

筑波大学大学院博士課程

システム情報工学研究科修士論文

有向グラフにおける変化過程の
可視化手法の開発

菱田 哲史

(コンピュータサイエンス専攻)

指導教員 三末 和男

2012年3月

概要

時間の経過とともに変化する有向グラフを対象にそれが変化する過程を視覚的に提示する手法を開発した。実現にあたり、有向グラフにおける変化を定め、情報可視化の適用可能な領域を拡張した。開発した手法は、エッジに局所的な座標系を導入し、線分の形状を用いて有向グラフにおける変化を提示する。これにより、向きの逆転も含めた接続関係の変化の履歴を一望することが可能となった。また、提案手法を実際のデータに適用させ、手法の有効性を示した。

目次

| | |
|----------------------|-----------|
| 第1章 序論 | 1 |
| 1.1 情報可視化 | 1 |
| 1.2 有向グラフと無向グラフ | 1 |
| 1.2.1 グラフにおける変化 | 2 |
| 1.3 有向グラフにおける変化過程 | 3 |
| 1.4 本研究の目的 | 3 |
| 1.5 論文の構成 | 4 |
| 第2章 関連研究 | 5 |
| 2.1 時系列グラフの可視化手法に関して | 5 |
| 2.1.1 アニメーションの利用 | 5 |
| 2.1.2 スナップショット群の列挙 | 6 |
| 2.1.3 3次元空間への拡張 | 6 |
| 2.1.4 2次元平面での表現 | 7 |
| 2.2 複合的なグラフ可視化手法に関して | 7 |
| 第3章 可視化における要求 | 8 |
| 3.1 時系列有向グラフにおける変化 | 8 |
| 3.2 時系列グラフの可視化における要求 | 8 |
| 3.3 有向グラフの可視化における要求 | 9 |
| 第4章 提案手法 | 11 |
| 4.1 提案手法の概要 | 11 |
| 4.2 対象の定式化 | 11 |
| 4.3 描画スタイル | 12 |
| 4.4 配置手法 | 13 |
| 4.4.1 配置手法の概要 | 13 |
| 4.4.2 ノード, エッジの配置手法 | 13 |
| 4.4.3 連結成分の配置手法 | 13 |
| 4.5 要素の表現手法 | 16 |
| 4.6 時間情報間の距離 | 22 |
| 第5章 実装 | 25 |
| 5.1 ツールの概要 | 25 |

| | | |
|------------|-------------------|-----------|
| 5.1.1 | 構成 | 26 |
| 5.2 | 機能説明 | 26 |
| 第6章 | ケーススタディ | 28 |
| 6.1 | アクセスログの可視化 | 28 |
| 第7章 | 議論 | 33 |
| 7.1 | 考察 | 33 |
| 7.1.1 | 局所的な座標系の導入に関して | 33 |
| 7.1.2 | 時間情報の表現の可読性に関して | 33 |
| 7.1.3 | 無向辺を含むグラフへの適用に関して | 34 |
| 7.2 | 今後の課題 | 34 |
| 7.2.1 | エッジの形状の自動的な調整 | 34 |
| 7.2.2 | クリークへの対応 | 35 |
| 7.2.3 | 動的なデータ追加への対応 | 35 |
| 7.2.4 | サブグラフの同型判定 | 35 |
| 第8章 | 結論 | 36 |
| | 謝辞 | 37 |
| | 参考文献 | 38 |

目次

| | | |
|------|-----------------------------------|----|
| 1.1 | グラフの可視化例 | 2 |
| 1.2 | 本論文で挑戦する領域 | 3 |
| 3.1 | 矢印記号と有向辺の交差 | 10 |
| 4.1 | 連結成分をレイアウト空間の中心に寄せる力 | 15 |
| 4.2 | バウンディングボックスを縮める力 | 15 |
| 4.3 | エッジの時間情報の表現 | 16 |
| 4.4 | 該当する時刻に対応する位置の決定 | 17 |
| 4.5 | 線分の端点の決定 | 18 |
| 4.6 | 線分の描画 | 19 |
| 4.7 | 描画された1本のエッジの時間情報 | 19 |
| 4.8 | 時間情報の表現の改善 | 20 |
| 4.9 | 読み取りが期待できるパターン | 21 |
| 4.10 | ”行き来”が発生したパターン | 22 |
| 4.11 | 動的時間伸縮法の算出イメージ | 23 |
| 5.1 | アプリケーションの概観 | 25 |
| 5.2 | エッジ形状の操作 | 27 |
| 6.1 | ページ遷移のイメージ | 28 |
| 6.2 | 適用結果 | 29 |
| 6.3 | /software/ディレクトリ化の連結成分 | 30 |
| 6.4 | エッジの形状を変化させた図 | 31 |
| 6.5 | 連番ページ群を囲った図 | 32 |
| 6.6 | エッジの接続先の属性ごとに異なる線分のパターン | 32 |
| 7.1 | 提案手法を用いた無向辺の描画 | 34 |

