

# Artisoc で作成する基本的 Fruchterman Reingold アルゴリズム シミュレーションとコミュニティ抽出について

末 木 俊 之\*

## Basic Fruchterman Reingold Algorithm Simulation with Artisoc and Community Extraction

Toshiyuki SUEKI

### Abstract

I have created a simulation based on the Fruchterman Reingold algorithm with the simulator artisoc. It was able to finely set the value of the various parameters that affect the simulation with control panel. Simulation was easy to create. I was confirmed that the various parameter settings can affect the display. When the number of nodes is small, node arrangement is beautiful. And it is possible to extract community from the display. When the number of nodes is large, node arrangement is chaotic. And it is not possible to extract community from the display. It was not the expected result.

キーワード：p a j e k   ばねモデル

### 1. *Fruchterman Reingold* アルゴリズムによる *artisoc* シミュレーション作成

2014年度研究紀要<sup>1)</sup>では、BAモデルネットワークを Pajek<sup>2)</sup>の Fruchterman Reingold コマンドを使い自動レイアウト表示をさせてみた。あらかじめ Artisoc<sup>3)</sup>で最適なコミュニティ抽出を行い、各ノードに所属コミュニティ番号付与済のネットワークモデルを使ったので、表示結果をコミュニティ分割の視点から観察できた。適切な Factor 値を指定して Fruchterman Reingold コマンドを実行すると目視で大まかなコミュニティ分割（抽出）を確認可能な表示となっていた。以下の4点が注目される。(1)ノード（点）が円形内に配置されている。(2)重要度の高い（より多くのノードとリンクしている）ノードがより円の中心に来るように配置されている。(3)同一コミュニティに属する点が扇型に近い範囲内に集めて表示されている。(4)ノード同士をつなぐリンク（線）の交差が最小化された表示にもなっている。

Pajek の Fruchterman Reingold コマンドは、ばねモデルに基づく計算を全ノードで実行するステップを繰り返し実行し、ある程度経過した時点でネットワーク表示を行っている。表示されるまで少々待たされる。一旦ネットワーク表示を更新した後、再度計算処理状態に入る。それを繰り返している。

---

\*人間健康学部 健康栄養学科

ばねモデルでは、1ステップの計算処理毎に、全てのノードについて自分に作用している力（全ての反発力と吸引力の合力）を計算し、その力に基づいて次の表示位置（座標）を決定する。Artisocのようなシミュレータで作るなら、1ステップ毎にネットワーク表示を更新する動作が自然であろう。Pajekはシミュレータでは無く、計算処理1ステップ毎の表示の更新・変化を確認することはできない。またノード（点）間距離を「Factor」値で指定することができるが、それ以外の計算処理上のパラメータを自由に覚えてコマンドを実行できない。ノード間距離以外のパラメータもネットワーク表示に影響すると思われる。シミュレータ Artisoc で Fruchterman Reingold アルゴリズムに基づくシミュレーションを作成すれば、1ステップ毎のネットワーク表示の変化が観察できる。各種パラメータがネットワーク表示に影響する効果も確認できるであろう。また、Artisoc

で作成したシミュレーションでもコミュニティ抽出（分割）が目視で確認可能な表示が生成されるか確認したい。

基本的なばねモデルによるネットワーク自動レイアウトアルゴリズムでは、ノードを帯電粒子とみなし、リンクしないノード間には反発力が働き、リンクするノード間にはフックの法則に従うばねに相当する力が作用すると考える。1ステップの計算処理では、各ノードで自分に作用する反発力と吸引力の総計を計算し、それに基づき新たな座標に移動する。全てのノードの新座標が計算されて1ステップの処理を終える。そのステップを繰り返すことにより次第に全ノードに作用する力の総計が小さく収束し、配置が安定する。Fruchterman Reingold アルゴリズムは、ばねモデルの一種であり、より速い計算、速い収束を得るために以下の改良が加えられている。<sup>4)</sup>

