

論 文

グラフ描画アルゴリズムへの多視点遠近画法の導入

南雲 淳[†] 田中 二郎^{††}

Introducing Fisheye-view into graph drawing algorithm

Jun NAGUMO[†] and Jiro TANAKA^{††}

あらまし 本論文では、無向グラフ自動描画アルゴリズム、特に、力指向再配置アルゴリズムに対して多視点遠近画法を導入するためのアルゴリズムの改良を提案する。

節の位置の情報を用い、節の間を結ぶ辺の長さを変化させることによって注目点付近の拡大と注目点から遠い部分の縮小を表現するために、従来の力指向再配置アルゴリズムへの組み込みが容易に行うことができた。また、力指向アルゴリズムを使うことによって得られる「反復計算の過程を順次表示することでグラフの変化をスムーズに提示できる」という特長を生かしたまで多視点遠近画法を実現できた。

キーワード グラフ描画アルゴリズム、力指向配置、多視点遠近画法

1. はじめに

多数の情報やそれら同士の関係を記述するために使われるものとしてグラフ構造がある。グラフ構造は一般に、節とそれらの関係を表わす辺によって論理的に表現されるが、グラフの論理的表現に対し、人間にとてわかりやすい形でグラフを画面に表示するためには必要とされる視覚的情報（主として、節の座標情報）を与えるものとしてグラフ自動描画アルゴリズムが研究されている[1]。

従来、グラフ描画アルゴリズムの研究においては、大規模グラフを見やすくするための研究は多くはないなかつたが、本研究では、節や辺の多い規模の大きなグラフの可読性向上・一覧性と詳細性の両立を目的として、単純無向グラフを対象としたグラフ自動描画アルゴリズム、特に、力指向の配置アルゴリズムに多視点遠近画法を導入する手法の提案を行う。

ここで提案する手法の導入により、節や辺の多い大規模なグラフの一覧性と詳細性の両立が図れる。

また、力指向アルゴリズムには「グラフの変化をスムーズに提示しやすい」という、インタラクティブな

システムに利用する上で重要な長所があるが[2]、その特長を生かしたまま多視点遠近画法を導入できるため、ユーザの画面上のグラフに対するインタラクティブ性を向上させることができる。

2. 対象とするグラフと基にするアルゴリズム

今回は、グラフ構造の中でも特に単純無向グラフを対象とする。単純無向グラフとは、ある2つ節の間を結ぶ辺がたかだか一本であり、辺に向きが定められていないグラフである。

単純無向グラフ描画アルゴリズムとしては、節を物体、辺をバネとみなした力学系の釣合いを考えることによって節の配置を求める力指向アルゴリズムがあり、Eadesスプリングモデル[3]やそれを改良したFruchtermanのアルゴリズム[4]、Kamada-Kawaiのアルゴリズム[5]などが知られている。今回は計算量やインタラクティブ性の問題を考慮して、その中でも基本的なものであるEadesスプリングモデルを改良のベースとした。

Eadesスプリングモデルでは、辺によって接続された節同士の間のバネの力 f_s と直接接続されない節同士の間に働く斥力 f_r を、節の間の距離 d 、辺の理想長 d_0 （表示領域の広さやグラフの規模に応じて決める）、定数 C_s, C_r を用いて以下のように定義する。

[†] 筑波大学大学院理工学研究科

Master's program in Science and Engineering Univ. of Tsukuba, Ibaraki, 305-8573 Japan

^{††} 筑波大学電子・情報工学系

Institute of Information Sciences and Electronics Univ. of Tsukuba, Ibaraki, 305-8573 Japan

