

CASE ツールにおけるオブジェクト図の レイアウト手法

Automatic layout method for object diagrams in CASE tool

野口 隆佳[†]

Takayoshi NOGUCHI

田中 二郎^{††}

Jiro TANAKA

[†]筑波大学博士課程工学研究科

Doctoral Program in Engineering, University of Tsukuba

^{††}筑波大学電子情報工学系

Information Science and Electronics, University of Tsukuba

概要

オブジェクト図を描画するための CASE ツールでは、レイアウト機能はオブジェクト図を描画し終えた後に用いるだけでなく、むしろその描画の途中段階において、描画を支援するために用いることが有効である。描画の途中で用いるレイアウトでは、レイアウト後の作業をスムーズに行なえるようにするために、ユーザのメンタルマップを考慮したインタラクティブ性のあるレイアウトが重要である。そこで本研究ではマグネティックスプリングモデルを改良して実現したオブジェクト図のレイアウト手法をさらに改良し、ユーザのメンタルマップを保存し、レイアウト後の作業をスムーズに行なうためのレイアウト手法を提案し、実装した。またマグネティックスプリングモデルでは考慮されていないノードの大きさに対して、新たなノード間の距離を提案することで解決し、ノードの重なりを除去した。

1 はじめに

我々はオブジェクト図というオブジェクト指向方法論の一つである OMT [1]法で用いられる図の自動レイアウトに関心がある。オブジェクト図は OMT 法におけるソフトウェア開発においてシステムの基盤となる最も重要な図であり、そのような図の自動レイアウトを考えることはソフトウェアの開発に大変有効であると思われる。オブジェクト図はクラスを矩形のノードで表し、クラス間に存在する関連、集約、継承という 3 種類の関係をエッジで表した図である。このようなオブジェクト図を描画するための CASE ツールが開発されている。しかしそのような CASE ツールでは図のレイアウトはユーザが自分で行わなければならないか、レイアウト機能があったとしても原始的であり、満足のいくレイアウトが行えないといった問題があった。

そこで我々は [4]の論文においてマグネティックスプリングモデル [2]を用いたオブジェクト図のレイアウト手法を提案し、評価を行なった。この手法では

3 種類のエッジが存在するというオブジェクト図の特徴に注目し、レイアウトを行った。3 種類のそれぞれのエッジに対して方向を決めてやり、なるべく全てのエッジが決められた方向に揃うようにノードを配置する手法である。本論文では、単にオブジェクト図を綺麗にレイアウトするのみでなく、CASE ツールにおいてオブジェクト図の描画をサポートするために、新たに改良したアルゴリズムを提案する。

2 マグネティックスプリングモデルの改良

2.1 マグネティックスプリングモデル

マグネティックスプリングモデルは無向グラフの描画アルゴリズムである Eades のスプリングモデル [5]を改良し、複数のエッジを扱えるようにしたアルゴリズムである。マグネティックスプリングモデルでは図に存在する各ノード間に 3 つの力を定義し、それらの力のつりあう位置にノードを移動すること

によってレイアウトを行う。3つの力は以下の通りである。

$$\begin{aligned} f_s &= c_s \log\left(\frac{d}{d_0}\right) \\ f_r &= c_r \frac{1}{d^2} \\ f_m &= c_m b d^\alpha |\theta|^\beta \end{aligned}$$

$c_s, c_r, c_m, \alpha, \beta$ は定数、 d はノード間の距離、 d_0 はスプリングの自然長、 b は磁場の強さ、 θ ($-\pi < \theta < \pi$) は磁場の方向とエッジの方向のなす角をそれぞれ示す。

f_s はエッジによって結ばれているノード間に働く力である。エッジをスプリングに見立てエッジが一定以上長い場合にはノードが引き合うように力が働く。短い場合にはノードが反発するような力が働く。 f_r はエッジで結ばれていないノード間に働く力である。エッジで結ばれていないノード同士は常に反発する力が働きその強さは距離に反比例する。 f_m はエッジの回転する力を表す。オブジェクト図は常にある一定の磁場の上にあると仮定し、エッジはその磁場の方向に回転するような力を受ける。この3つ目の力を用いることによってエッジの方向を制御する。