(1) 全てのノード間には、式1の反発力が働き、リンクしているノード間には、式2の吸引力が働くと考える。

式中の $p_v$ は、ノード $v$ の座標ベクトル $p_v = (x_v, y_v)$ を意味する。

式1は、ノード $v$ に作用する他のノード $u$ に対する反発力ベクトルの式である。式2は、ノード $v$ に働くリンクするノード $u$ への吸引力ベクトルの式である。 $len$ はノード同士を繋ぐと仮定するばねの伸縮されていない自然な長さ（ばね自然長）である。

$$f_{rep}(p_u, p_v) = \frac{len^2}{||p_u - p_v||} \cdot \overrightarrow{p_u p_v} \quad \text{式1}$$

$$f_{attr}(p_u, p_v) = \frac{||p_u - p_v||^2}{len} \cdot \overrightarrow{p_v p_u} \quad \text{式2}$$

図1は、ばね自然長=1の場合にノード間距離とノード間に作用する力のグラフである。リンクするノード間には反発力と吸引力の合力が働くと考える。

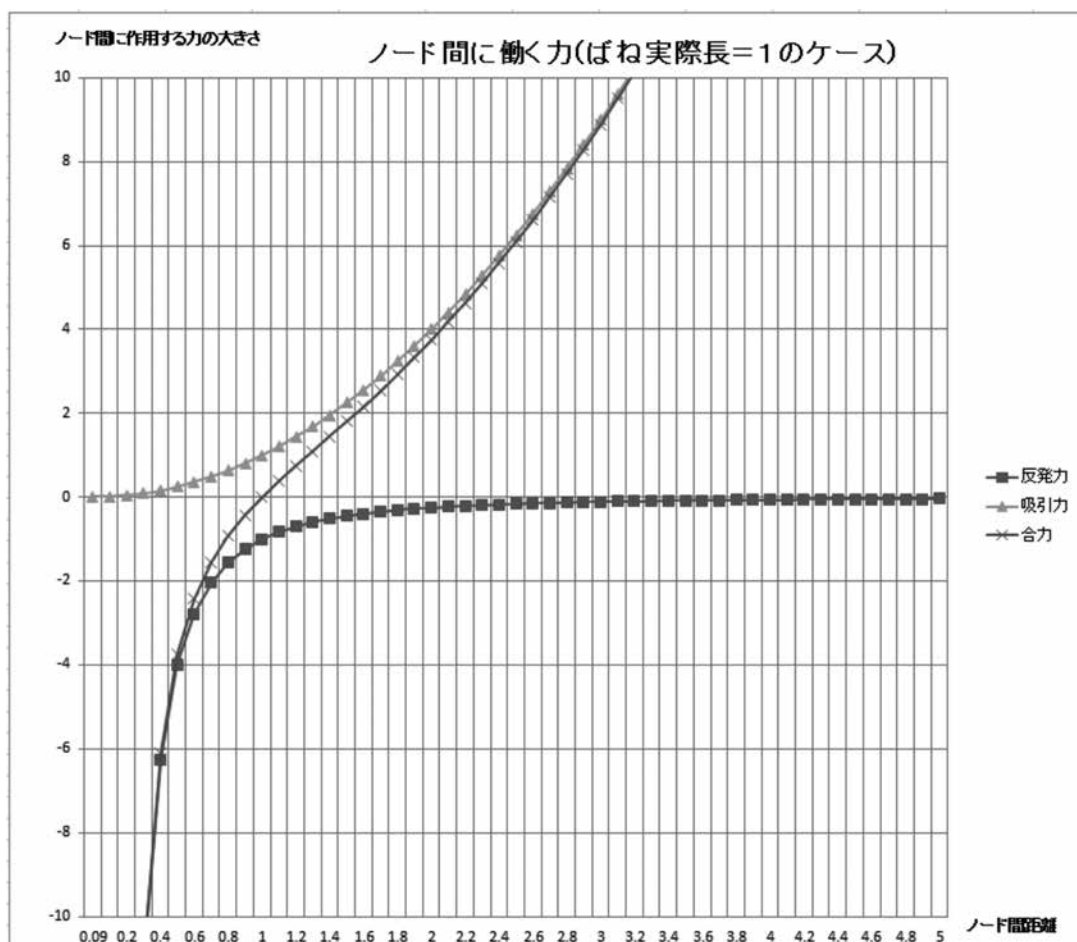


図1 ノード間距離とノードに作用する力

(2) 計算を速くするために、反発力計算は近傍ノードのみ対象とする。Artisoc でシミュレーションを作成するなら、これは簡単に実現できる。MakeAllAgtsetAroundOwn コマンドを使用すると、自分(ノード)の近傍に存在するオブジェクト(エージェント)の集合を容易に取得することができる。

(3) ノード  $v$  はある時点 ( $t$ ) で自分に作用する力ベクトルの総計  $F_v(t)$  に基づいて新しい座標に移動するが、移動ベクトルが大きすぎると、収束までに時間がかかる。そこで力の作用を減ずるファクター  $\delta(t)$  を掛けた値  $F_v(t) \cdot \delta(t)$  が使われる。またこのファクターはステップが進むにつれて次第に小さくして、収束近くに大きな変動を起こすことを防ぐ。

(4) 座標の刈り込みを行う。ノードの新座標が考慮中の空間を逸脱した座標になることが多い。その場合には考慮中の空間内に収まるように移動ベクトルを縮める。

(5) 振動・回転(ローテーション)の防止

各ノードに作用する前ステップの力  $F_v(t-1)$  を記憶し、 $F_v(t)$  と方向を比較する。反対方向 ( $180^\circ$ 、 $-180^\circ$ ) だと振動を繰り返し収束までに時間がかかる可能性がある。また角度差が垂直方向 ( $90^\circ$ 、 $-90^\circ$ ) に近いと、ノードが回転移動を起こし収束までに時間がかかる可能性がある。好ましくない角度差の場合には  $F_v(t)$  の効果を減ずるファクターを掛ける。今

