

階層型グラフデータのための可視化手法

土井 淳*, 伊藤 貴之*, 梶永 泰正*, 池端 裕子*

*日本アイ・ビー・エム（株）東京基礎研究所
神奈川県大和市下鶴間 1623-14

doichan@jp.ibm.com, itot@computer.org, kajinaga@jp.ibm.com, ikehata@jp.ibm.com

Layouting and Graphics Techniques for Visualization of Hierarchical Graph Data

Jun Doi, Takayuki Itoh, Yasumasa Kajinaga, Yuko Ikehata
IBM Research, Tokyo Research Laboratory
1623-14 Shimotsuruma Yamato-shi, Kanagawa-ken

1. はじめに

本研究では、ウェブサイト、計算機ネットワークシステム、テキストや画像、遺伝子情報などのデータベース、金融・交通・社会組織などのネットワーク、などのようにノードと関連性をもったリンク構造からなるグラフデータを可視化するための技術を提案する。

これらのグラフ構造を持ったデータは情報量が数千、数万と規模が大きく、また日々変化したり情報量が増えつづけていたりする性質のあるものもある。このようなデータをユーザにとって意味のある形で、限られたディスプレイ空間上に可視化する技術は大変有用である。

大規模なグラフデータを限られたディスプレイ空間に表示する場合、誤読を防ぐため、あるいは処理を効率よく行うために、グラフデータをいくつかのグループに分けて適応的に可視化処理を行う。また、グループを入れ子構造にし、グループが子グループを持つような階層構造をグラフデータに持たせる。

本稿では、階層構造を持ったグラフデータをディスプレイ空間にレイアウトするための手法と、階層構造を持ったグラフデータをインタラクティブに可視化するための手法を述べる。

2. 階層型グラフデータ

あるグラフデータの中からノードを取り出してグループを作り、グループをノードとしたグラフデータを作る。これを再帰的に繰り返して作成されたものが階層型グラ

フデータである。

たとえば、ウェブページ間のリンク構造を表現したグラフをディレクトリ構造で階層化したデータ、階層構造をもつ人事組織間の交流をグラフで表現したデータ、ある種のパラメータなどによってグループ分けされた関連付きデータベースのデータなどがあげられる。

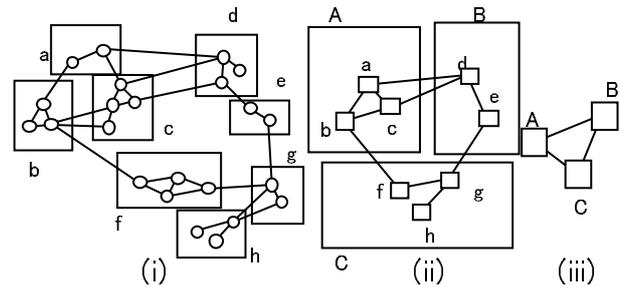


図 1 階層型グラフデータ

本稿が想定している階層型グラフデータの例を図 1 に示す。この例では、図 1(i)のようなグラフデータから、四角で囲ったノードをグループとして(ii)のようなグラフを作る。このような上位階層のグラフを親グラフと呼び、グループ化された下位階層のグラフを子グラフと呼ぶ。このように階層化されていないノードのグループ化処理を繰り返していくことで、最終的に(iii)のような最上位グラフが得られる。

本稿ではグラフデータのグループ化/階層化の手法には触れず、可視化される入力グラフデータは、別の手法によって既に階層構造を持っているものとする。通常、意味のあるグループ化/階層化が行われており、同じような属性を持ったノードや関連性の強いノードは同じグループ

に属し、可視化するときに近い位置に表示することが可能である。

階層型データを表現する手法は、Treemaps [Johnson91], Cone Tree [Carriere95], Hyperbolic Tree [Lamping96] など、従来から多く知られている。これらの手法が階層型データの階層関係(上下関係)のみを表現するのに対して、本手法は階層関係だけでなく、同一階層間のノードの接続関係を同時に表現するという点で根本的に異なる。また、大規模なグラフデータを高速描画しやすいように階層化する手法も提案されている [Quigley00] が、本手法は階層構造がすでに与えられているという点において目的意識が異なる。

3. 階層型グラフデータのレイアウト

ウェブサイトのグラフデータのように、位置情報を持っていないグラフデータを可視化するには、グラフを構成する各ノードの座標値を決定する必要がある。

本研究では、ノード同士の重なりを避け誤読の少ない可視化のために、力学モデルを用いたグラフデータの画面配置手法 [itoh01] を用いる。

3.1. 力学モデルによるグラフのレイアウト

本手法では、図2のように、アークによって連結されたグラフノード間の引力と、アークによって連結されていない近接するノード間の斥力を力学モデルとして定義し、グラフデータ全体でこれらの力のつりあいをシミュレートすることでノードの位置を決定する。

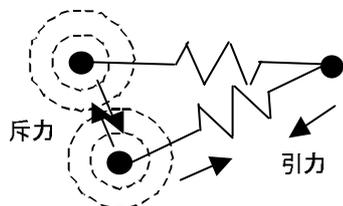


図2 ノードレイアウトのための力学モデル

文献 [itoh01] では、グラフデータを読みやすく配置するための条件として、図3のような条件を設定している。図2に示したような力学モデルベースの手法は、図3に示した条件を合理的な計算時間で実現する有用な手法であ

