DEIM Forum 2011 B10-4

2部グラフ可視化法の高速化

久保田大和† 伏見 卓恭†† 斉藤 和巳†,††

† 静岡県立大学経営情報学部経営情報学科 〒 422–8526 静岡県静岡市駿河区谷田 52-1
†† 静岡県立大学経営情報学研究科 〒 422–8526 静岡県静岡市駿河区谷田 52-1
E-mail: †{b08038,j09118,k-saito}@u-shizuoka-ken.ac.jp

あらまし カテゴリとカテゴリに属するオブジェクト間の関係を分析するために，2 部グラフを用いた研究が盛んである．大規模かつ複雑な 2 部グラフの有する特徴や構造を視覚的にとらえるための手段として可視化が考えられる．2 部グラフの可視化法として Spherical Embedding (既存 SE) 法がある．しかし，既存 SE 法はプログラムの実行に時間がかかってしまう．そのため本論文では，既存 SE 法に触れるとともに，実行時間を短縮する新たな SE（提案 SE）法を提案する．Yahoo ! 映画と Yahoo ! 知恵袋のデータを用いて 2 部グラフを構築し，既存 SE 法と提案 SE 法の実行時間の比較をする．実験の結果，提案 SE 法で実行時間の短縮が可能になることを確認した．

キーワード 2 部グラフ，可視化

Speed-up of bipartite graph visualization method

Yamato KUBOTA†, Takayasu FUSHIMI††, and Kazumi SAITO†,††

† School of Management and Information, University of Shizuoka
52-1 Yada, Suruga-ku, Shizuoka, 422–8526 Japan
†† Graduate School of Management and Information, University of Shizuoka
52-1 Yada, Suruga-ku, Shizuoka, 422–8526 Japan
E-mail: †{b08038,j09118,k-saito}@u-shizuoka-ken.ac.jp

Abstract In various fields, bipartite graphs are often used when analyzing the relation between categories and objects that belong to the categories. To visually uncover the feature and the structure of a large-scale bipartite graph, one natural approach might be the visualization. A good method for bipartite graph visualization method would be a Spherical Embedding (existing SE) method. However, it takes a large amount of time to obtain visualization results. In this paper, after briefly mentioning the existing SE method, we propose a new visualization (proposed SE) method for speeding-up the execution time. We compare the proposed SE mothod with the existing SE method by using data of Yahoo!Movie and Yahoo!Chiebukuro. As a result of the experiment, it was confirmed that the proposed SE method could achive significant speed-up.

Key words bipartite graph, visualization method

1. はじめに

我々が大規模かつ複雑なネットワークを理解し把握しようとするとき，その膨大さ，煩雑さゆえに困難な場合もある．そのようなネットワークの有する特徴や構造を理解する有効なアプローチとして「可視化」がある．可視化することにより，対象ネットワーク内のノードの相互関係や内在する特徴など，多くの情報をわかりやすく，直観的に把握することができる．そのため，ネットワークの可視化は重要であり，今までに様々なネットワーク可視化法が提案されている．複数のカテゴリにまたがって所属するオブジェクト間の関係を表すのには，2 部グラフが頻繁に使われている．大規模かつ複雑な 2 部グラフの効率的な可視化法として Spherical Embedding (既存 SE) 法 [3] がある．しかし，既存 SE 法では非線形処理を必要とするため，計算量がかかるという問題がある．このことはネットワークの規模が大きく，複雑になるほど顕著である．本論文では，2 部グラフ可視化法の高速化を焦点に当てた新しい SE（提案 SE）法を提案する．

本論文の構成は次の通りである．2 章で可視化アルゴリズムについて述べ，3 章で大規模ネットワークデータを用いて提案 SE 法の評価実験をし，実験結果の考察を述べる．最後に 4 章で本研究のまとめを述べる．