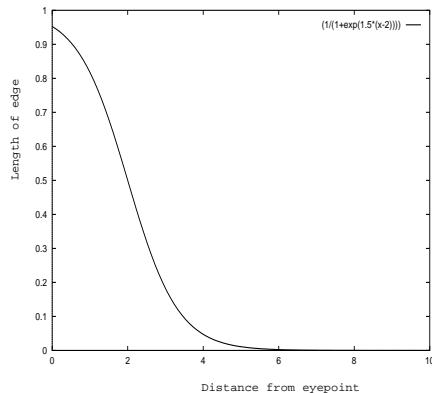


図 1 関数 d_0
Fig. 1 Function d_0

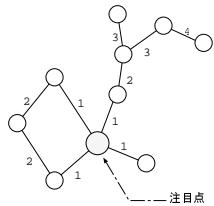


図 2 パラメータ l の考え方 (注目節が 1 つのとき)
Fig. 2 Counting parameter l (Single eyepoint case)

$$f_s(d) = C_s \log \frac{d}{d_0} \quad (1)$$

$$f_r(d) = C_r \frac{1}{d^2}$$

辺の理想長 d_0 が定数であるために、このアルゴリズムは辺の長さをできるだけ一定にしようとする。

「ある 1 つの節と他全ての節の間で上の式に従って力 \vec{f}_s および \vec{f}_r を計算し、これらのベクトルの和に比例した距離だけその節を動かす」ということを全ての節に対して行う。これを何度も反復することによって節を適切な位置に収束させていくのが Eades スプリングモデルである。

今回、式 (1) に含まれる辺の理想長 d_0 を注目節から当該辺までの距離 (間にいくつの辺があるかによって決まる論理的距離) の関数として多視点遠近画法を実現する。詳しい手法については次で述べる。

3. 多視点遠近画法の導入

対象とするグラフ構造の規模が大きくなると、その

グラフを配置した結果も複雑なものとなり、わかりにくい表示となってしまう。また、「全体を概観しながら部分的な詳細も見たい」というような要求も出てくる。また、詳細を見たいという部分は一つであるとは限らず、複数の部分を注目したいということも起こりうる。

このために、全体を概観できる一覧性と（複数の）部分的な詳細を見ることができる詳細性を両立させるための多視点遠近画法が研究されている [6]。

これは、元の画像に写像を施してユーザが注目したい部分を拡大し、そうでない部分を縮小することによって一覧性と詳細性の両立を図るものである。

画像に施す写像の種類によってさまざまなものがあるが、今回、グラフの辺の長さを操作することによって多視点遠近画法を実現する手法を提案する。

3.1 辺の長さに基づく多視点遠近画法の実現

多視点遠近画法においては、まず注目点が指定される必要がある。本手法においては、注目点はグラフの節に置かれるものとする（以後、注目節と呼ぶことにする）。

本手法では、注目節に近い辺の長さを長く、遠い辺の長さを短かくすることによって、注目点付近の拡大および注目点から遠い部分の縮小を擬似的に表現する。

注目節に接続する辺を長くすることによってできたスペースにその節の詳細な情報を表示したり、そこにサブグラフを表示するなどすることができる。

3.2 注目節が 1 つの場合

ユーザが注目節として指定した節と再配置のためにこれから移動させようとしている節の間の距離（2 つの節の間にある辺の数の最小値）を l 、注目する節に接続する辺の理想的な長さを d_0 とする。

また、 a と b を、グラフの見えかたを変化させるための定数、具体的には、 a を注目点から遠くなるにつれて辺が短くなる度合（大きくなるほど、遠い部分の辺の長さの現象が「急激に」起こる）、 b を、注目節の周辺において拡大表示される範囲（大きくなるほど、注目節から遠い部分まで拡大される）とする。

これらの値を用いて、再配置の際の「注目節を考慮した理想的辺長」 $d_{0'}$ を

$$d_{0'} = d_0 \left\{ \frac{1}{1 + e^{a(l-b)}} \right\} \quad (2)$$

のように定義する。

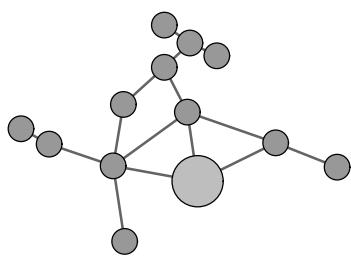


図 3 パラメータ a を変える例：上から $a = 0.8, 1.6$ と変化させている

Fig. 3 Changing parameter a : $a = 0.8, 1.6$

この $d_{0'}$ を式(1)の d_0 に適用することによって、辺の長さを変化させることによる多視点遠近画法が実現できる。

この関数は図1に示す形をしている。グラフ横軸が注目する節から当該辺がどのくらい離れているかを表わす l であり、縦軸がその辺の長さの d_0 に対する割合である。

