**Connecting ORD *aka* Operation Bacon***By N. Pollesch and D. Hoff*

# Introduction

EPA’s Office of Research and Development is getting reorganized. This reorganization will see managers changing roles and new managers being hired. In December, ORD will have a management meeting. One goal of this meeting is to build connections among managers and strengthen the overall ORD network. Given that we are the Office of Research and Development, we want to see if we can make the most of this networking opportunity, using network analysis, mathematics, and Bacon (Kevin Bacon, that is).

The concept here is similar to the “Kevin Bacon number” idea. In a nutshell, this is how far each actor/actress is from Bacon. If you were in a film with Kevin, you are a distance of 1, if you were in a film with someone who was in a film with Kevin, 2, and so on. However, since there are no Kevin Bacons in ORD, instead of trying to make everyone close to a single person, we will try to make everyone close to *everyone*, i.e. minimize the average *distance* in the network. The concept of distance in a network varies based on the application; for this application distance is the shortest route through the network between two people. The goal of this exercise is to use the meeting, and the opportunity to create small-groups of managers at the meeting, to build connections in an informed way. We hope to lower the average distance between all managers at ORD, and thus bringing everyone at ORD closer together.

# Methods

## Creating the ORD network:

Using the images released about the upcoming reorganization (https://intranet.ord.epa.gov/about-ord/proposed-ord-organizational-structure), an edge list of connections was created in an excel spreadsheet (Supplement A). The edge list was imported into R and used the igraph package to create the network. Edges are not assumed to be directional and the current structure assumes that there is a single manager per node. These assumptions can be modified.

## Group Sizes Algorithm

In order to determine optimal grouping strategies, one must first decide how many groups to divide the network into and how many people are in each group. A group size algorithm was developed that, by default, seeks to balance the number of groups and the number of people in those groups. For example, if there were 100 people, the algorithm would divide them into 10 groups of 10. If there were 101, then 9 groups of 10 and 1 group of all. The algorithm uses the closest perfect square (e.g. 10 for 101) as the starting point for group sizes and creates groups of different sizes as needed by the number of people specified. It also defaults to having more people in a group than more groups. However, facilities may be constraining so that the default algorithm may determine more groups than there is breakout space. Thus, the algorithm has the option of allowing for group size to be specified, in which case it is balances the number of people among the number of groups specified. In either case there are at most 2 different groups sizes. For example, 103 people will be grouped into 7 groups of 10 and 3 groups of 11, as opposed to 8 groups of 10, 1 group of 11, and 1 group of 12. Also, group size differs by no more than one, so that 102 people will be in 8 groups of 10 and 2 groups of 11, as opposed to 9 groups of 10 and 1 groups of 12. The exact form of the algorithm is in the R Code provided as Supplement B.

## Grouping Algorithms

With a specified number of groups and number of individuals in those groups, as well as the underlying network structure, the effect of different grouping strategies can be explored. Three grouping strategies were explored; “*gregarious”* maximum distance grouping, “s*hy”* minimum distance grouping, and “*pass out playing cards or draw numbers from a hat”* random grouping. For each strategy an algorithm was developed and programmed in R (available in Supplement B).

In each of the three approaches, it is assumed that once a group is chosen, everyone in that group is as close to everyone else in the group as possible. Specifically, an edge is added between every individual in the new group, so that any pair of individuals in the group are at most a distance of 1 from each other. Thus, groups, new connections are made, become complete subgraphs of the network. Another assumption is that no one can be in more than one group.

### Random Grouping:

A random grouping algorithm was developed as both a conceptual and computational baseline. In this strategy, groups of the specified number of individuals are chosen at random.

In the case of non-random grouping, there are two sets of decisions to be made. 1) How is the first group member chosen, 2) How are subsequent group members chosen. In some cases, the same technique can be used for choosing the first and then all group members, but it doesn’t have to be so.

### Maximum Distance – *Gregarious Grouping*:

In this approach, network distance is used to determine groups such that group members are chosen because they are as far as possible away, it preferentially builds groups, and thus connections, among those that are furthest apart. The first member of each group is randomly chosen from ungrouped individuals that possess a distance equal to the maximum distance that can be found in the network. For example, the IO office representative is at most a distance of 3 away from any other manager, but branch chiefs in different centers are all a distance of 6 away from each other. So, if there are any ungrouped branch chiefs from different offices, they will be chosen as the first member of any new groups first. Once a first member is assigned to the group, all additional members are added sequentially and uniformly randomly from the ungrouped members that are a maximal distance away. See Figures ### for a visualization of this algorithm in action.