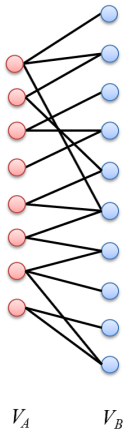

図 1　2 部グラフ

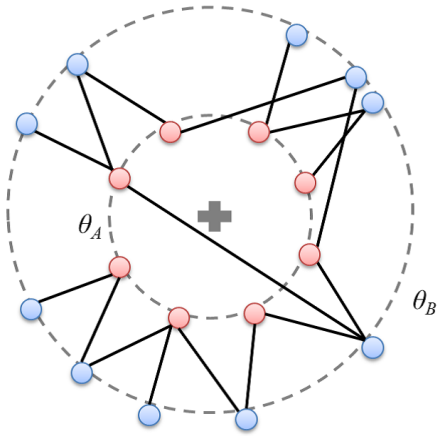

図 2　Spherical Embedding

2.　2 部グラフ可視化法

ノード集合を V，リンク集合を E とし，ネットワーク G は $G := (V, E)$ と定義する．ノード集合 V が以下を満たす二つの集合 $V = V_A \cup V_B$ に分割できるとき，ネットワーク G を 2 部グラフと呼ぶ．

- $\forall u, v \in V_A,\ \{u, v\} \notin E$
- $\forall w, x \in V_B,\ \{w, x\} \notin E$

$|V_A| = M, |V_B| = N$ とする．本論文では，無向の 2 部グラフを考える．2 部グラフの可視化法として既存 SE 法と提案 SE 法を説明する．

2.1　既存 SE 法

Spherical Embedding (既存 SE) 法は，2 部グラフのために設計された可視化法である．2 つの部分集合 V_A と V_B は，2 次元ユークリッド空間内の 2 同心円上にそれぞれのノードを配置して表される．部分集合 V_A は内側の円（θ_A）で描かれ，部分集合 V_B は外側の円（θ_B）で描かれる．θ_A は半径 $r_A = 1$，θ_B は半径 $r_B = 2$ としている．この可視化手法は，同じカテゴリに属する（似ている）ノード同士を近くに，異なるカテゴリに属する（似ていない）ノードを遠くに配置するという方法になっている．図 2 は図 1 を既存 SE 法を用いてた 2 次元空間での可視化を示している．

既存 SE 法はリンク集合 E のすべてのリンクに対するノード間の距離を最小にすることに帰着される．すなわち，ノード i の座標ベクトルを $\mathbf{z}_i$ とし，$\mathbf{z}_i^T \mathbf{z}_i = r_i^2$ という制約のもとで以下の目的関数を最小化するようなノード座標行列 $\mathbf{Z} = \{\mathbf{z}_i\}$ を決定することになる．

$$J(\mathbf{Z}) = \frac{1}{2} \sum_{i=1}^{M+N-1} \sum_{j=i+1}^{M+N} w_{ij} (c_{ij} r_i r_j - \mathbf{z}_i^T \mathbf{z}_j)^2 \qquad (1)$$

ここで，ノード i とノード j 間にリンクが存在すれば $c_{ij} = +1$ とし，それ以外では -1 とする．そして，ノード $i \in V_A$ に対しては $r_i = r_A$ でノード $j \in V_B$ に対しては $r_j = r_B$ である．w_{ij} は隣接するノードペアを強調する任意の重みとする．式 (1) をノード i とノード j のなす角度を用いて記すと，

$$J(\mathbf{Z}) = \frac{1}{2} \sum_{i=1}^{M+N-1} \sum_{j=i+1}^{M+N} w_{ij} r_i r_j (c_{ij} - \cos\theta_{ij})^2 \qquad (2)$$

となる．ここで，$\mathbf{z}_i^T \mathbf{z}_j = \cos\theta_{ij} \|\mathbf{z}_i\| \|\mathbf{z}_j\| = \cos\theta_{ij} r_i r_j$ である．式 (2) より，隣接するノード同士は原点からの角度が小さいときに，$\cos\theta$ が小さくなり，各項は最小化されることがわかる．式 (2) を最小化するノードの配置により可視化する．