定数 a を変化させたときの表示の変化を図3、 b を変化させたときの表示の変化を図4に示す。

図3では、注目節に接続する辺の長さはほとんど変わっていないが離れた部分の辺が a を大きくすることによって短くなっている。

図4では、パラメータ b を大きくするに従って「注目節から離れた」部分の辺の長さも長くなっている。 b の操作によって拡大表示の範囲を制御できているのがわかる。

3.3 注目節が複数ある場合

多視点遠近画法においては、注目する点が2つ以上ある場合も考慮されなくてはならない。その場合の本

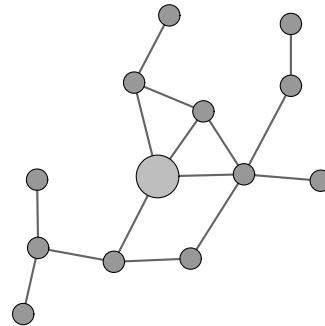
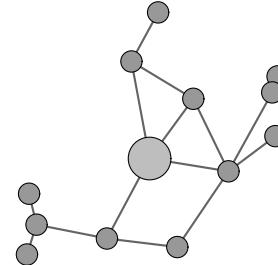
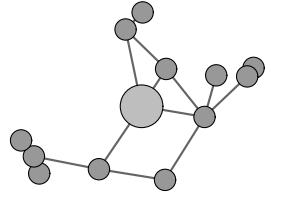


図4 パラメータ b を変える例：上から $b = 1.0, 2.0, 3.0$ と変化させている

Fig. 4 Changing parameter b : $b = 1.0, 2.0, 3.0$

手法の適用について述べる。

ある1本の辺の長さを考えた場合、複数の視点が存在することにより、式2によって計算されるその辺の理想長も視点の数だけ存在することになる。再配置の計算の際にはこれらの辺の理想長の候補の中からどれか1つを選ぶ、あるいは候補に対して演算を行なうなどして辺の理想長を1つに定める必要がある。

ここでは注目節の周りに空間を作ることによって擬似的に節の拡大を実現するのが目的なので、辺の長さの候補の中の最長のものを用いることすれば、注目節が複数ある場合でもそれらの節に直接接続された辺

の長さを長くすることによって擬似的な拡大が実現できる。

このことは、注目節 $1, \dots, n$ のそれぞれについて式(2)で計算した $d_{0'}$ の値を d_n とすれば、その辺に対して使うべき $d_{0'}$ は

$$d_{0'} = \max\{d_1, \dots, d_n\}$$

と表わされる。

あるいは、式(2)における l を、注目節 $1, \dots, n$ のそれぞれから辺への距離を l_1, l_2, \dots, l_n として

$$l = \min\{l_1, l_2, \dots, l_n\}$$

と表わしてもよい。この場合の l の数えかたの例を図5に示す。

この $d_{0'}$ を式(1)の d_0 に適用、あるいは、 l を式(2)に適用することによって注目節が複数ある場合が実現できる。

注目節を複数与えたときの表示の例を図6に示す。図の左側が従来のEadesスプリングモデルによる配置で、右側が注目節を複数与えた例である。

複数の注目節の周囲のみ辺が長くなっている、注目節より遠い部分の辺が短くなっているのがわかる。

以上から、本手法は注目節が複数あるときにも適用可能であると言える。

3.4 画面上のグラフに対するインタラクティブ性
グラフ構造を使うアプリケーションに本手法を使うメリットについて以下で述べる。

画面上のグラフ構造が動的に変化する場合、その変化の様子はスムーズなアニメーション的表現によることが望ましい。そうでないと、「どの節がどこに移動したか」を把握するのが困難になるからである。

力指向アルゴリズムは反復法によって節の配置を求めるため、この計算の過程を順次表示することによって節の位置の変化をスムーズに見せるようにすることが可能である。たとえば、Eadesスプリングモデルの場合、 $f_r(d)$ を以下のように変更する[2]。

$$f_r(d) = \begin{cases} C_{r'} & (d < d_0) \\ C_r \frac{1}{d^2} & (\text{Otherwise}) \end{cases} \quad (3)$$

