *Location* and click "Create."

The next to last step involves assigning actor one to location one, actor two to location two, and so on. We do this by right-clicking on the "Agent by Location" network we just created, which brings up its own dialog "bubble" box (not shown). From the list of options, select "Set

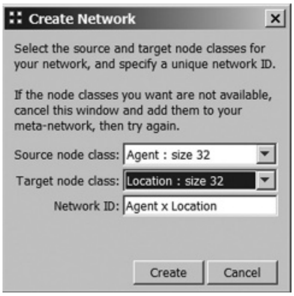


Figure 10.17. ORA's Create Network Dialog Box

Diagonal." The default sets the diagonal to one – that is, True (+1), the option we want – so click "OK." To examine the result, with the agent-by-location network still highlighted in the Meta-Network Manager panel, select the Editor tab found under the Information/Editor panel, which allows you to see the matrix underlying the agent-by-location network (not shown).[11] It is by setting the diagonal to one that we assign each actor to a unique location. The final step involves entering the latitude and longitude coordinates for each actor/location as we did before.[12] Now we are ready to compare standard measures of actor centrality with geospatially weighted ones.

Let's begin by first calculating standard social network analysis metrics for the 2001 alive operational network using ORA's *Analysis>Generate Reports>Locate Key Entities>Standard Network Analysis* command (or use the Generate Reports speed button located on the Editor portion of ORA's main interface). At the first dialog box (not shown) select the network, at the second dialog box (not shown) select only the agent node class for analysis because in this case we are not interested in location metrics, at the third dialog box (not shown) indicate the number of ranked nodes to display (the default is ten), and at the final dialog box (not shown) select the desired output options (e.g., text, HTML, CSV) and click "Finish." As we saw in Chapter 7, this command generates a series of commonly used social network metrics for the trust network.

Analysis> Generate Reports>Locate Key Entities>Standard Network Analysis

Next, analyze the network using ORA's *Analysis>Generate Reports> Geospatial>Geospatial Assessment* command. At the first dialog box (not shown) select the network, at the second dialog box (not shown) accept ORA's defaults because ORA needs the location information in order to estimate the geospatially weighted metrics, at the third dialog box (not shown) indicate that the location node class is the correct one (this option is here just in case there is more than one location node class), and at the final dialog box (not shown) select the desired output options and click "Finish." This report generates a series of distance statistics (e.g., agents closest together, agents farthest apart) as well as betweenness, closeness, and degree centrality metrics weighted by their spatial location. Because we are working with a disconnected graph, we will ignore the closeness scores and focus on the degree and betweenness scores (Table 10.2).

Analysis> Generate Reports> Geospatial> Geospatial Assessment

At first glance the standard metrics output may strike readers as strange because Noordin is not among the top-ranked actors in terms of degree centrality, and he ranks seventh in terms of betweenness centrality. Keep

[11] ORA provides two ways of looking at an underlying matrix: Either in "binary view" where relations between objects are simply check marks, or in "numeric view" where you can see the actual cell values.

[12] It is unnecessary to repeat those steps here. However, the following analysis requires latitude and longitude data, rather than MGRS data, so be sure to use the former rather than the latter.

Table 10.2. *Comparison of standard and geospatially weighted centrality metrics*

Standard		Geospatially weighted	
Degree	Betweenness	Degree	Betweenness
Azhari Husin (0.387)	Salman (0.161)	Umar Wayan (0.085)	Salman (0.311)
Hambali (0.323)	Azhari Husin (0.138)	Muchtar (0.083)	Azhari Husin (0.238)
Dulmatin (0.290)	Dulmatin (0.090)	Dani Chandra (0.078)	Dulmatin (0.160)
Umar Patek (0.290)	Umar Patek (0.090)	Hambali (0.074)	Umar Patek (0.160)
Imam Samudra (0.258)	Harun (0.041)	Azhari Husin (0.069)	Hambali (0.058)
Salman (0.258)	Hambali (0.039)	Noordin Top (0.067)	Harun (0.048)
Abdul Rauf (0.226)	Imam Samudra (0.009)	Salman (0.066)	Noordin Top (0.014)
Ali Ghufron (0.226)	Noordin Top (0.009)	Imam Samudra (0.049)	Imam Samudra (0.010)
Iqbal (0.226)		Aris Munandar (0.049)	
6 actors w/same score (0.161)		Zulkarnaen (0.049)	

in mind, however, that this is the network in 2001 when Noordin was still a full and faithful member of Jemaah Islamiyah, so it is not surprising that he does not rank too high. Still, the fact that he was one of eight individuals ranked in terms of betweenness centrality (all the rest had betweenness scores of 0.00) is evidence that Noordin was in a position to form a dark network of his own. Now compare the standard scores with the geospatially weighted ones. Although there is very little difference between the two sets of betweenness scores, the degree centrality scores do differ substantially from one another. Individuals who are not ranked in terms of standard measures of degree centrality are ranked in terms of the geospatially weighted measures (e.g., Umar Wayan, Muchtar, Dani Chandra, and Noordin Top).

