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

# Introduction

EPA’s Office of Research and Development undergoing a 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” concept. In a nutshell, this number is an indication on how far each actor/actress is from Kevin Bacon. For example, if you were in a film with Kevin, your Kevin Bacon number is 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 mean *distance* in the ORD 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, branch chiefs are a distance of one from their division director, 2 from their center directors, and so on. 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.

## 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 11. 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 needing to be grouped. However, it is understood that there may be constraints on the total number of groups for a given venue. Thus, the algorithm has the option of allowing for the number of groups to be specified, in which case it is balances the number of people among the number of groups specified. The algorithm will never return more than 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 group of 12. The exact form of the algorithm is in the R Code provided in 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. In this work, three grouping strategies were derived and studied; “*gregarious”* a maximum distance grouping, “s*hy”* a minimum distance grouping, and “*pass out playing cards or draw numbers from a hat”* a random grouping. For each grouping strategy an algorithm was developed and programmed in R (R code is 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, once chosen, are made to be complete subgraphs of the network. In addition to complete connectivity of groups, another assumption is that no one can be in more than one group.

### Random Grouping:

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

A visualization of this algorithm in action is available here, [gregarious grouping](https://github.com/npollesch/ORDNetwork/blob/a7801bdcaf01e5d8c4968feafbc8db887edf6d71/ORD-Network/images/nets_g_tree.gif), on Nate Pollesch’s Github site.

### 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 a global minimum.

Of note, 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, and that is to their division director, whereas division directors and other upper management can (and often do) have more connections of distance 1. Therefore, upper-level managers are preferentially chosen to start new groups in the shy algorithm, this is in opposition to the maximum algorithm where branch chiefs are preferentially chosen to start new groups.

A visualization of the shy grouping algorithm in action is available here, [shy grouping](https://github.com/npollesch/ORDNetwork/blob/a7801bdcaf01e5d8c4968feafbc8db887edf6d71/ORD-Network/images/nets_s_tree.gif) on Nate Pollesch’s github site.

# Results and Discussion

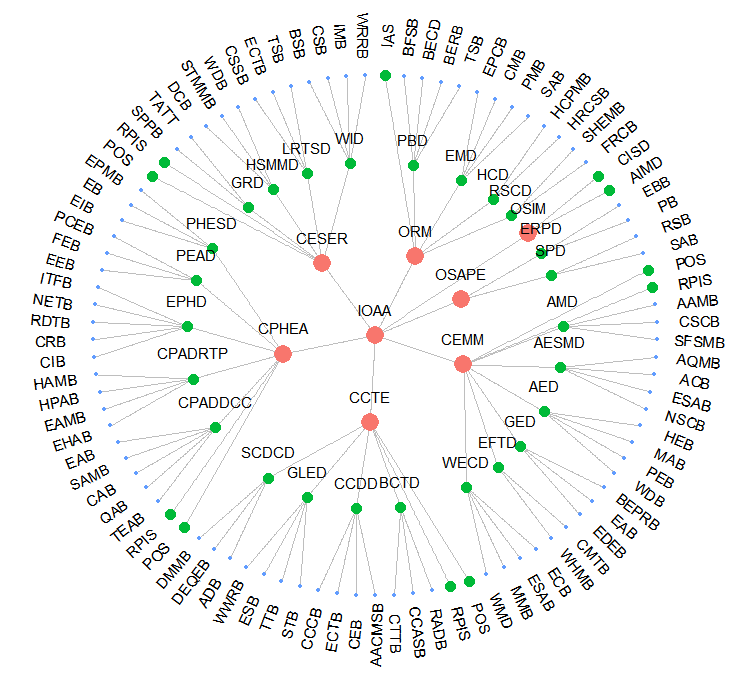


Figure 1: The Reorganized ORD Network. Center/Offices are represented by the largest nodes, followed by divisions and then branches. This network was assembled based on the proposed reorganization depicted in the diagrams available at https://intranet.ord.epa.gov/about-ord/proposed-ord-organizational-structure