表目次

| | |
|----------------------|----|
| 4.1 差分テーブル | 23 |
|----------------------|----|

第1章 序論

1.1 情報可視化

科学技術計算における計測, 計算結果を視覚的に提示する科学的可視化と呼ばれる研究分野が存在する. また, 先述の科学的可視化の流れを汲み, データから人間が情報を獲得する場면을支援することを目的とした情報可視化と呼ばれる研究分野が存在する. 本研究では, 情報可視化を研究の対象とする.

情報可視化の分類

我々は関係構造や数値情報を視覚的に示し, それを把握しようとする試みを日常的に行っている. また, そのような活動は情報を直感的に把握する場面において有用であることを経験的に認識している. そのため, これまでに情報を可視化する手法はこれまで多数提案されてきた. 出原は, 図の基本形として可視化手法を以下の4種に分類している [1]. 可視化の例を図 1.1 に示す.

- 連結系: ネットワーク図など
- 座標系: 株価チャート, バーチャート (棒グラフ) など
- 行列系: 行列, クロス集計表など
- 領域系: オイラー図, treemap など

また, 連結系と領域系の手法を組み合わせた複合系も手法として併せて挙げている.

座標系, 行列系に関する研究は進んでおり, 汎用性の高い表現手法が確立されている. 他方, 連結系やこれを含む複合系に関しては, 視覚的表現として有用でありかつ日常的に利用されているのにも関わらず, 表現手法が確立されていない [2]. また, 連結系は, つながりや関係など座標系などで表現される数値や文字列情報と比較して把握が困難な情報を適用される事が多い. そのため, 連結系を基とした表現手法に関する研究が現在も活発に行われている. 本研究もまた, 連結系を基とした表現手法を開発する.

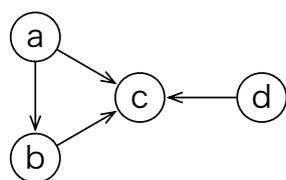
1.2 有向グラフと無向グラフ

本論文では有向グラフを対象とし, これの提示手法について研究を行った. 以下にグラフについて説明する. グラフとは, 頂点となるノードとそれらの関係を表すエッジか

$$G = (V, E)$$

$$V = \{a, b, c, d\}$$

$$E = \{\langle a, b \rangle, \langle b, c \rangle, \langle a, c \rangle, \langle d, c \rangle\}$$



網図を用いた表現

| | a | b | c | d |
|---|---|---|---|---|
| a | 0 | 1 | 1 | 0 |
| b | 0 | 0 | 1 | 0 |
| c | 0 | 0 | 0 | 0 |
| d | 0 | 0 | 1 | 0 |

行列を用いた表現

図 1.1: グラフの可視化例

ら構成される抽象的なデータ型である。エッジの性質より、関係において向きを持つ有向グラフ, 持たない無向グラフに大別できる。無向グラフは, 対象とする要素同士の関係が同値関係である場合に多く利用される。有向グラフは順序関係や従属関係などに代表される, 同値でない関係を要素間が持つ場合に利用される。一般に, 有向や無向を問わず, グラフは何らかの情報や要素間の関係を抽象的に表現するのに有用である概念であり [2], 数学的問題の解決のみならず, 近年ではインターネット上における Web ページのリンク構造や人同士の交遊関係 [3] などの持つ性質を明らかにする場面で頻繁に利用される。