図7の上段のグラフをアニメーション表現向けに改良したEadesスプリングモデルで再配置すると図7

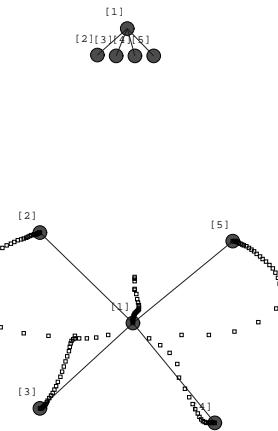


図7 計算過程を順次表示することでスムーズな動きを見せる例：上が初期配置、下が配置終了時
Fig. 7 Example of animated graph transformation

の下段のようになる。下段の図の中に描かれている点線は、各節の移動の軌跡を示している。

このように、反復過程を順次表示することによって節の移動のアニメーション的表現ができれば、従来アニメーション表現に用いられていた「コマ割り」などの特別な処理が必要なくなり、節や辺の移動とそれにに対するユーザのインタラクションの処理を平行に処理できることになり、画面上のグラフに対する節の移動だけでなく追加や削除まで含めたインタラクティブ性の向上に繋がると考えられる。

従来の手法では、まずグラフのレイアウトを行なってからできた画像に写像を施して多視点遠近画法を実現するという手順になるため図の変化をスムーズに見せることや、画面上のグラフに対する直接操作が困難であるが、今回提案した手法を用いれば、アニメーション的表現に向くという特長を生かした今まで多視点遠近画法を組み込むことが可能になる。

これは、本手法がEadesスプリングモデルにおける辺の理想長を変化させるというやり方のため、元々のアルゴリズムの計算によって得られる節の位置に直接影響を与えないからである。

このため、本手法を用いればグラフ構造を表示するアプリケーションのインタラクティブ性を高めながら多視点遠近画法を容易に導入することが可能になる。

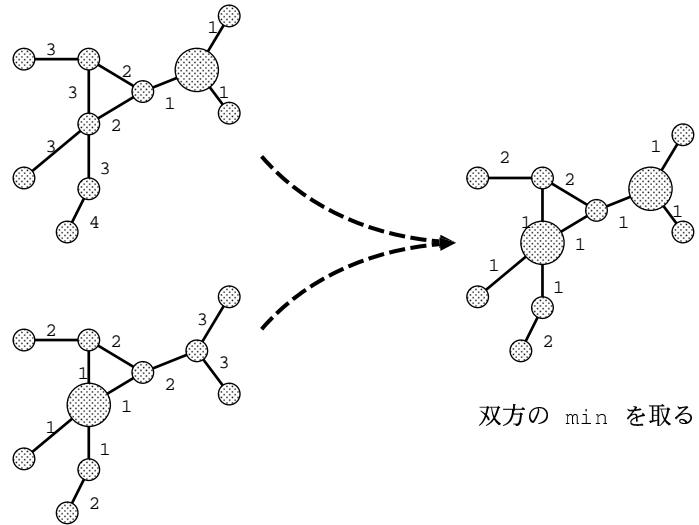


図 5 パラメータ l の数えかた（注目節が複数のとき）
Fig. 5 Counting parameter l (multiple eyepoint case)

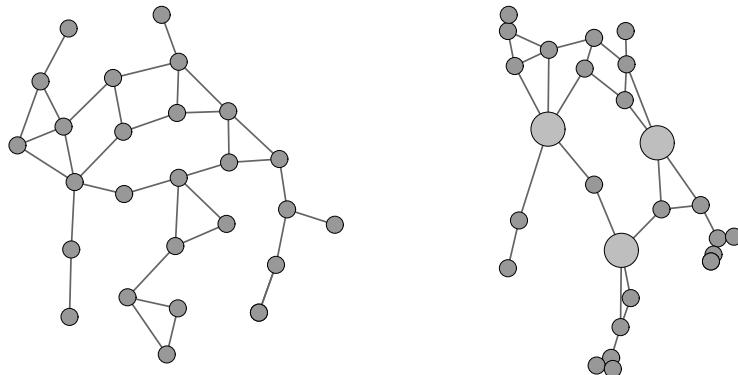


図 6 複数視点による表示例
Fig. 6 Multiple eyepoint example

4. 議論

4.1 計算量とスケーラビリティ

今回、改良のベースとして Eades のアルゴリズムを用いた。力指向のアルゴリズムとしては他に Kamada-Kawai のアルゴリズム [5] が知られているが、計算量を比較すると Eades の $O(|V|^2)$ に対して Kamada-Kawai は $O(|V|^3)$ (V はグラフの節の数) であり、節の数が増えてグラフが複雑になったときのことを考慮すると、Eades の方が有利である。