## 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

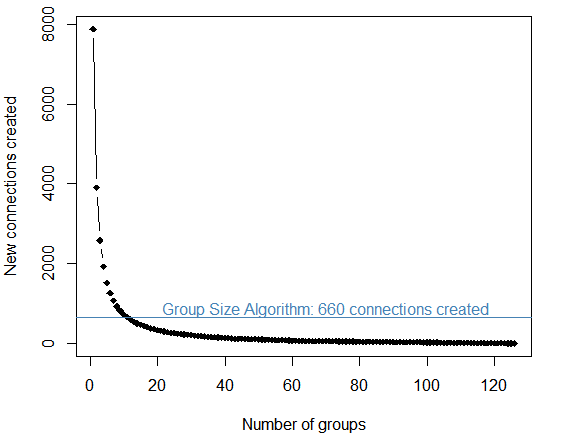


Figure 2: New connections gained by increasing number of groups. Using a total number of 126 nodes, this figure shows how number of new connections decreases monotonically as the number of groups grow and the group sizes get smaller and smaller.

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, with the random technique being completely so. The stochasticity in the gregarious and shy groupings result from making uniformly random choices among possible new connections that 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 closeness.

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. An animation for sequential grouping in the gregarious and shy algorithms are hyperlinked here, [gregarious grouping](https://github.com/npollesch/ORDNetwork/blob/a7801bdcaf01e5d8c4968feafbc8db887edf6d71/ORD-Network/images/nets_g_tree.gif), and here, [shy grouping](https://github.com/npollesch/ORDNetwork/blob/a7801bdcaf01e5d8c4968feafbc8db887edf6d71/ORD-Network/images/nets_s_tree.gif).

The random and gregarious grouping strategies were very close in result, but the random grouping had a higher variance than the gregarious strategy. The shy grouping had the largest variance. The shy grouping having the largest variance is once again, surprising. One might anticipate that the random grouping would have the largest variance of the three approaches.

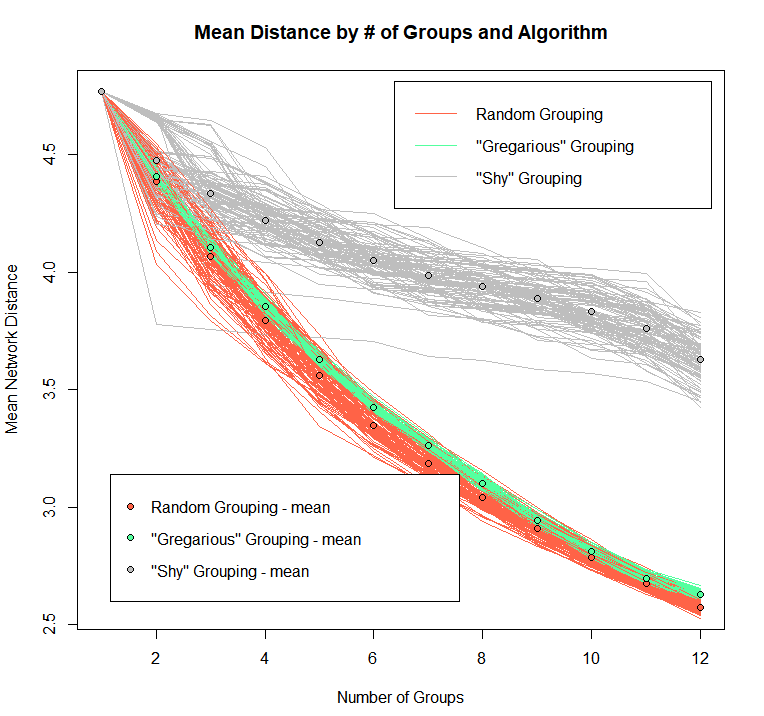


Figure 3: Mean distance in ORD network as successive groups are added. Three different grouping strategies are shown; random, gregarious, and shy. Each strategy was simulated 100 times, simulations and mean values after each group have been added are shown.

Further investigation is needed to fully pinpoint why the shy algorithm shows the largest variance. One idea is that 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. So, the types of individuals, (DD,BC, etc…) chosen to start the groups will vary much more widely, perhaps leading to more variance in distance reduction achieved by the groups.

The three algorithms explored represent three possible strategies to form groups. The maximal grouping approach, in practice, would take some effort to implement prior to an event without the aid of the computer, one might have difficulty in strategically building groups of folks that are maximally distant in the organization. The minimum grouping approach is likely to the be the result if individuals were left to form groups, assuming that individuals would tend to group with those that they already know well.