1.2.1 グラフにおける変化

現実中存在する情報は静的にその内容を保ち続ける事は少なく, 多くの情報は絶えず変化し続ける。近年, このような前提に基づいて変化する情報, 特に変化するグラフに関して, これらを視覚的に提示する試みがなされている。時間の経過に応じて変化するグラフの集積を時系列グラフと呼ぶ。また, 時間経過により変化する有向グラフを時系列有向グラフと呼ぶ。時系列グラフにおける変化については様々な定義が存在するが, それらは以下の2種に大別される。

- ノードやエッジの追加, 削除, エッジの向きの逆転による関係構造の変化
- ノードやエッジに付与された重みの変化

後者はグラフの一種である重み付きグラフにおいて定義される変化の一つであり, 本論文では一般的なグラフを時系列空間へ拡張した構造を対象としている。本論文にお

けるグラフの変化は前者を対象としており,特にエッジの向きの逆転による変化に注目する.

1.3 有向グラフにおける変化過程

本論文で扱うグラフにおける変化過程とは,グラフ内の各要素の変化の履歴である.これを観測,分析する事により,ある時間帯において何度も変化した部分グラフ,ある部分グラフに関して変化が頻発した時間帯などグラフと時間における関係性の発見が期待できる.また,有向グラフの変化過程であれば,一方的な接続関係,双方向からの密な接続関係など接続関係の時間的特徴の発見も期待できる.

このような情報を視覚的に提示する事により,閲覧者はグラフおよびその変化の全体像を俯瞰する事が可能となる.これにより直感的に要素間のつながりやそれらの変化を認知できるようになる.また,時間変化の全体像が提示されることにより,予測されなかったパターンの検出 [4] やチャンス発見 [5] など,閲覧者の知識獲得支援が期待できる.

1.4 本研究の目的

本研究の目的は,関係構造が変化する有向グラフを対象に,これの変化する過程の把握を支援することである.従来の手法では十分にサポートされていなかった,有向グラフにおけるエッジの向きの変化(図 1.2)も含めたグラフ変化の把握が容易にすることを旨とする.

時間経過により変化するグラフ

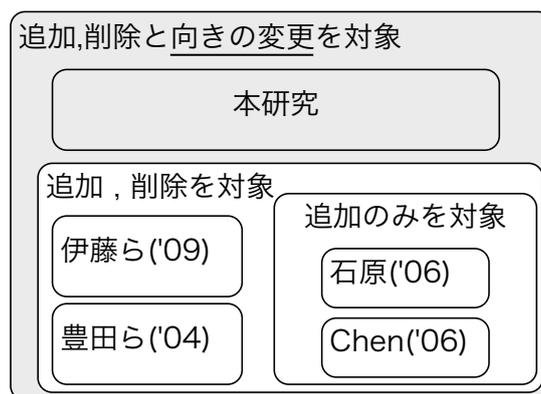


図 1.2: 本論文で挑戦する領域

1.5 論文の構成

1章以降の本論文の構成は以下の通りである。2章にてグラフの変化を視覚的に提示する手法を紹介し、本研究が挑戦する領域を示す。続く3章で可視化における要件を示す。4章は、要件を基に提案する手法についての説明と有向グラフにおける変化過程の定式化を行う。5章で提案手法を実装したツールを示す。6章では、ケーススタディを行い、手法の有効性を示す。7章で提案手法に関して議論を行い、8章での結論を以て本論文を結ぶ。

第2章 関連研究

2.1 時系列グラフの可視化手法に関して

本章では、時系列グラフの可視化手法をその持つ時間情報の表現に対するアプローチから以下の4種に整理し、それぞれの研究について述べる。

- アニメーションのように、現実の時間軸を利用して時間軸を表現するアプローチ
- 各時刻におけるグラフを時系列順に並べて提示することにより時間軸を表現するアプローチ
- 3次元空間上の1つ以上の基底を用いて時間軸を表現するアプローチ
- 2次元平面上で関係構造および時間軸を表現するアプローチ

2.1.1 アニメーションの利用

本節では、現実の時間軸を利用して時間を表現する手法について述べる。この手法はアニメーションだけでなくユーザのインタラクションにより提示する時刻、時間帯を変化させるアプローチも含む。以下に、そのようなアプローチを用いた研究を紹介する。