また、各節の動きかたの面で見ると、Eades は「す

べての節を同時に少しづつ動かして収束させる」のに対して Kamada-Kawai は「ある一つの節を適正と思われる位置に移動してから次に動かすべき節を選ぶ」やり方があるので、全ての節を同時に動かして見せるためには結局一旦計算を完了してから元の配置との間でコマ割りをする必要がある。

なお、グラフの収束性については、どちらのアルゴリズムも最終的に節がごく小さい範囲で振動する場合があるが、計算 1 ステップあたりの節の移動量が一定以上小さくなったら計算を打ち切ることにすれば

実用上充分である。

これらから、アニメーション的な節の移動の表現とインタラクティブ性の向上の両立には力指向アルゴリズムが向いており、その中でも特に Eades のアルゴリズムをベースにするのが良いと言える。

また、グラフを画面表示するアプリケーションにおいて出てくる問題として、グラフのサイズが大きくなつたときにどのように対処するのかというスケーラビリティの問題がある。これに対する対処としては、

- いくつかの節をまとめるモジュール化
- 不要な情報を表示しないようにする省略表示
- 三次元空間を利用する三次元表示

を考えることができる。モジュール化や省略表示については単に節の数が減るだけと考えれば提案手法をそのまま適用することが可能である。三次元表示についても辺の長さが二次元のときと同様に定義できるため、本手法が適用可能である。

また、Eades のアルゴリズムにおいてある節を動かそうとする場合、上述したように計算量は $O(|V|^2)$ となる。これでは、節の数が極めて多くなつたときに問題が生じるが、ある節について、他の節との力を全て計算するのではなく、接続した節、およびある程度近い部分にある節との力を考えることによって計算量を減らすという解決法を考えることができる。この場合、1つの節に接続した辺の数は節の数とは比例しないと考え、節は空間上に平均に分布していると考えれば $O(|V|^2)$ より少ないオーダで計算が可能になる。

以上の議論から、本手法はグラフ構造をインタラクティブに閲覧・操作するためのアルゴリズムとして、グラフの規模が大きくなつた場合まで考慮しても応用が可能であると考えられる。

4.2 応用

我々は、並列論理型言語をグラフ構造を用いて視覚化するビジュアルプログラミングシステムである“PP”[7]を開発している。このシステムでは、並列論理型言語の要素であるアトムや述語や変数といったものをグラフ構造の節や辺にマッピングすることによってプログラムを視覚化し、実行の過程をグラフ構造の変形によって表現する（図8）。

従来はグラフ構造の見やすさやグラフの変化のわかりやすさに問題があつたが、力指向グラフ描画アルゴリズムを利用することによってプログラム実行の表現（グラフの変形の表現）がわかりやすくでき、また、グラフに対するインタラクションがやり易いものに

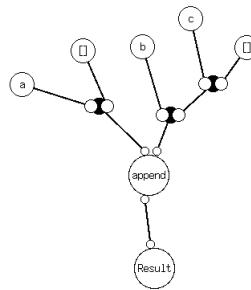


図8 並列論理型言語のプログラムをグラフにマップする例

Fig. 8 Example of mapping parallel logic program to graph structure

なつた[8]。

このような、情報をグラフ構造で表現しそれを画面に描画するアプリケーションにとって、グラフの変形などをスムースに表現でき、また、インタラクティブ性を高めることができるような手法は重要である。