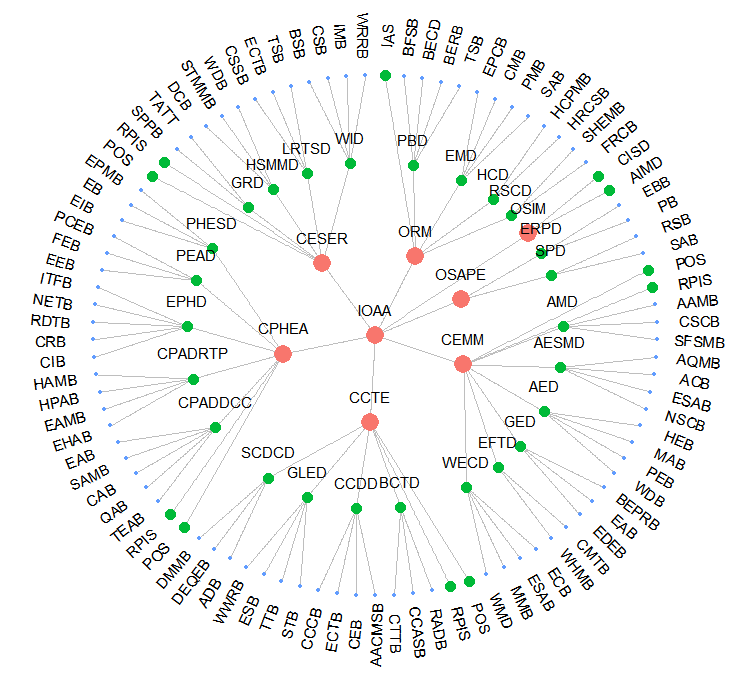
### Minimum Distance – *Shy Grouping*:

The shy approach to grouping is, in some senses, the opposite of the gregarious algorithm. Once a first group member is chosen, all subsequent group members are randomly chosen from those that are a minimum distance away. For example, if a branch chief is the first member in a new group, the next member will always be the division director (if they aren’t already in a group). Grouping proceeds in this way, preferentially adding group members based on those that are closest to the first member. The first individual in the group is randomly chosen from among those individuals that have a distance in the network that is minimum globally. This choice of starting individual will preferentially choose individuals that have higher degree, since degree is equivalent to the number of connections of length 1 that an individual has. Before any grouping, all individuals in the ORD network have at least one person they are a distance of 1 from, however, branch chiefs each only have a single connection of length 1, to their division director, whereas division directors and other upper management can (and often do) have more connections of distance 1. Therefore, upper management are preferentially chosen to start new groups, this is in opposition to the maximum algorithm where branch chiefs are preferentially chosen to start new groups.

# Results and Discussion

## The ORD Network

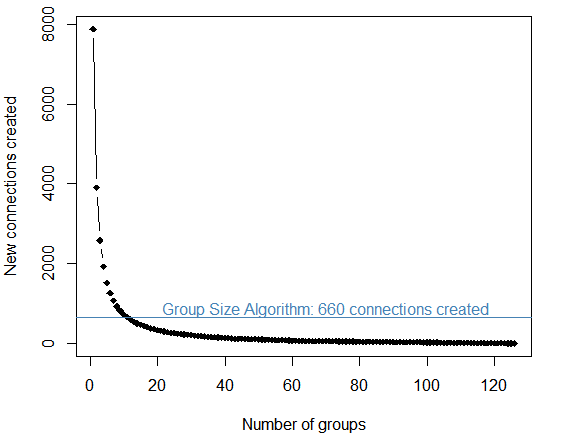
Under the new organization the ORD network of managers has the following basic attributes. There are 126 nodes, 125 edges. The current network is considered a rooted tree, with the IO office as the root. A visualization of the network is given in Figure ##.

In this network the following can be observed - the most remote connections are those between branch chiefs in different office/centers, these are distance 6 in the network. All direct supervisory links are distance 1. The average distance of all managers in the ORD network is 4.8. The IO is an average distance of 2.6 from the rest of the managers and branch chiefs are an average distance of 5.1 from the rest of the managers.