これらの3つの力の安定状態を求めることによってレイアウトを行う。

2.2 インタラクティブレイアウト

CASE ツールにおいてレイアウト機能を使用するのはオブジェクト図を描画し終えた後だけではない。むしろオブジェクト図を描画している途中の段階において、描画のサポートとなるレイアウト機能は重要である。そのような描画の途中の段階で行うレイアウトに必要なのはインタラクティブ性である。描画の途中の段階で用いるということは、レイアウトの後、描画の作業をスムーズに行えることが必要である。インタラクティブ性のあるレイアウトとはレイアウトを行った後、ユーザが次の作業にスムーズに移行できるレイアウトである。ユーザが次の作業にスムーズに移行するためにはレイアウトの結果がユーザのメンタルマップを破壊しないことが重要である。メンタルマップを破壊するとは、レイアウトによりノードの位置関係が急激に変わってしまい、どのノードがどの位置にあるのかを認識できなくなってしまうことである。よってユーザのメンタルマップを破壊しないようなレイアウトを行う

ことがインタラクティブ性のあるレイアウトを行うことになる。このようなレイアウトを実現するために、我々は4つ目の力を新たにマグネティックスプリングモデルに導入する。この力はレイアウトを行う前の位置を保存しようとする力である。メンタルマップを破壊する一番の要因はノードが大きく動きノードの相対位置関係が大幅に変わってしまう点である。よって現在の配置を保存するようにレイアウトをすることはメンタルマップを破壊しにくくし、ユーザが次の作業を行なうのに適したレイアウト手法であると考えられる。この力を我々は慣性力 (f_i) と呼び下の式で表現する。

$$\begin{aligned} &\text{if } (d_k > s) \\ &\quad \text{then} \\ &\quad \quad f_i = c_i (d_k - s)^2; \\ &\quad \text{else} \\ &\quad \quad f_i = 0; \end{aligned}$$

式における c_i は他の力とのバランスをとるための定数である。 d_k はノード k のレイアウト前の位置から現在の位置までの距離である。そして s はノードの対角線の半分の長さである。

ユーザが新しいノードをグラフに描くとレイアウト機能が動作し計算をはじめ。マグネティックスプリングモデルはイタレーションにより少しずつノード間に働く力を緩和するようにノードを移動していく。よってイタレーションの初期の段階ではノードはほとんど動いていないため f_i は微弱である。しかし計算が進むにつれノードが初期の位置から遠ざかると f_i が強く働くようになる (図1)。そしてレイアウト終了後、ユーザはまた新たにオブジェクト図にノードを加える。その次に行なわれるレイアウトでは各ノードの初期配置は前のレイアウト結果の位置にリセットされているため f_i は前のレイアウト後の位置から計算される。

このインタラクティブ性を考慮したレイアウトでは4つ目の力を導入したために、3種類のエッジの方向を考えたオブジェクト図の綺麗さよりも、次の作業を行なうためのインタラクティブ性を重視している。しかし、描画をはじめてから終了するまでに繰り返しこのレイアウトを用いることによって緩やかに3種類のエッジの方向を考えたレイアウトに収束していくために、各レイアウトの段階ではインタラクティブ性を考慮し、最終的にはオブジェクト図の特徴を考えた綺麗なレイアウトを行なうことが出

来るのである。

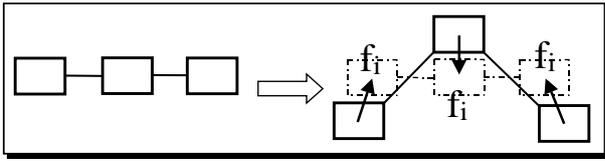


図1 f_i の力のかかり方

f_i の定義の中で s を用いる理由は大きいノード程慣性力が影響しはじめるのを遅くするためである。なぜなら動いた距離が同じであっても、大きいノードは前の位置と重なる部分が大きく見ための位置の変化が少ないからである。