回のシミュレーションでは、 $F_v(t-1)$ と $F_v(t)$ の角度差に基づいて、単純に図2のファクターを掛けて計算している。

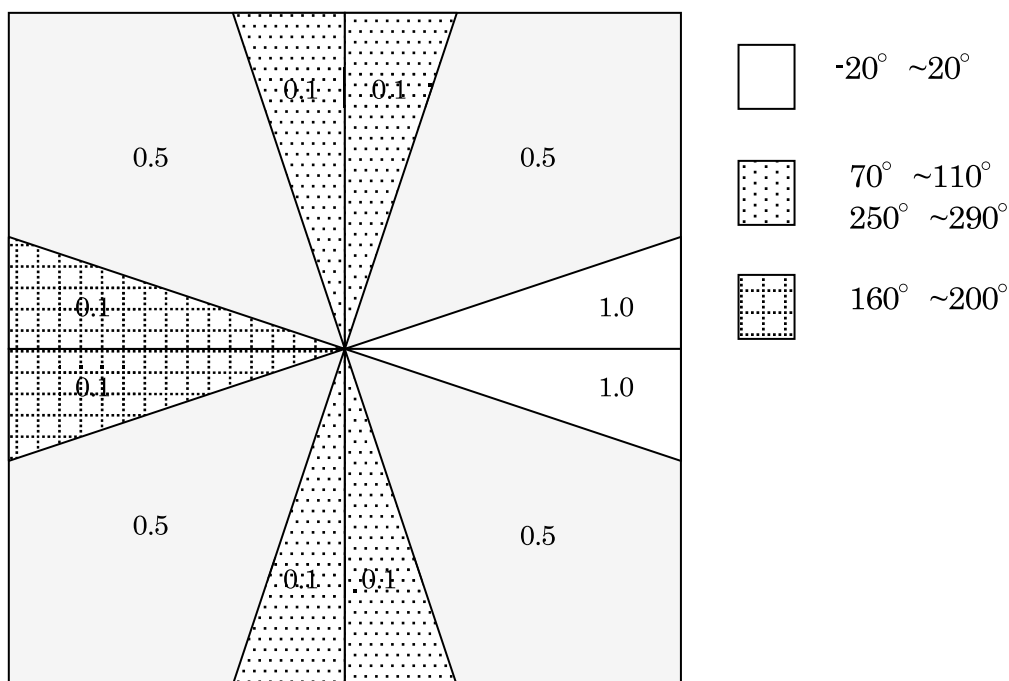


図2  $F_v(t-1)$ と $F_v(t)$ の角度差に基づくファクター

以上(1)から(5)を満たすFruchterman Reingold アルゴリズムに基づく Artisoc シミュレーションを作成した。各種パラメータは、図4のコントロールパネルで初期値を指定する。

「反発力限界値設定」スライドバーは、(2)の反発力の作用する近傍（ノードからの距離）を設定する。シミュレーション空間 X 座標の最大値（横幅）を1とした比率で設定する。今回作成したシミュレーションの X 座標最大値は50なので、図3の設定「反発力限界値設定」=0.1では、実際値は $50 \times 0.1 = 5$ となる。

「 $\sigma$ 値設定」スライドバーは、(3)の力の作用を減ずるファクター $\delta(t)$ 初期値を設定する。

「ばね自然長設定」スライドバーは、ノード同士を繋ぐと仮定するばねの自然長を設定する。これも

シミュレーション空間 X 座標の最大値（横幅）を1とした比率で設定する。図3の設定では、実際のばね自然長は $50 \times 0.1 = 5$ となる。

「 $\sigma$ 値減少分」スライドバーは、シミュレーション1ステップ毎に $\delta(t)$ から減ずる数値を意味する。

「反発力倍率」スライドバーは、ノード間に作用する反発力に掛け、反発力の効果を強めるパラメータを設定する。実際にはあまり利用しなかった。

「設定値更新」トグルボタンは、ON にするとコントロールパネルで設定する各種パラメータ値を再度読み込み直してそれ以降のステップの計算処理を実行する動きをする。シミュレーションの途中からパラメータ値を変化させてシミュレーションを実行できる。これも実際にはあまり利用しなかった。

「振動回転考慮」トグルボタンがON の場合は(5)の振動・回転を防止する機能が働く。OFF では働

かない。



図3 各種パラメータ値を設定するコントロールパネル

## 2. シミュレーション実行結果

### 2-1. ある大学の空手部部員の友人関係を表現したグラフ図表示結果

参考にした書籍で扱われていたノード数34、コミュニティ数2のネットワーク<sup>5)</sup>について、Artisocで

作成したFruchterman Reingoldアルゴリズムシミュレーションを実行した。パラメータの初期値は、反発力作用域=5.0、ばね自然長=5.0、 $\delta(t)=0.05$ 、 $\sigma$ 値減少分=0.0001、反発力倍率=1である。