All this suggests that analysts will want to want to take such metrics into account (when possible, of course) when crafting strategies. For example, actors who rank lower in terms of geospatially weighted centrality metrics than in terms of standard centrality metrics may indicate that they play roles that, although important, are more peripheral to the network's day-to-day operations (e.g., operating training camps but not participating in actual operations). Moreover, key members who are geospatially removed from the network may be easier to isolate than those who are not (e.g., by disrupting transportation routes or communication links). Of course, we can also estimate these metrics over time, which may help analysts monitor the network to identify emerging leaders and important clusters of actors. Put simply, a close examination of a dark network's geospatial patterns and variance over time can supplement standard forms of analysis. As we noted previously, these should not be seen as a substitute for standard social network analysis but rather as a complement to something that can inform analysts as they attempt to track and disrupt dark networks.

10.4 Summary and Conclusion

In this chapter we have examined two approaches for investigating the dynamic nature of dark networks: the analysis of how networks change and adapt over time and the fusion of geospatial and social network data. With regard to longitudinal networks, we explored both descriptive and statistical detection approaches for studying network change. As noted at this chapter's outset, these are not the only options available to analysts studying longitudinal networks. The actor-based models developed by Tom Snijders and his colleagues (Snijders 2001, 2005; Snijders et al. 2010) model variation in network structure over time, taking into account both processes that are endogenous (i.e., internal) to the network itself as well as factors that are exogenous (i.e., external) to the network, such as actors' attributes (Prell 2011:215). Although promising for the study of dark networks, they have yet to be implemented in UCINET, Pajek, or ORA and are still somewhat difficult for the average analyst to use. The fusion of geospatial and social network data represents an exciting leap forward in the analysis of social networks, allowing analysts to complement traditional social network analysis with geospatially informed ones. On the one hand, the ability to geospatially plot social network data could help analysts detect patterns (both cross-sectionally and longitudinally) that they might have otherwise missed. On the other hand, the availability of geospatially weighted metrics may help in the detection of central and peripheral actors; to be sure, only a few geospatially weighted metrics are currently available, but there is reason to hope that more will be available in the near future.

How have the approaches explored in this chapter improved our understanding of Noordin's network? Our longitudinal analysis of the alive operational network found that it became increasingly more centralized and less fragmented from 2003 to 2006 and then reversed course when Noordin and a few of his close associates were killed in 2009, which corresponded with the essential collapse of the network. This finding is consistent with Bakker et al.'s (2011) that centralized dark networks are less resilient to shocks than are decentralized ones; the authors' work also lent support for the use of kinetic strategies to remove key network members. However, while this conclusion may be warranted, the network only became more centralized over time, and this may have been in response to various strategies used by the Indonesian forces. If this was in fact the case, then when exploring possible strategies disrupting dark networks, analysts may want to consider those that cause networks to become more centralized. For example, sowing distrust may lead a dark network's leaders to become concerned that the security of the network is at risk and cause them to consolidate (i.e., centralize) the network's operations into the hands of a few individuals.

Interestingly, the application of statistical network change detection techniques to the same network indicated that a significant increase in centralization began not long after Noordin acquired explosives left over from the Christmas Eve 2000 bombings, suggesting that the increase in centralization may not have been in response to exogenous factors, such as the efforts of Indonesian authorities, but rather in response to endogenous factors, such as Noordin's leadership style. Of course, the increased centralization of the network could be a result of both internal and external factors. Perhaps just as important, however, this finding highlights the important role that the tracking and seizure of explosives and other weapons may have in the prevention of terrorist attacks such as those carried out by Noordin and other dark networks.

Although the geospatial data used in the examples in this chapter were fictional, they nevertheless point to the potential role that the fusion of geospatial and social network data can play in the disruption of dark networks. For example, apart from the inclusion of geospatial data in the analysis, a network may appear to be relatively static, but once geospatial data are taken into account, analysts make discover that the network is anything but static and is instead constantly on the move. Moreover, the inclusion of geospatial network data highlighted why analysts, when possible, should take geospatially weighted metrics into account when crafting strategies. We saw, for example, how key members who are geospatially removed from the network may be easier to isolate than those who are not. Geospatially weighted metrics should not be seen as being a substitute for standard social network metrics but rather as a complement to them, something that can inform analysts as they attempt to track and disrupt dark networks.