2.2　提案 SE 法

提案 SE 法では，2 部グラフの隣接行列 $\mathbf{A} = \{a_{ij}\}$ に対して中心化を施す [4]．隣接行列 $\mathbf{A}$ は，$M \times N$ の矩形行列であり，$\forall i \in V_A, \forall j \in V_B$ に対して $\{i, j\} \in E$ なら $a_{i,j} = 1$ でその他は $a_{i,j} = 0$ である．中心化行列 $\mathbf{H}_M = \mathbf{I}_M - \frac{1}{M}\mathbf{1}\mathbf{1}^T$ および $\mathbf{H}_N = \mathbf{I}_N - \frac{1}{N}\mathbf{1}\mathbf{1}^T$ と定義すると，左右より中心化された隣接行列は $\mathbf{B} = \mathbf{H}_M \mathbf{A} \mathbf{H}_N$ となる．ここで，$\mathbf{I}_N$ は，$N \times N$ の単位行列であり，$\mathbf{1}^T = (1, \cdots, 1)$ である．

提案 SE 法では，行列 $\mathbf{B}$ の要素を $\{b_{ij}\}$ とし，以下の目的関数を最大化するようにノード座標行列 $\mathbf{X} = [\mathbf{x}_1, \cdots, \mathbf{x}_M]^T$ および $\mathbf{Y} = [\mathbf{y}_1, \cdots, \mathbf{y}_N]^T$ を求める．

$$J(\mathbf{X}, \mathbf{Y}) = \sum_{m=1}^{M} \sum_{n=1}^{N} b_{m,n} \frac{\mathbf{x}_m^T}{r_A} \frac{\mathbf{y}_n}{r_B} + \frac{1}{2} \sum_{m=1}^{M} \lambda_m (r_A^2 - \mathbf{x}_m^T \mathbf{x}_m) + \frac{1}{2} \sum_{n=1}^{N} \mu_n (r_B^2 - \mathbf{y}_n^T \mathbf{y}_n). \qquad (3)$$

式 (3) において $\frac{\mathbf{x}_m^T}{r_A} \frac{\mathbf{y}_n}{r_B} = \cos\theta_{mn}$ であり，隣接するノード同士が原点から見て同じ方向に配置されることによって，$\cos\theta_{mn}$ は最大化される．また，式中の λ_m および μ_n はノードが円周上に配置されるための制約に対するラグランジュ乗数である．目的関数を最大化するようなノードの配置を求めるため，ベクトル $\mathbf{x}_m$ で目的関数を微分すると以下を得る．

$$\frac{\partial J(\mathbf{X}, \mathbf{Y})}{\partial \mathbf{x}_m} = \frac{1}{r_A r_B} \sum_{n=1}^{N} b_{m,n} \mathbf{y}_n - \lambda_m \mathbf{x}_m. \qquad (4)$$

よって，ベクトル群 $\mathbf{Y}$ を固定すれば，ベクトル $\mathbf{x}_m$ の最適配置は以下のように求まる．

$$\mathbf{x}_m = \frac{r_A}{\|\tilde{\mathbf{x}}_m\|}\tilde{\mathbf{x}}_m, \quad \tilde{\mathbf{x}}_m = \sum_{n=1}^{N} b_{m,n}\mathbf{y}_n \tag{5}$$

ここで，$r_A/\|\tilde{\mathbf{x}}_1\|, \cdots, r_A/\|\tilde{\mathbf{x}}_M\|$ が成分の対角行列を $\mathbf{\Lambda}_M$ とし，$m = 1, \cdots, M$ で式 (5) を行列表記すれば以下となる．

$$\mathbf{X} = \mathbf{\Lambda}_M \mathbf{B}\mathbf{Y} = \mathbf{\Lambda}_M \mathbf{H}_M \mathbf{A} \mathbf{H}_N \mathbf{Y}. \tag{6}$$

したがって，ベクトル群 $\mathbf{Y}$ を中心化し，それらに隣接行列を乗じ，さらに中心化を施した結果に対し，半径 r_A の球面上に配置されるように正規化すれば，ベクトル群 $\mathbf{Y}$ を固定したときの最適な配置 $\mathbf{X}$ を得る．

同様に，ベクトル群 $\mathbf{X}$ を固定すれば，ベクトル $\mathbf{y}_n$ の最適配置は以下のように求まる．

$$\mathbf{y}_n = \frac{r_B}{\|\tilde{\mathbf{y}}_n\|}\tilde{\mathbf{y}}_n, \quad \tilde{\mathbf{y}}_n = \sum_{m=1}^{M} b_{m,n}\mathbf{x}_m \tag{7}$$