る。文献 [itoh01] は、力学モデルを用いたグラフ配置手法の計算時間と適応性の改善に貢献している。

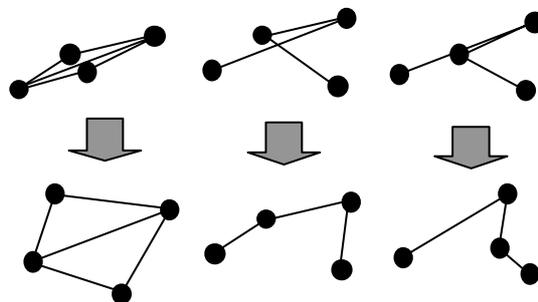


図3 グラフデータの画面配置問題の典型的な条件。(左)近隣ノードが適切な距離を保つ。(中)アークの長さの総和を低減する。(右)アークの端点以外の場所で別のノードが重なるのを避ける。

本稿では、力学モデルによるグラフデータの配置手法を、階層型のグラフデータの配置に拡張する。

3.2. 階層型グラフデータのレイアウト

階層型グラフデータは、再帰的なグラフデータであるので、階層型グラフデータからある階層のグラフデータを取り出し、力学モデルによる配置手法でノードの配置を行うことが可能である。ここでは、効率よく階層型グラフ全体の配置を行う手法として次のような処理を行う。

- (1) 階層型グラフの上位階層からレイアウトを行う
- (2) 子グラフをレイアウトし占有領域を上位階層へフィードバック
- (3) 親グラフの再配置
- (4) (1)~(3)を下位階層へ再帰的に繰り返す

まず、図4(i)のように最上位階層のグラフABCを力学モデルによる配置手法を用いてレイアウトする。続いてABCそれぞれのノードの子グラフをレイアウトする。図4(ii)では、ノードCの子グラフfghがレイアウトされている。このとき、レイアウトされた子グラフfghの占有領域を親ノードCにフィードバックする。このとき図4(iii)のようにノードの

サイズに反映され、ノードのサイズを考慮してグラフABCを再びレイアウトする。

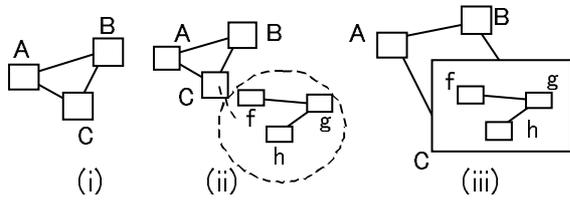


図4 階層型グラフデータのレイアウト

このような処理を行うことで、下位の階層のノード同士や別の階層のノード同士が干渉することなく階層型グラフデータをレイアウトすることができる。このとき、文献[ito01]に示すグラフ配置アルゴリズムを適用することで、図4(iii)に示す上位階層の再配置の計算を局所化し、計算時間の増加を抑えることができる。

また、階層型グラフデータの上位階層グラフから適応的にレイアウトし可視化できるので注目したい部分だけをインタラクティブに選択してレイアウトさせながら可視化することが可能である。図7に実際のグラフデータでの実行例を示す。

4. 階層型グラフデータの可視化手法

レイアウトされた階層型グラフデータを、ユーザによりわかりやすく可視化するための手法について述べる。レイアウトされたグラフデータは2次元的な位置情報を持っており、互いのノードが重ならないように可視化することができるように配置されている。

しかし、階層構造を理解したり、特定のグループに注目したりするには不十分である。そこで、①階層構造を理解するための階層型グラフデータの透視投影手法および、②特定のサブグラフを注目するために空間を膨張させて階層型グラフデータを表示する手法を提案する。

4.1. 階層型グラフデータの透視投影

階層型グラフデータの階層の深さを奥行き方向に割り当てて透視投影を行う。階層の深い子グラフは視点から遠くに見える、別の階層のノードとの区別が付きやすくなる。また、選択したノードの子グラフへズームインすることで

それよりも上の階層のノードを表示しないようにすることができたり、逆に特定の子グラフだけを手前に飛び出すように表示させたりすることも可能である。

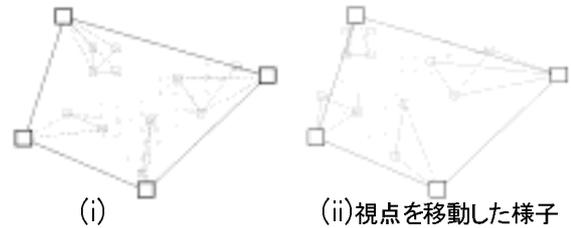


図5 透視投影による階層型グラフデータの表示

4.2. 空間膨張による階層型グラフデータ表示手法

階層型グラフデータの子グラフを展開して表示するとき、子グラフの占有する部分のディスプレイ空間を膨張させることにより、ユーザの注視する子グラフをズームアップさせて、画面にレイアウトする。同じような手法にFisheye[Sarkar92]がある。