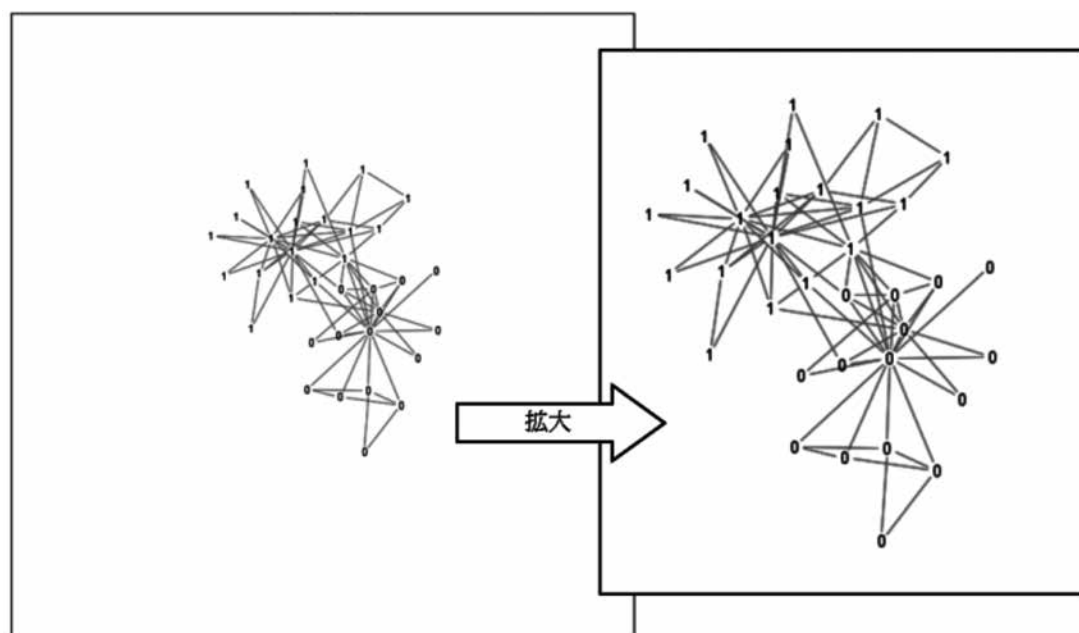


図4 米国のある大学の空手部部員の友人関係を表現したネットワーク表示結果

図4は、シミュレーション開始後132ステップ時の表示である。2つのコミュニティに分かれて、ノードは綺麗に配置されている。しかし、Pajek による表示での同一コミュニティに所属するノードが扇型（円の一部）領域に集められ配置される傾向は弱い。

振動・回転（ローテーション）防止機能は有効であった。図3コントロールパネルの「振動回転考慮」

トグルボタン= OFF でシミュレーションを実行すると、振動・回転（ローテーション）防止機能は働かない。すると明らかに収束が遅れた。図5のシミュレーション例では、ステップ182と183で逆方向に振動して動くノードの存在が確認された。何ステップにも亘りこのようにノードが振動した後に、ステップ200付近で表示がほとんど変化しない状態に収束した。

