Generating Biological Networks: A Software Tool

Brian Mulhall
University of Connecticut
Project Outline

- Problem Statement/Motivation
- Methodology
- Current Results
- Future Work
Problem Statement/ Motivation

- lack of any standardized set of input graphs
- Leading to an inconsistent set of results
Project Goal

- Implementation of a comprehensive software solution
- Capable of generating a biological network with selected properties
Methodology

• Identify the relevant network properties desirable in a “real world” network

• Choose an appropriate algorithm to generate network with those identified properties

• Test the efficacy of the chosen approach to provide insight into the creation of a final software tool
Network Properties and Topological Structure

- Network Modularity (Hierarchical Organization)
- Small Mean Geodesic Distance (Small World Property)
- High Local Clustering Coefficient
- Power Law Degree Distribution
Graph Properties

Small World Property

• $L := \text{The mean geodesic distance}$

$$L \propto \log N$$

Network Modularity

Twitter network: June 06 2011.
User: Krislikesmath
Ericksondata.com
Made Using Gephi and NodeXL.
Clustering Coefficient
Power Law Degree Distribution

$P(k) \sim k^{-\gamma}$
Program Functionality
Degree Algorithm

Input Parameters
Clustering Coefficient
Degree Sequence

Algorithm:

```
DEG(n, d, C)
1 /* input: number of nodes n, degree sequence array d and target clustering coefficient C */
2 /* output: graph (V, E), where V is set of nodes and E is set of edges */
3 V ← {1, 2, ..., n}
4 E ← ∅
5 M ← \frac{\sum_{i \leq j} d[i]}{2} /* number of edges to create */
6 rd ← d /* residual degree array */
7 T ← \frac{C \cdot \sum_{i \leq j} r[d[i]]}{3} /* number of triangles to create */
8 while T > 0
9   do Choose three distinct nodes
10      u with probability \frac{rd[u]}{\sum_{i \leq j} rd[i]}
11      v with probability \frac{rd[v]}{\sum_{i \leq j} rd[i]}
12      w with probability \frac{rd[w]}{\sum_{i \leq j} rd[i]}
13      Check rd[u], rd[v] and rd[w]
14      Add edges needed to create a triangle between u, v and w
15      update rd[u], rd[v], rd[w], T and M appropriately.
16
17 while M > 0
18   do Choose two distinct nodes
19      u with probability \frac{rd[u]}{\sum_{i \leq j} rd[i]}
20      v with probability \frac{rd[v]}{\sum_{i \leq j} rd[i]}
21      if (u, v) \notin E
22         then E ← E ∪ \{(u, v)\}
23         rd[u] ← rd[u] − 1
24         rd[v] ← rd[v] − 1
25         M ← M − 1
26 return (V, E)
```

Figure 8.1: Degree sequence algorithm DEG
Newman Algorithm

Input Parameters

Triangle Degree

Single Edge Degree

\begin{figure}[h]
\begin{algorithm}
\caption{Algorithm CONF-1}
\begin{algorithmic}
\State \textbf{Input Parameters:}
\State Triangle Degree
\State Single Edge Degree

\State \textbf{Output:} graph \((V, E)\), where \(V\) is a set of nodes and \(E\) a set of edges
\State \(V \leftarrow \{1, 2, \ldots, n\}\)
\State \(E \leftarrow \emptyset\)
\State \(rt \gets i\) \Comment{residual triangle degree array}
\State \(rs \gets s\) \Comment{residual single edge degree array}
\State \(T \gets \frac{1}{2} \sum_{u \in V} \text{rt}[u]\) \Comment{number of triangles to create}
\State \(M \gets \sum_{u \in V} \text{rs}[u]\) \Comment{number of single edges to create}
\While {\(T > 0\)}
\Do {Choose three distinct nodes \(u, v, w\) from 1 to \(n\) with equal probability \(\frac{1}{n}\)}
\If {\(rt[u] > 0\) and \(rt[v] > 0\) and \(rt[w] > 0\)}
\Then\ \(E \leftarrow E \cup \{(u, v), (v, w), (u, w)\}\)
\State \(rt[u] \gets rt[u] - 1\)
\State \(rt[v] \gets rt[v] - 1\)
\State \(rt[w] \gets rt[w] - 1\)
\State \(T \gets T - 1\)
\EndIf
\EndDo
\While {\(M > 0\)}
\Do {Choose two distinct nodes \(u, v\) from 1 to \(n\)}
\If {\(rs[u] > 0\) and \(rs[v] > 0\)}
\Then\ \(E \leftarrow E \cup \{(u, v)\}\)
\State \(rs[u] \gets rs[u] - 1\)
\State \(rs[v] \gets rs[v] - 1\)
\State \(M \gets M - 1\)
\EndIf
\EndDo
\EndWhile
\State \textbf{return} \((V, E)\)
\end{algorithmic}
\end{algorithm}
\end{figure}

Figure 7.1: Algorithm CONF-1
Klemm-Eguillez Algorithm

For: all nodes $i = 1...N$
For: all nodes $j$ = the set of all nodes adjacent to node $i$
Set: Chance = a uniform random variable between 0 and 1
If: $d > \text{Chance}$
Set: node $h$ = uniformly randomly chosen from the set of all nodes, excluding $i$ and nodes adjacent to $i$
Disconnect: directed edge from node $i$ to node $j$
Connect: directed edge from node $i$ to node $h$
EndIf
EndFor

//Replace an active node with node $i$. Active nodes with lower degrees are more likely to be replaced.
Set: node $i$ as an Active_Node
While: node $j$ is not chosen
Set: $j$ = uniformly randomly chosen node from Active_Nodes
Set: $p_i = \frac{1/k_j}{\text{sum}(1/k)}$
Set: Chance = a uniform random number between 0 and 1
If: $p_i > \text{Chance}$
Set: node $j$ as chosen
Remove: node $j$ from Active_Nodes
EndIf
EndWhile

Edge Switching
Incorporated to preserve the small world property

Preferential Attachment
Used to promote the generation of a network with hubs
Acknowledgements

Special Thanks
Saad Quader
Professor Huang