Moodyらは、アニメーションによるグラフ可視化手法としてstaticflipbooksとdynamicmovieを提案した[6]。staticflipbooksはノードの位置を固定したアニメーションを提示する。dynamicmovieはノード間の関係性の変化とその位置を対応付けてたアニメーションを提示する。彼らはこれらの手法をソーシャルネットワーク分析に利用した。

Christianらは、CVSを基にソフトウェア開発の過程を可視化する手法を提案した[7]。彼らは、各時間のグラフに力学的モデルを適用させるだけでなく、グラフ間にまたがって存在する同一のノードに対して同じ位置を保持するような力を加えてレイアウトを求める。また、描画に用いる色情報と異なる2つの属性を対応付けている点が可視化手法の特徴である。

鈴木らは、一般的な時系列グラフを可視化するツールを開発し、これを情報伝播モデルを基にした時系列グラフに適用させている[8]。

豊田は、大規模な動的グラフの可視化において、閲覧者のメンタルマップを保持するレイアウト手法を開発した[9]。ユーザのインタラクションに対応できるよう、スーパーグラフと表示するグラフのレイアウトを同時に計算する点がこの手法の特徴である。

中園らはグラフの変化をインタラクティブに閲覧するツール Nel2 を開発した [10]. このツールは 2 次元平面上に各時刻のグラフを時系列順に重ねて配置し, 差分を色情報で表現する. ユーザはスライダーバーで提示する期間を操作し閲覧する事により変化を知覚できる.

2.1.2 スナップショット群の列挙

Erten らは, 同一のノードやエッジを持った一連のグラフを閲覧できるよう AggregateLayout, MergedLayout, IndependentIterationLayout の 3 つのレイアウト手法とそれぞれの手法を用いた AggregateView, MergedView, SplitView の 3 つのビューを提案している [11]. 提案している IndependentIterationLayout を用いた SplitView は 2 つの図を並べて提示するものである.

豊田らは, Web ページのリンク構造の発展過程を可視化する WebRelievo を開発した [12]. この可視化手法は, 差分ビュー, クラスタビューを時系列に沿って並べて表示する点の特徴である. 特に差分ビューは時系列において連続するの 2 グラフ間の変化を 4 つに分類し, それぞれの変化と色情報を対応付けている. 著者らはこれら 2 つの手法が新しいページや関係の追加, グラフ中のクラスタの分裂や結合の把握に有用であると述べている.

2.1.3 3次元空間への拡張

3次元空間を利用する利点は, レイアウトに利用できる領域が広がるため, ノード数が大規模なグラフや時間ステップ数が多い時系列グラフの表現に対応できる点である.

Chi らは, Web ページの構造とその発展過程を可視化する TimeTube を提案した [13]. この手法はまず, 対象ページをルートとするディスクツリーを 2 次元平面上に描画する. そして各時刻におけるディスクツリーを 3 次元空間上に並べる.

先に挙げた Erten らの MergedLayout を用いた MergedView は, 2 次元平面に描画されたグラフを 3 次元空間上に重畳したビューを提示する. 著者らは, この手法とビューによって閲覧者はメンタルマップを保持したまま各グラフを閲覧できると述べている.

Marco, Ulrik らは, 各時間のグラフをレイヤーとして捉え, それらを 3 次元空間に積み重ねる手法を提案している. 彼らが提案した手法のように, 2 次元平面で描画した各時刻のグラフのスナップショットを時系列順に積み重ねる 3 次元レイアウト手法は 2.5 次元レイアウト手法と呼ばれる.

伊藤らは 2.5 次元レイアウト手法を基として, アニメーションやスナップショット群を提示するアプローチを用いたビューを適時生成するためのインタラクション手法を開発している [14].

2.1.4 2次元平面での表現

先に挙げた Erten らの AggregateLayout, AggregateView は単一の図に全てのグラフレイアウトを反映させる手法である. 可視化手法の特徴として, 各グラフ中のノードやエッジに異なる描画属性を与えることにより, 閲覧者がそれぞれのグラフを判別できるようにした点が挙げられる.