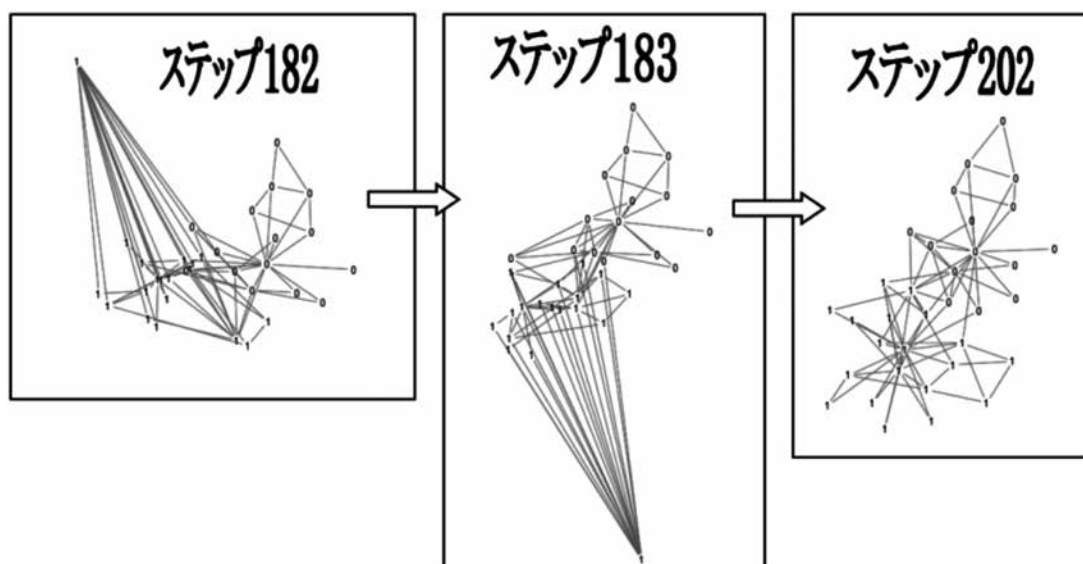


図5 振動するノードの存在を示す表示図

Pajek の Fruchterman Reingold 自動レイアウト表示では、ばね自然長が小さい場合、ネットワーク表示は円状から樹状に変化した。詳細なコミュニティ分割はされず、大まかなコミュニティ分割が強調される表示である。Artisoc で作成したシミュレーションでも同様な表示となるか試みた。反発力作用域=5.0、ばね自然長=0.5、 $\delta(t)=0.02$ 、 $\sigma$ 値減少分=0.0001、反発力倍率=1のパラメータで実行した例が図6である。ばね自然長が小さい場合、値が大きいと収束しにくい。

図6は、74ステップ時の表示である。ノード配置は棒状で2つのコミュニティは概ね分離している。



図6 ばね自然長=0.5、ステップ74での表示図

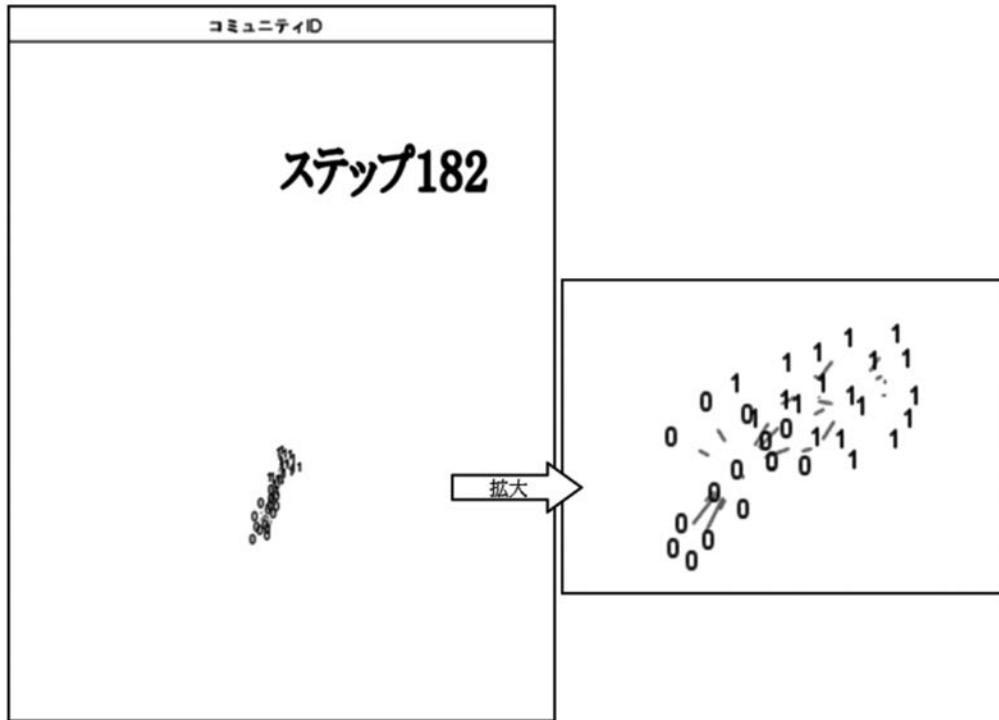


図7 ばね自然長=0.5、ステップ182での表示図

図7はさらに進めた182ステップ時の様子である。ノードが棒状配置される状態が長く続いたが、最終的に小さな棒状に縮まり、さらにつばみが開くようにノードの配置が少しばらけた状態に収束した。2014年度の研究紀要<sup>1)</sup>で、同じネットワークをPajekのFruchterman Reingold自動レイアウトで表示した場合、ノード間距離（ばね自然長）が小さい場合には図6のような棒状表示に収束したが、それは最終的には図7のように小さくまとまって表示される表示に至る途中段階ではないかと思われる。ノード間距離（ばね自然長）が小さいと表示が落ち着く（収束）までに時間がかかる。また $\sigma$ 値が大きいと収束は難しい。Pajekではノード間距離を小さく設定するに従い、自動的に $\sigma$ 値設定も小さくするプログラム上の工夫がされているのかもしれない。