よって，$r_B/\|\tilde{\mathbf{y}}_1\|, \cdots, r_B/\|\tilde{\mathbf{y}}_N\|$ が成分の対角行列を $\mathbf{\Lambda}_N$ とすれば以下を得る．

$$\mathbf{Y} = \mathbf{\Lambda}_N \mathbf{B}^T \mathbf{X} = \mathbf{\Lambda}_N \mathbf{H}_N \mathbf{A}^T \mathbf{H}_M \mathbf{X}. \tag{8}$$

したがって，提案 SE 法のアルゴリズムは以下となる．

1. ベクトル群 $\mathbf{X}$ と $\mathbf{Y}$ を初期化する．
2. ベクトル群 $\mathbf{Y}$ を固定し，式 (6) で $\mathbf{X}$ を求める．
3. ベクトル群 $\mathbf{X}$ を固定し，式 (8) で $\mathbf{Y}$ を求める．
4. 目的関数 $J(\mathbf{X}, \mathbf{Y})$ の変化が十分小さければ終了する．
5. ステップ 2. へ戻る．

明らかに，提案 SE 法のアルゴリズムは HITS アルゴリズム [2] と類似した構造を持つことが分かる．ただし，ベクトル群に対して 2 重の中心化を施す点，および，正規化の施し方の点に特徴を持つ．提案アルゴリズムの 1 反復は，2 部グラフのリンク数に比例した計算量となる．よって，ネットワーク可視化の代表手法の一つバネモデル法 [1] と同様な非線形最適が必要な既存 SE 法と比較して，提案 SE 法を適用すれば大幅な高速化が期待できる．

3. 評価実験

既存 SE 法と提案 SE 法での可視化結果と実行時間を比較する．実験データとして，Yahoo！映画と国立情報学研究所を通じて提供されている Yahoo!知恵袋のそれぞれから生成した 2 部グラフを使用する．

3.1 実験データ

Yahoo!映画では映画のカテゴリを V_A，映画を V_B とし，カテゴリと映画の関係を既存 SE 法と提案 SE 法で可視化，分析する．また，年代ごとのカテゴリと映画の関係にどのような変化があるのかを分析するため，映画を年代別に分けたデータを用いる．カテゴリ数は 16 あり (図 5)，年代別の映画数，リンク数を表 1 に示す．

Yahoo！知恵袋では，約 300 のカテゴリが設けられ，すべての投稿は 1 つのカテゴリに属する．分析対象データのカテゴリ，質問数，回答者数およびリンク数を表 2 に示す．なお，このネットワークは各質問に対するベストアンサー（BA）から生成した 2 部グラフであり，BA に選ばれた回答者を V_A，質問を V_B とし，評価に用いる．各質問には一人の BA しか存在しえないため，リンク数は質問数に一致する．また，表 2 のノード数は質問数と回答者数を合わせたものである．本研究では，実行速度の比較に焦点をあてるため，最もシンプルなネットワークとしている．表 2 の数値は，提供されているデータの全体にあたる 2004 年 4 月から 2005 年 10 月までの期間における投稿数が多かった上位 4 カテゴリである．投稿数において上位となる「恋愛相談，人間関係の悩み」,「パソコン，周辺機器」,「政治，社会問題」,「Yahoo! オークション」,（以下,「恋愛」,「パソコン」,「政治」,「オークション」と記す）について，カテゴリ別で分析する．

表 1　Yahoo！映画

年代	1950	1960	1970	1980	1990	2000-04	2005-09
映画数	594	1079	1314	1805	2659	2948	3264
リンク数	899	1617	2071	2994	4424	6057	6564

表 2　Yahoo！知恵袋

質問カテゴリ	恋愛	パソコン	政治	オークション
質問数	210105	171848	78777	190432
回答者数	62717	27420	25766	37727
ノード数	272822	199268	10543	228159
リンク数	210105	171848	78777	190432