5. 関連研究

グラフ自動描画アルゴリズムに対する多視点遠近画法の導入については三末らによる研究[9]やWangらによる研究[10]、Storeyらによる研究[11]がある。

Wangらによるものは力指向アルゴリズムをベースとしたものであり、節の大きさまでも考慮に入れたものになっているが、パラメータ変更によるグラフ描画の表現の変化が考慮されていないなどの点で本研究と異っている。三末らは、複合グラフを対象としたアルゴリズムを、またStoreyらは、木構造や無向グラフなどに適用可能なより一般的な手法を提案しているが、これらは節の位置に基づくものであり本研究の手法とは異なっている。

6. まとめ

無向グラフ自動描画アルゴリズムに多視点遠近画法を導入する手法として、辺の理想長を注目点からその辺までの距離の関数とすることによって注目節の周囲の拡大と離れた節の縮小を実現する、辺の長さに基づく多視点遠近画法を提案した。

また、画面上のグラフ構造に対するインタラクティビティという点から、力指向アルゴリズムを用いる利点と本手法の導入によるインタラクティビティの向上について述べた。

本手法により、多視点遠近画法までも取り入れた上でグラフ再配置の過程における節の移動をスムースなアニメーション表現として得ることができ、グラフ構造を用いるアプリケーションのインタラクティブ性向上が図れる。

文 献

- [1] 杉山 公造 グラフ自動描画法とその応用 — ビジュアルヒューマン インタフェース – 計測自動制御学会, 1995.
- [2] 南雲淳, 田中二郎, “インタラクティブシステムのためのグラフ描画アルゴリズム,” 情報処理学会 インタラクション'98, pp. 41–48, 1998.
- [3] P. Eades, “A heuristics for graph drawing,” *Congressus Numerantium*, Vol. 42, pp. 149–160, 1984.
- [4] T.M.J. Fruchterman, E.M. Reingold, “Graph drawing by force-directed placement,” *Software-Practice and Experience*, Vol. 21, No. 11, pp. 1129–1164, 1991.
- [5] T. Kamada, S. Kawai, “An algorithm for drawing general undirected graphs,” *Information Processing Letters*, Vol. 31, pp. 7–15, 1989.
- [6] G.W. Furnas, “Generalized fisheye views,” *Proc. CHI '86*, pp. 16–23, 1986.
- [7] J. Tanaka, “Visual programming system for parallel logic languages,” *Workshop on Parallel Logic Programming and its Program Environments, the University of Oregon*, pp. 175–186, 1994.
- [8] 南雲淳, 田中二郎 “viewPP”: グラフ構造とアニメーション表現に基づくプログラム実行の視覚化,” 日本ソフトウェア学会第14回大会論文集, pp. 17–20, 1997.
- [9] 三末和男, 杉山公造, “図的思考支援を目的とした図の多視点遠近画法について,” 情処学論, Vol. 32, pp. 997–1005, 1991.
- [10] X. Wang, I. Miyano, “Generating customized layouts,” *Graph Drawing, Symposium on Graph Drawing, GD'95, LNCS 1027*, pp. 504–515, 1995.
- [11] M.D. Storey, H.A. Müller, “Graph layout adjustment strategies,” *Graph Drawing, Symposium on Graph Drawing, GD'95, LNCS 1027*, pp. 487–499, 1995.

(平成年月日受付, 月日再受付)

田中 二郎 (正員)

1975年東京大学理学部卒, 1977年同大学院理学系研究科修士課程修了。1978年から1984年に米国ユタ大学に留学, Ph.D. in Computer Science. (財)新世代コンピュータ技術開発機構及び(株)富士通研究所を経て, 1993年より筑波大学に勤務, 現在, 電子・情報工学系教授。最近は, ビジュアル・プログラミングやユーザ・インターフェースに関心を深めている。ACM, IEEE Computer Society, 日本ソフトウェア学会, 情報処理学会, 人工知能学会各会員。

南雲 淳

1996年筑波大学第三学群情報学類卒。1998年同大学大学院修士課程理工学研究科修了。現在, 富士ゼロックス(株)オフィスドキュメントプロダクト事業本部ソフトウェアモジュール開発統轄部I&U開発部勤務。ビジュアル・プログラミングシステムやグラフ描画アルゴリズム, ユーザインターフェースに興味を持つ。日本ソフトウェア学会, 情報処理学会各会員