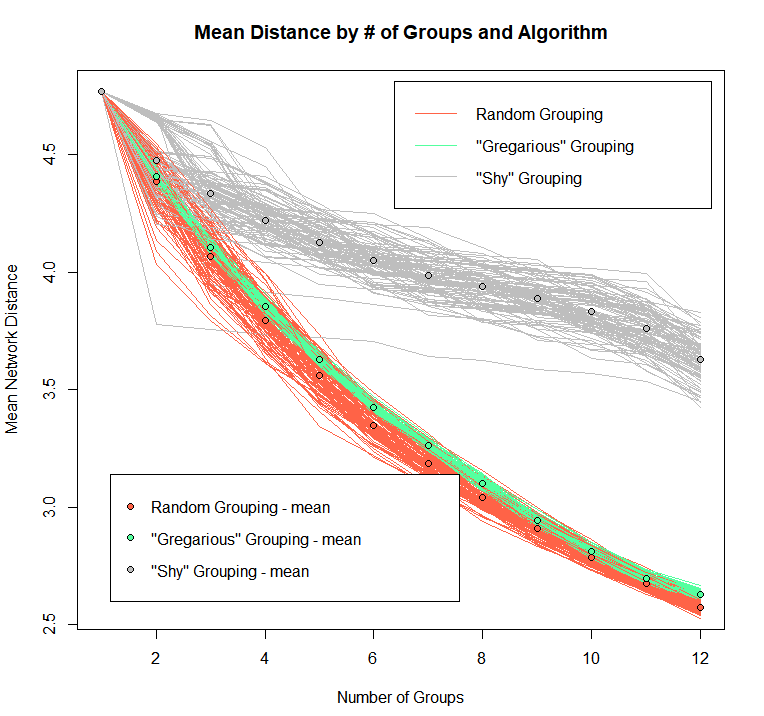
## Grouping Results

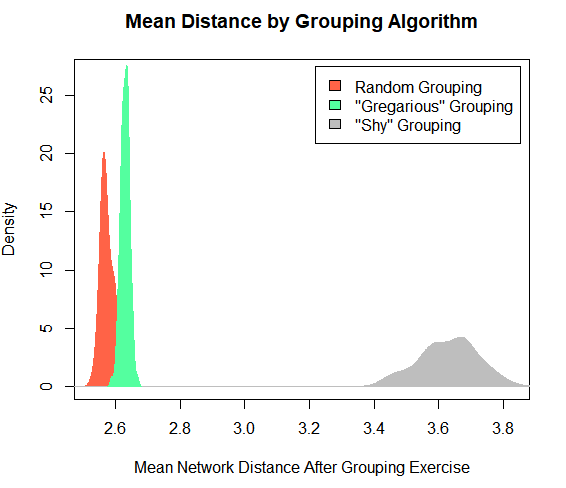
### Group Size

Using the group size algorithm, the 126 people in the network were broken down into 11 groups. 5 of the groups had 12 people and 6 of the groups had 11. The edges created by groups of this size is 660.

Different group sizes create different numbers of new connections, even if the number of people being grouped doesn’t change. A plot of the number of new connections made by group size using the group size algorithm for the ORD network is given in Figure ###. As it can be seen, the number of new connections decreases monotonically as group size decreases, in other words, as group sizes become smaller and smaller, less connections are made through the grouping exercise. The extreme cases include everyone in a single group, where the maximum number of connections are made. The other extreme case is if everyone were in their own group, in which case no new connections are made.

### Grouping Strategies

Each grouping technique has a stochastic element to it. The random technique is completely so. While the stochastic elements to the gregarious and shy groupings, result from making uniformly random choices among possible new connections if they are a maximum or minimum distance from the first group member, which is also chosen at random. For example, if a branch chief is the first group member chosen in the gregarious grouping, any other branch chief from a different office is equally like to be chosen and added to the group since they are all a maximum distance (6) away. If a division director is chosen to start a group in the shy grouping approach, their center director or any of their branch chiefs will be added to the group with equal probability, since they are all a minimum distance (1) away. To account for stochasticity in the grouping algorithms, the network grouping was repeated 100 times for each grouping strategy, Figures ### and ### show the results.

Surprisingly, the random grouping approach turned out be the most effective of the three at lowering the average distance in the network. The gregarious grouping was second, and the shy grouping, was, unsurprisingly, the worst at building connectivity.

The random and gregarious grouping strategies were very close in result, and the random grouping had a higher variance than did the gregarious strategy. The shy grouping had the largest variance. This, again, is surprising, since one might anticipate that the random grouping would have the largest variance of the three approaches. Further investigation is needed to understand why the shy algorithm shows the largest variance, one idea is this: since every person in the network has a connection of distance one, anyone can be chosen to start the groups, whereas in the gregarious grouping not all individuals have connections of distance 6, therefore, all of those individuals will be chosen to start the groups before anyone else. Animations of simulated group can be found at <https://nathanpollesch.files.wordpress.com/2019/07/nets_g_tree-1.gif>