3.2 実験結果と考察

実行時間に関する分析結果を図 3，図 4 に示す．図 3，図 4 は Yahoo！映画と Yahoo！知恵袋の分析結果であり，それぞれ平均実行時間と平均実行時間から 1 標準偏差の値を示している．横軸はノード数，縦軸は実行時間となっており，既存 SE 法と提案 SE 法を 100 回実行した平均と標準偏差をプロットしてある．図 3 と図 4 を見ると，Yahoo！映画と Yahoo！知恵袋ともにノード数が増えるにつれ平均実行時間は増えているが，それぞれ既存 SE 法と比較すると，平均実行時間は提案 SE 法の方が高速であることが分かる．図 3 の既存 SE 法の 2000-04 平均実行時間が 2005-09 平均実行時間よりやや大きいが，これはネットワーク構造の影響だと考えられる．また，図 4 の既存 SE 法によるオークション平均実行時間がパソコン，政治に比べると，やや平均実行時間がノード数に関係なく下回っていることが見て取れる．これは，BA に選ばれた回答者の平均次数がパソコン，政治より少ないことが起因していると考えられる．提案 SE 法の標準偏差は既存 SE 法の標準偏差より小さく，提案 SE 法の方が安定していると言える．

可視化結果は図 6〜11 に示し，各カテゴリの色は図 5 に示す．提案 SE 法での可視化結果では，同じようなカテゴリ群に属しているノードは近くに配置される．また，同じようなノードに属されているカテゴリは近くに配置される．これは既存 SE 法

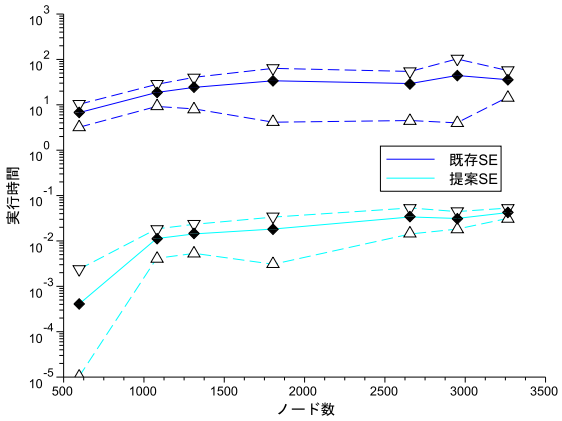

図 3　Yahoo!映画

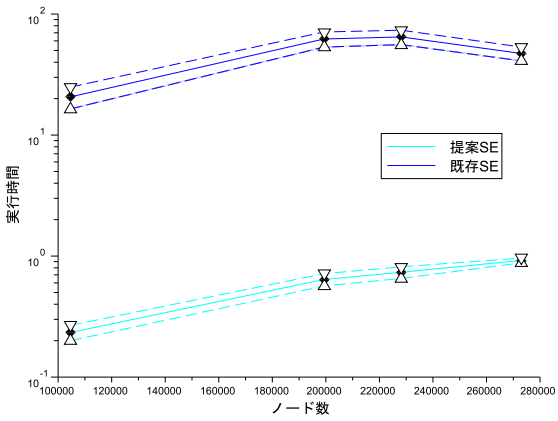

図 4　Yahoo!知恵袋

の可視化結果にも同様に見られ，提案 SE 法と同じような配置になっている．具体的には，提案 SE 法の 1950 年代の青春カテゴリはラブストーリー，ドラマカテゴリに挟まれている．この結果は既存 SE 法でも同様に認識できる．提案 SE 法では，カテゴリへの属し方が同じノードは重なってプロットされる．従って，映画とカテゴリ間のリンクも重なり，既存 SE 法より可視化結果がすっきりと見える．

それぞれの年代でカテゴリを比較してみると，すべての年代で近くに配置されているのもあれば，年代が変わると離れるのもある．例えば，SF・ファンタジーカテゴリとホラーカテゴリはすべての年代で近くに配置されている．また，1950 年代のアクション・アドベンチャーとドキュメンタリーが近くに配置されているが，1980 年代では離れて配置されている．