他の力に対する f_i の割合を大きくとればより現在の位置を保存したレイアウトができメンタルマップを破壊しない。よって描画途中のメンタルマップを保存したレイアウトが可能になるのである。

2.3 ノードの大きさへの対応

マグネティックスプリングモデルでは、ノードはすべて無限小の点として扱われるため、そのままオブジェクトのレイアウトに用いてしまうと、ノードの間に重なりが生じてしまう。このような重なりを除去する方法としては以下のような手法がある。

- レイアウト後にノードの重なりを直線的に除去する手法 [3]

この手法ではレイアウトが終わった後にノードの重なりがあった時、重なりのあるノード同士をそのノードの中心を結んだ直線上を移動させることでノードの重なりを除去する。この手法には2つの大きな問題がある。一つ目はこの手法はレイアウトアルゴリズムとは独立しているために前もって Eades のスプリングモデルでレイアウトをしておかなければならない。二つ目は Eades のスプリングモデルでレイアウトした結果をこの手法によって破壊してしまう点である。そこでこの問題を解決するための手法として以下のような手法が提案された。

- ノードの大きさによってスプリングの自然長を変化させる手法 [4]

この手法ではノードが大きい場合そのノードにつながっているエッジの自然長を長くするという方法である。この手法はノードが円の場合は有効であるがオブジェクト図で扱うノードは矩形であるため

縦に長いノードや横に長いノードの場合ノード間の理想距離が変化してしまう問題が残されていた (図2)。

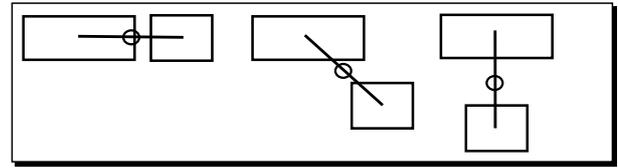


図2 ノード間の中心間の距離と実際のノード間の距離の変化

先の手法においての問題はノード間の距離をノードの中心から中心までの距離としていたために起きた問題である。なぜならマグネティックスプリングモデルではノードを無限小の点で表しており、そのアルゴリズムを拡張したことからノード間の距離はもともとそのノードを示す中心の点を用いて計算されていたためである。

この問題を解決するために今回、スプリングの長さとしてノード間の距離をノードの端から端までの実際の距離に変更した (図3)。この方法ではノード間の距離にノードの大きさは一切関係しないために、ノードの形状に関らずノード間の理想距離は一定となり、ノードの大きさを加味したレイアウトが可能になる。

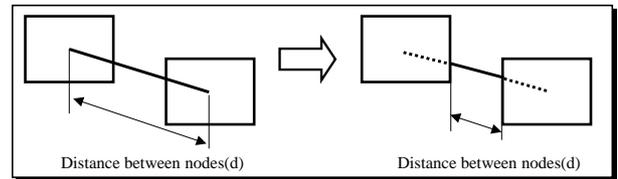


図3 ノード間の距離の定義

しかしこの場合問題となるのがノードが重なっていた時のスプリングの長さやノード間の距離である。重なっていた場合のノード間の距離を0とすると f_s や f_r の定義からそれぞれの力が無限大になってしまい、レイアウトをおこなった時にノードが無制限に飛んでいってしまう問題が発生する。それを回避するためにノードが重なっていた場合にはノード間の距離を予め決めておいた極めて短い長さにし、適度に大きい力がかかるようにし、ノードの重なりを許さないようにした。