## 2-2. BAモデル ( $m_0=0$ 、 $m=1$ 、ノード数2000、リンク数3998) 表示結果

ノード数が多いネットワークモデルでも同じシミュレーションを実行してみた。2014年度研究紀要<sup>1)</sup>にてPajekでFruchterman Reingold自動レイアウト表示を行った同じネットワークモデルを使った。反発力作用域=5.0、ばね自然長=5.0、 $\delta(t)=0.05$ 、 $\sigma$ 値減少分=0.0001、反発力倍率=1で実行した例を示す。1時間38分経過、716ステップ目の表示が図8である。コミュニティは綺麗に分割されない。円周上に多くのノードが集まる。

シミュレーションが進むにつれて円周上に固まったノードがほどけるように広がっていく。 $\sigma$ 値が大きい場合、また反発力作用域が大きい場合には、円周上（周辺）に配置されるノードが増える傾向がある。

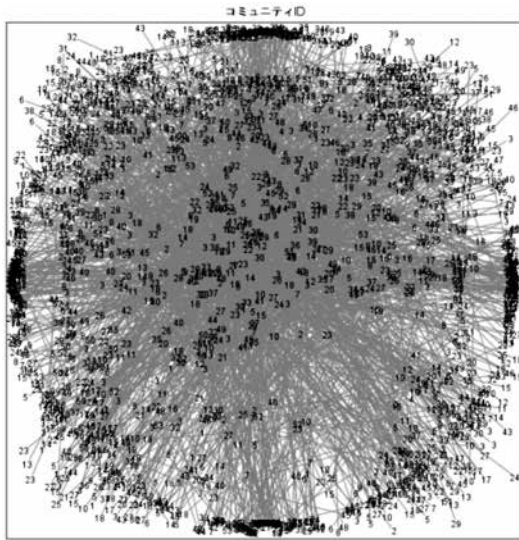


図8

次に反発力作用域=1.0、ばね自然長=5.0、 $\delta(t)=0.02$ 、 $\sigma$ 値減少分=0.0001、反発力倍率=1で実行し、20分経過、303ステップ目の表示が図9である。反発力を考慮する近傍の範囲が狭いのでシミュレーションの動作は速い。これもコミュニティは綺麗に分割されない。 $\sigma$ 値、反発力作用域が小さいと円の中心に配置されるノードが多くなる。

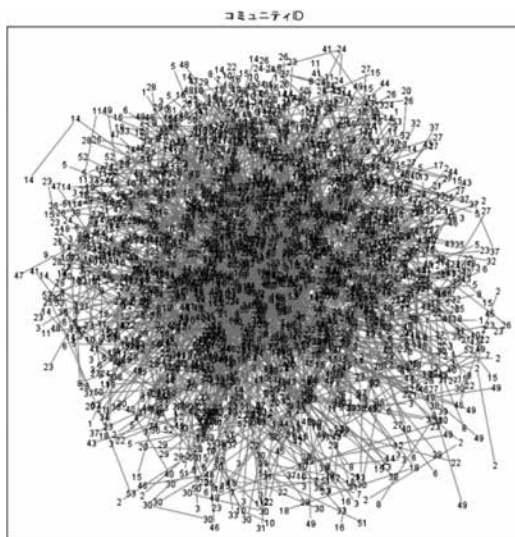


図9

その他、ばね自然長を小さくするなどパラメータをいろいろと変えて実行したが、Pajekによる自動レイアウト表示のようにコミュニティが綺麗に分割される表示にはならない。また樹状表示にもならなかった。