石原は, 動的ネットワークの成長過程を可視化する波紋表現と差分を可視化する差分表現を提案した [15]. 波紋表現は時間情報をエッジの長さに対応付けて変化過程を表現する手法である. 石原はこの手法を Weblog 記事のリンク関係に適用し, 話題の広がりや可視化した. Chen は, ノードやエッジの追加を年輪のメタファを用いて表現する手法を開発した [16]. 手法を論文と語の引用関係の発展過程に適用させ, 研究分野のトレンドやパターンの可視化を行っている. Yarden らは, グラフ構造と履歴情報における時間と属性の相関関係を1つの図で表現する手法を提案した [17]. 時間と属性の情報がレイアウトされた結果を囲むように円型配置される点がこの手法の特徴である.

2.2 複合的なグラフ可視化手法に関して

近年, 複合的な可視化手法に関する研究が行われており, その実現を助けるフレームワークが開発されている. 伊藤らは, グラフ構造の変化および要素に付与された重みの変化を3次元空間上で可視化するフレームワークを開発した []. フレームワークは3次元空間の1軸を時間軸に対応づけ, 任意の時刻における像を提示する. また, 赤石と共に史料データを基に, 異なる関係構造を抽出しそれらの可視化を行っている [18]. Steffen らは, 既存の時系列グラフの可視化手法と別の側面から抽出した情報を統合して提示するための手法を提案した [19]. 本研究で提案する手法はそのようなフレームワークでの使用も想定している.

第3章 可視化における要求

本章では、時系列有向グラフの変化について検討し、可視化に求められる要求について整理する。

3.1 時系列有向グラフにおける変化

本論文においては、時系列有向グラフにおける変化をノードやエッジの追加と削除およびエッジの向きへの逆転と定めている。以下の節において、本論文で扱う変化が時間経過による関係構造の変化を包括しているか否かを検討する。

要素の変化に関して

1.2.1 節で述べたように時系列グラフにおける変化は関係構造の変化と付与される重みの変化に分類される。後者は、[20]のように量的な推移を提示する場合に関係構造の表現に強く依存しない座標系を基としたアプローチが確立されつつある。本研究では、提示手法において十分な検討がなされていない関係構造の変化に焦点を当てる。

関係構造そのもの変化に着目すると、ノードの統合や分離などもまた変化の一種として挙げられるであろう。ノードの統合や分離の例は企業とそれらの取引関係 [21] かなるグラフにおける企業の合併や分社化などが挙げられる。

しかしながら、いずれの変化も先に挙げたノードやエッジの追加、削除を組み合わせる事が可能である。ノードの統合は複数ノードの削除と1ノードの追加、ノードの分離は1ノードの削除と複数ノードの追加としてそれぞれの変化を表現できる。よって、本研究ではノードやエッジの追加と削除、エッジの向きへの逆転を時系列有向グラフの変化とする。

3.2 時系列グラフの可視化における要求

各要素における全時刻の変化を一覧できる

変化過程の全体像があらかじめ提示される必要がある。そのため、アニメーションのような表現を用いて、一部の時間帯、要素を省略した像が連続して提示される場合、閲覧者はそれぞれの像を記憶しそれらを比較することにより変化の把握が可能となる。グラフの規模、総時間ステップ数が増大すると、閲覧者が記憶する像の数、比較の頻

度が増大し、認知的な負担もまた増大する。また、ログファイルなど瞬間的に関係が発生する情報をアニメーションに適用させた場合、時刻ごとに関係構造が大きく異なるグラフが提示される傾向にある。隣接する時刻においてそれぞれ異なる像が提示されるため、少ない時間ステップ数であっても認知的に負担のかかるビューが提示される。よって、可視化を行うにあたっては、全時刻の関係構造を1つの像で提示する表現が要求される。

同定を行うことなく同一要素の変化を閲覧できる

グラフにおける時間ステップ数が少なければ、スナップショットの列や2.5次元表現に代表される、時間ステップにおけるグラフと像を対応づけた表現を用いることにより、要素の同定を行いつつ変化を把握することは可能である。しかしながらこのような手法は、総時間ステップ数の増大に伴い提示される像の数が増大し、閲覧者に認知的な負担がかかる。よって要素の変化を表す情報が1つの像に集約された表現が要求される。