The fact that random grouping did better than the maximum distance greedy grouping highlights an important subtlety to this exercise; there are better connections to make than connections between two people that are the furthest apart. Given the approaches utilized here, it also points out the fact that decisions that are locally optimal, i.e. connecting two individuals that are maximally distant, do not always lead to globally optimal solutions. For example, a new connection between two center directors only reduces their distance from 2 to 1. However, managers under them in the hierarchy would then have their distance reduced by one as well, thus the total distance gained with that connection can certainly be larger than total distance gained by connection two branch chiefs that are a distance of 6 apart. With further investigation, one could certainly systematically explore the types of connections that have the largest effect on connectivity. One might also develop algorithms to represent other grouping approaches. For example, a stratified grouping approach could be developed that sought to balance the levels of managers in the groups (see Future Directions).

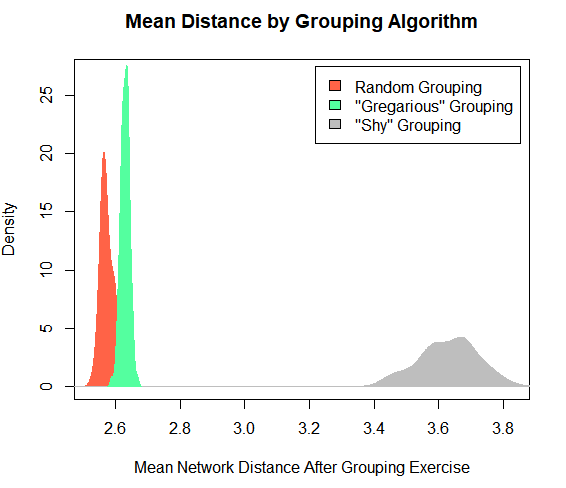


Figure 4: Distributions of mean distance after all groups were created. The random grouping had the lowest mean distance, followed by gregarious, then shy. Shy had the most variability in results, followed by random, then gregarious.

Considering implementation of strategies also brings to the surface an important point, an organization’s social network is likely to be different from their organizational network. For example, a center director may be well acquainted with everyone in their center, as well as the other center directors; Branch chiefs from a single division might be much more well acquainted to each other than their division director. The intersection of the ORD social network with the ORD organizational network, would make for an interesting set of dual constraints to considering when optimizing connectivity. There are many dimensions to social interaction and many ways in which groups could be chosen in order to bring an organization closer together. Can one truly judge a kinship to Kevin Bacon based solely on the films they co-starred, or co-co-starred in? Of course, one cannot.

# Recommendations and Implementation

The results of this work were quite surprising. A mathematician might even feel some disappointment that their algorithm to systematically scan the network and sequentially build groups from those folks that are furthest apart was outperformed by simple randomization. However, a *practical* mathematician might also take delight in the fact that the recommendation is quite simple. Grouping people randomly works surprisingly well at building new connections. Assuming that the shy grouping approach is analogous to an organizer saying, “Form up into groups”, and letting people form their own groups, then taking the lead and randomly grouping people will do much better on average for bringing the whole organization closer.

# Future Directions

The ORD organizational structure is not some random assemblage of people. It is a hierarchical structure. However, the branching is less regular than a standard tree branching structure where each manager in the hierarchy has the exact same number of managers below and above them. It seems feasible to be able to determine a globally optimal approach within a regularly branched hierarchical structure, in doing so, one might gain insight into the global effect of introducing specific types of connections. For example, if connections are built from the bottom up, top down, or across a hierarchical level, how does each add to the closeness of the network? The fact that a greedy, locally optimal grouping strategy, was beat by random grouping exemplifies that answer has more subtlety than one might expect. However, given the applicability of results from this research effort, it may be worth pursing the regular branching case and attempting to extrapolate results to a less regular hierarchy, such as that present in the ORD network.

Only three of many feasible strategies were explored, and only two of those involved systematic grouping to build connections. There are likely to be many more desirable strategies that would be worth investigating. A stratified, i.e. grouping across approach was mentioned previously, and although involves more computational complexity to implement, would be possible and of interest moving forward.