空間の膨張処理は、次のように既にディスプレイ空間上に配置されているノードを移動させる処理である。

- (a) ノードが子グラフの占有矩形内にある場合、膨張の中心からの距離と同じだけ中心から離れる方向に移動する。
- (b) 矩形外の場合、中心からの距離の2乗に反比例する距離だけ中心から離れる方向に移動する。

図6(i)の中心に子グラフを持つノードがあるとき、図6(ii)のように空間を膨張させてできた隙間の部分に子グラフがレイアウトされる。

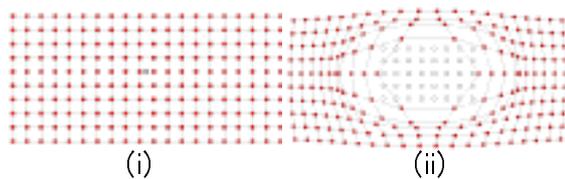


図6 空間膨張処理

空間膨張処理によって、グラフ全体の位置関係を大きく変えずに、注目したい部分だけを表示するような効果を得

られる。このとき、ユーザの選択した子グラフをインタラクティブに可視化することができる。

また、適応的な階層型グラフデータのレイアウト処理において占有領域を上位階層にフィードバックするときに再配置処理を行う代わりに空間膨張処理を行うことでノード同士の干渉を回避できる。大規模な階層型グラフデータのレイアウトを行う場合により高速にレイアウト処理を行うことが可能である。

空間膨張による階層型グラフデータのレイアウトの例を図8に示す。

5. まとめ

階層型グラフデータのレイアウト処理について、階層ごとに力学モデルを用いてレイアウトし上位階層に占有領域をフィードバックする処理を導入することにより、大規模な階層型グラフデータのレイアウト処理の効率化と、適応的なレイアウト処理を実現した。

また、階層型グラフデータの表示手法に透視投影と空間膨張処理を組み合わせることにより、階層構造を理解しやすく、また、グラフの形状を大きく変えることなく一部を注目して表示できるようにした。また、インタラクティブに、階層型グラフデータのレイアウト処理と可視化処理を行うことができるようにした。

6. 参考文献

[Johnson91] B. Johnson, et al., Tree-Maps: A Space-Filing Approach to the Visualization of Hierarchical Information Space, IEEE Visualization '91, pp. 275-282,1991.

[Carriere95] J. Carriere, et al., Research Report: Interacting with Huge Hierarchies Beyond Cone Trees, IEEE Information Visualization '95, pp. 74-81,1995.

[Lamping96] J.Lamping, et al., The Hyperbolic Browser: A Focus+context Technique for Visualizing Large Hierarchies", J. Visual Languages and Computing, 7, 1, pp.33-55, 1996.

[Quigley00] A. Quigley, et al., FADE: Graph drawing, clustering, and visual abstraction, Graph Drawing 2000.

[itoh01]伊藤貴之 他, 力学モデルを用いたグラフデータの画面配置手法の改良,2001-CG-103, 情報処理学会グラフィックスとCAD研究会

[Sarkar92] Manojit Sarkar, Marc H. Brown, "Graphical fisheye views of graphs", In Proceedings of ACM SIGCHI 92 Conference on Human Factors in Computing Systems, pp. 83-91.

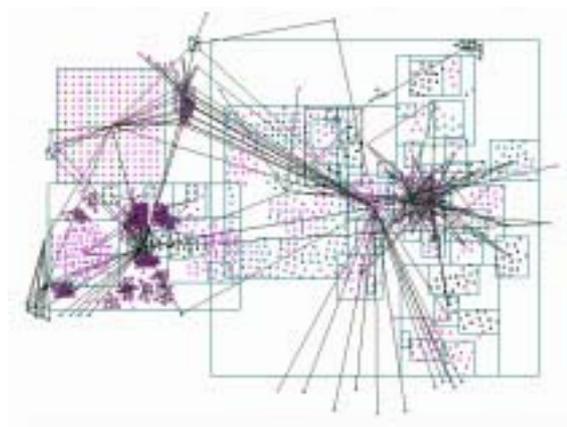


図7 ウェブサイトのグラフデータのレイアウト結果

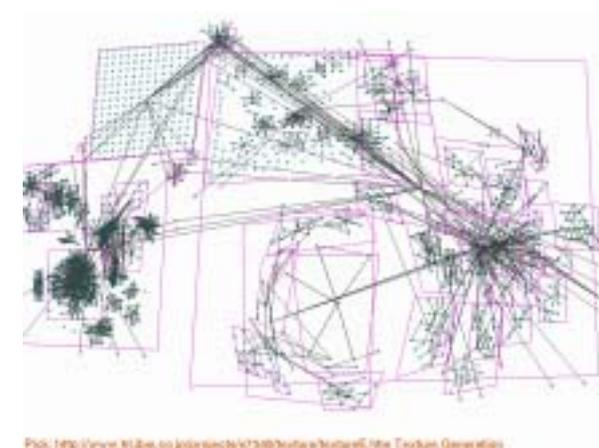


図8 空間膨張処理による子グラフのレイアウト