### 3. 結び

Fruchterman Reingold アルゴリズムに基づくシミュレーションを Artisoc で作成するのは簡単である。シミュレーション1ステップ毎に表示を更新し、ノード配置が変化する。反発力作用域を大きくするとノード数が多いネットワークではステップ毎の処理速度が極端に遅くなるが、小さな作用域を設定すれば2000ノードのネットワークでも1ステップの実行に数秒しかかからない。Fruchterman Reingold アルゴリズムに基づく計算では、各種パラメータの影響を受ける。ばね自然長、力の作用を減じるファクター値、反発力作用域等の初期設定をコントロールパネルで変更できるようにした。それらのパラメータ値によって、表示が落ち着くまでの時間（収束までの時間）、ノード配置の様子が変わることがわかる。

しかし、期待した大まかなコミュニティ分割を確認できるノード配置にはならなかった。ノード数が少ないネットワーク例では、図4のようにコミュニティ分割された表示になった。綺麗なレイアウトのノード配置である。しかし、重要なノードが中心に近く、その他のノードは周辺の円周上に配置される円形状の配置とは微妙に異なる。ばね自然長を小さくするとPajek 自動レイアウト表示のように棒状（樹状）の表示になった。最終的にノード間距離の小さな配置に収束するまでの中間段階の表示のようだ。棒状（樹状）表示のステップは長く、なかなか収束しなかった。ノード数が多いとコミュニティ分割を目視で確認できる表示とはならない。ばね自然長が小さいと単に中心に集まるノードが増えるだけで、樹状配置になる気配がない。各種パラメータ設定を調整すると、ノード数が多いネットワークモデルでも大まかなコミュニティ分割を確認できるノード配置になるのかも



しれないが、現状それは確認できなかった。

コミュニティとは、簡単に言えば密接にリンクしたノードの集合であり、同一コミュニティに属するノード同士を密接に結びつける吸引力に類似した力が働くものであろう。コミュニティとは、共通した認識・感覚・常識、悪く言えば共通した思い込み等の支配下に固く結びついたノードの集合と言える。そして別コミュニティに属するノード間には反発力に類似した力が働くのも自然な感覚である。実際、人間関係においても所属するコミュニティ内で通用する常識に外れた思想・考えを別コミュニティの人間から聞かされると強い違和感・嫌悪感を感じて離れたくなる。そう考えるとノードに作用する吸引力・反発力の作用で綺麗にコミュニティ分割されたノード配置が実現されても不思議ではない。実際 Pajek では、Fruchterman Reingold コマンドによりノード数が多いネットワークモデルでも大まかなコミュニティ分割を確認できるノード配置表示が実現されている。今回は、基本的な Fruchterman Reingold アルゴリズムに基づくシミュレーションを Artisoc で作成したが、期待した表示は得られなかった。シミュレーションのパラメータ設定の微調整、あるいはノードに働く別種の力を加えるなどの工夫がさらに必要であると考ええる。

#### [参考文献・出典]

- 1) 末木俊之 (2014) コミュニティ抽出と Pajek の自動レイアウトについて、駒沢女子大学研究紀要、p.241-254
- 2) ウオウター・デノーイ、アンドレイ・ムルヴェル、ヴラディミール・バタゲーリ著、安田雪監訳 (2009) Pajek を活用した社会ネットワーク分析、学校法人東京電機大学東京電機大学出版局、東京
- 3) 山影進 (2007) 人工社会構築指南 artisoc によるマルチエージェント・シミュレーション入門、有限会社書籍工房早山、東京

- 4) Michael Kaufmann, Dorothea Wagner (2001) Drawing Graphs: Methods and Models, p71-77. Springer-Verlag, Berlin
- 5) 林幸雄・大久保潤・藤原義久・上林憲行・小野直亮・湯田聴夫・相馬亘・佐藤一憲著 (2007) ネットワーク科学の道具箱 つなかりに隠れた現象をひもとく、p51-56、近代科学社、東京