4. おわりに

本論文では，既存 SE 法とは異なるアルゴリズムの提案 SE 法を提案し，大規模 2 部グラフデータを用いて評価実験を行い，既存 SE 法と提案 SE 法の比較をした．結果として，提案 SE 法は実行時間は高速化され，また既存 SE 法と同じような配置が得られた．これらより，提案 SE 法は大規模なデータでもより高速にノードの適切な配置を決定できる点で他の分野への応用が期待できることが示唆された．

今後は，階層性のある Web 上のハイパーリンク構造へ応用などのために，提案 SE 法のさらなる拡張を視野に入れ，提案 SE 法の有効性を確かめていきたい．

謝辞　本研究は，科学研究費補助金基盤研究 (C) (No. 22500133) の補助を受けた．

1 SF･ファンタジー	red
2 アクション･アドベンチャー	black
3 アニメーション	green
4 コメディー	blue
5 サスペンス	maroon
6 青春	orange
7 西部劇	purple
8 戦争	navy
9 ドキュメンタリー	olive
10 ドラマ	lime
11 ファミリー	
12 ホラー	
13 ミュージカル	magenta
14 ラブストーリー	aqua
15 特撮	yellow
16 その他	gray

図 5　映画カテゴリ

文　献

[1] T. Kamada and S. Kawai, “An algorithm for drawing general undirected graph”, Information Processing Letters, 31, 1989.

[2] J. Kleinberg, “Authoritative sources in a hyperlinked environment”, Proc. of 9th ACM-SIAM Symposium on Discrete Algorithms, 1998.

[3] A. Naud, S. Usui, N. Ueda, and T. Taniguchi, “Visualization of documents and concepts in neuroinformatics with the 3D-SE viewer”, Proc. of Frontiers in Neuroscience (Frontiers in Neuroinformatics), 2007.

[4] W. Torgerson, “Theory and methods of scaling”, Wiley New York, 1958.

