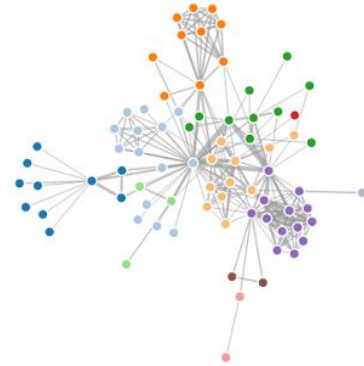


## CS 519 Scientific Visualization

### Machine Problem 3: Due December 4, 2015 at 11:55pm

#### Force Directed Graph Layout

You will implement the force-directed graph layout algorithm of Fruchterman and Reingold. The original paper can be found [here](#).



#### Implementation Requirements:

1. You will use an HTML5 canvas and JavaScript to implement the algorithm.

Your code should consist of the following files:

*GraphForceLayout.html*

You may base your code off of the code available on GitHub at

[https://github.com/shaffer1/Ullinois\\_SciVis/blob/master/MP3/GraphForceLayout.html](https://github.com/shaffer1/Ullinois_SciVis/blob/master/MP3/GraphForceLayout.html)

You can grab a copy of *GraphForceLayout.html* from the repo to serve as starter code.

2. The node positions in the graph are generated iteratively by calculating attractive and repulsive forces for each node:

$$F_a(n_i, n_j) = \frac{\|p_i - p_j\|}{k} (p_j - p_i)$$

$$F_r(n_i, n_j) = -\frac{k^2}{\|p_i - p_j\|} (p_j - p_i)$$

- $F_a$  is the attractive force calculated along graph edges
  - $F_r$  is the repulsive force calculated between all pairs of nodes
  - $n_i$  and  $n_j$  are nodes in the graph
  - $p_i$  and  $p_j$  are the positions in space of  $n_i$  and  $n_j$
  - $k$  is a constant typically set to  $\frac{\sqrt{A}}{N}$ 
    - $A$  is the area of the canvas
    - $N$  is the number of nodes
3. You may need to cap the maximum movement of a node per iteration as some constant  $t$  and the decrease that by some fraction  $\Delta t$  at each iteration
  4. (Not required, but...) You can animate the layout computation using:  
<http://www.html5canvastutorials.com/advanced/html5-canvas-animation-stage/>

#### Data

For the graph, use a randomly positioned cycle graph. You can generate an  $n$  node random

cycle with the following JavaScript:

```
function random_cycle(n,lim_x,lim_y)
{
    var edges=[];
    for(var i=0;i<n;i++)
    {
        edge = {"target":(i+1) % n, "source":i}
        edges.push(edge)
    }
    var nodes=[]
    for(var i=0;i<n;i++)
    {
        var rand_x = Math.random()*lim_x;
        var rand_y = Math.random()*lim_y;
        node = {"x": rand_x, "y": rand_y}
        nodes.push(node)
    }
    var g = {"nodes":nodes,"edges":edges};
    return g;
}
```

### **Submission**

Submit using Compass. Upload the following:

*GraphForceLayout.html*