3 適用結果

アニメーションシステム OSCAR [1]のオブジェクト図に実際に本アルゴリズムを適用した結果を図4～図6に示す。

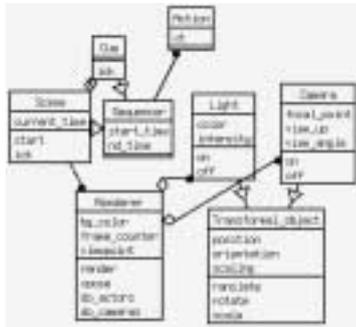


図4 レイアウト前

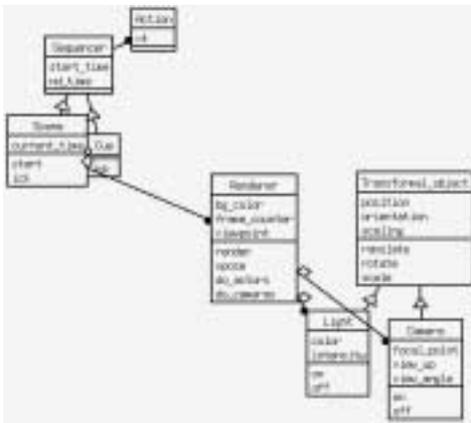


図5 改良前のマグネティックスプリング

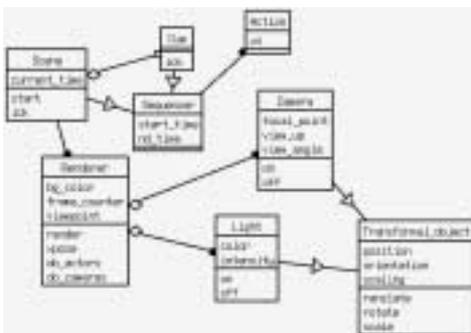


図6 メンタルマップを考慮したレイアウト後

図4はレイアウトをする前の状態である。図5は慣性力を導入していないマグネティックスプリングモデルによるレイアウトである。またノードの重な

りを除去する手法も [4]の手法を用いている。図6はマグネティックスプリングモデルに慣性力を導入し、ノードの重なりを除去する手法も今回新たに提案した手法を用いた結果である。

図5ではマグネティックスプリングで与えた制約をほぼ満たしており見やすいレイアウトを行うことが出来ている。しかし図4と比べるとノードの位置関係が大幅に変更されてしまっているためにこの後のオブジェクト図の描画の作業に支障がでてしまう。またノードの重なりを [4]の手法で除去したため、小さいノードのスプリングは短くなってしまい、Que クラスはとなりの Scene クラスとほとんどくっついてしまっている。図6では、Light クラスや Camera クラスは図5ほど大きくは動かずにレイアウトが徐々に進みつつある状態で安定している。よってノードの位置関係は図5ほど大きく変更されていない。これはメンタルマップが保存されたレイアウトが行われている結果である。

4 結論

本研究では、OMTにおけるオブジェクト図のレイアウト手法としてマグネティックスプリングモデルを採用した。本手法では各関係エッジの特徴からその方向を定め、エッジの方向によってクラス間の関係が容易に把握できるようなレイアウトを行った。またマグネティックスプリングモデルに新たな力を導入することでメンタルマップを考慮したレイアウトが行えるようになった。またノード間の距離の定義を新たなものに変更することでノードの重なりをより効率良く除去する手法を提案し実装した。

参考文献

- [1] James Rumbaugh et al(羽生田栄一 監訳): オブジェクト指向方法論 OMT - モデル化と設計 トッパン (1992)
- [2] Kozo Sugiyama, Kazuo Misue: *Graph Drawing by Magnetic Spring Model* Journal of Visual Languages and Computing, Vol.6, no.3, pp.217-231 (1995)
- [3] 中島哲 田中二郎: オブジェクト指向方法論に基づくオブジェクト図の自動レイアウト 情報処理学会論文誌, Vol.39, No.12, pp.3282-3293 (1998)
- [4] Takayoshi Noguchi, Jiro Tanaka: *New Automatic Layout Method based on Magnetic Spring Model for Object Diagrams of OMT* International Symposium on Future Software Technology 98 (ISFST98), pp.89-94 (1998)
- [5] Peter Eades: *A Heuristics for Graph Drawing* Congressus Numerantium, Vol.42, pp.149-160 (1984)