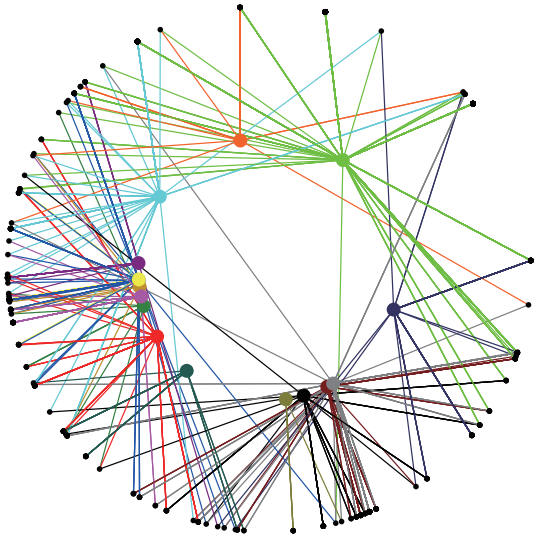

図 6　既存 SE 法 1950-1959

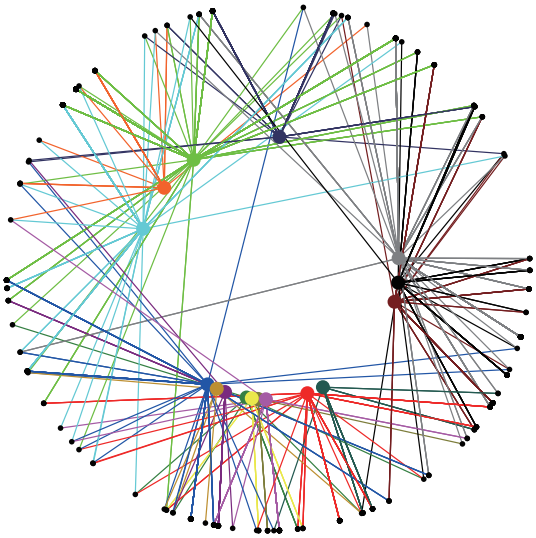

図 9　提案 SE 法 1950-1959

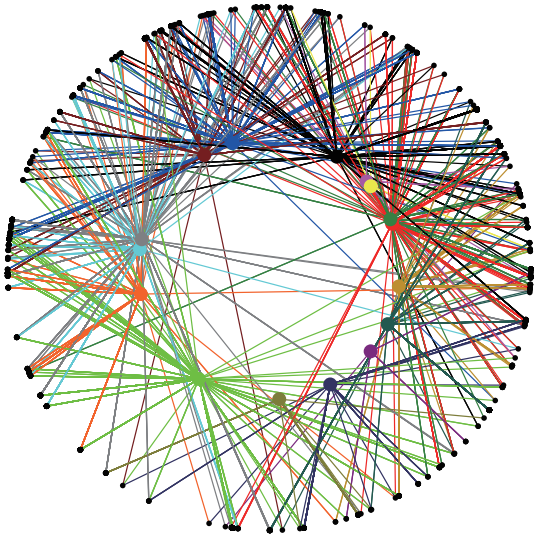

図 7　既存 SE 法 1980-1989

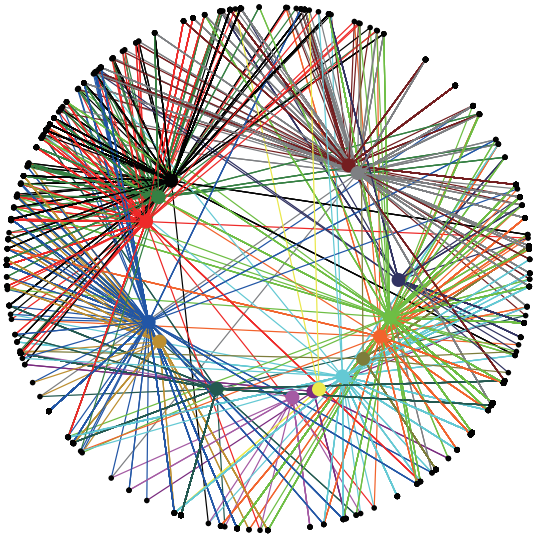

図 10　提案 SE 法 1980-1989

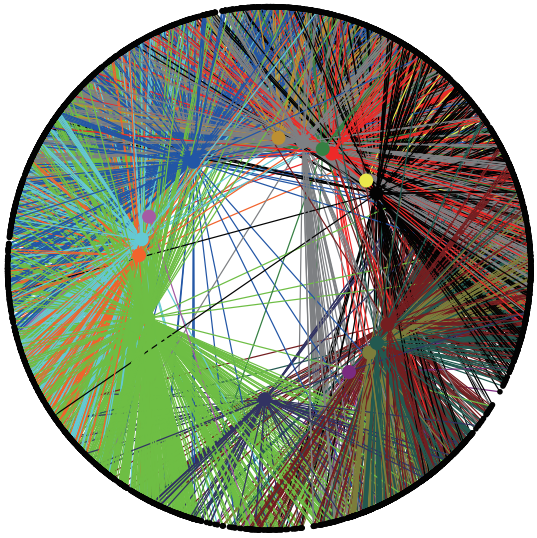

図 8　既存 SE 法 2005-2009

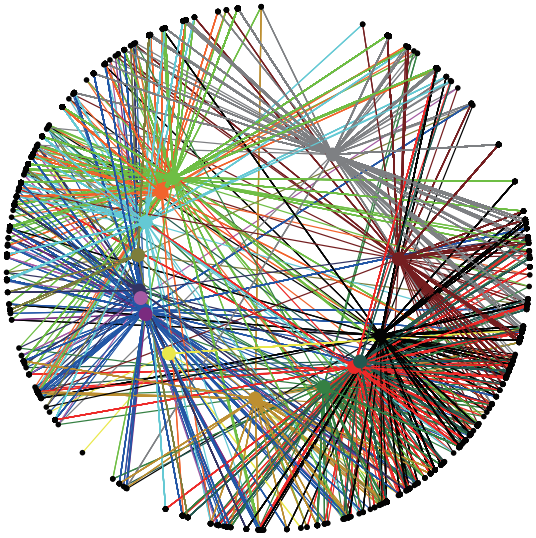

図 11　提案 SE 法 2005-2009