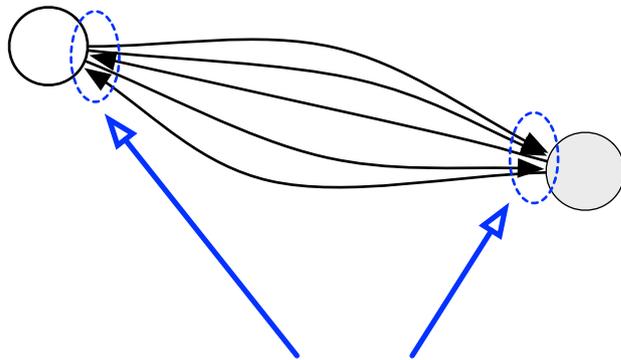
変化に関して近しい要素を視認できる

多くのグラフ可視化手法においては、接続関係の強い要素同士が互いに近くにおかれる傾向にある。時系列グラフにおいては、関係構造だけでなく、時系列も重要な情報であり、時間において近しい要素同士が視認できる表現が要求される。

3.3 有向グラフの可視化における要求

有向辺の向きを閲覧できる

一般に、有向辺の表現は矢印を用いて表現される。有向辺の接続方向の逆転が頻出するグラフにおいて、1回の接続関係を1本の矢印付きの線分を用いて表現した場合、図3.1のように描画要素同士の交差が発生し、向きの変化の可読性が極端に低下するおそれがある。よって、向きの変化の可読性が接続関係の変化が発生した回数に依存せず、変化の変遷の把握が容易となる提示が望まれる。



矢印記号と辺が交差しているため
それぞれの有向辺の向きが把握が困難

図 3.1: 矢印記号と有向辺の交差

第4章 提案手法

提案手法はグラフにおける関係構造と各要素の変化を2次元平面上に提示する。本節では、まず時系列有向グラフの定式化を行う。そして、手法における描画のルールについて述べる。その後、グラフの要素の配置手法およびそれらの表現手法について述べる。

4.1 提案手法の概要

提案手法は以下のような特徴を持つ。

- 2次元平面上に関係構造を時系列の情報を表現する。
- 平面上に関係構造における座標系と異なる局所的な座標系を設け、時間軸と対応づける
- 矢印記号などの記号を用いず、線分を用いてエッジの向きを表現する
- 同じ時刻ないし近い時刻にグラフに追加され、また同じ時刻にグラフから削除された要素同士を互いに近づけて配置する。

4.2 対象の定式化

多くの時系列グラフ可視化の研究では、対象とするグラフを各時刻におけるグラフを要素とした集合として定式化されている [22]。本論文では、これとは異なる定式化を行う。本論文においては、各時刻におけるグラフを総計したグラフを対象とする。つまり、ある要素に関して、追加、削除、追加の順に変化が発生した場合においても、削除が発生する前の要素と再び追加された要素を同じ要素と見なす。

なお、総計にあたりエッジに関して、向きの異なる有向辺は纏めて1本のエッジとして扱う。このように変換することにより、エッジの存在の有無は総計された時間帯において接続関係が発生したか否かを表し、有向辺の向きの逆転は一本のエッジにおいて発生した変化として表すことが可能となる。

総計されたグラフを構成するノード、エッジに関して、変化の内容とその変化が発生した時刻の組を要素とした列を付与する。以降、この列を時間情報と呼ぶ。本論文で扱うエッジは2本の有向辺を統合した辺であるため、それぞれ2つの時間情報が付与さ

れる. 時間情報 t は変化の種類を表す集合 C と全時刻の集合 T の直積集合の部分集合として式 4.1 のように表せる.

$$\begin{aligned} C &= \{add, remove\} \\ t &\subseteq C \times T \end{aligned} \quad (4.1)$$

本論文で扱う時間情報を付与したグラフは上記の操作を経て得る. 式 4.2 のこの式を示す. なお, 式中の $t_{\langle u,v \rangle}$ は u を起点とし v を終点とした有向辺から得た時間情報を表し, $t_{\langle v,u \rangle}$ は v を起点とし u を終点とした有向辺から得た時間情報を表す.

$$\begin{aligned} G &= (V, E) \\ E &\subseteq \{\{(u, t_{\langle u,v \rangle}), (v, t_{\langle v,u \rangle})\} \mid u, v \in V\} \end{aligned} \quad (4.2)$$

また, 総計されたグラフ G に関して, 連結成分が複数発生しうると考えられる. よって, 定式化したグラフ G が n 個の連結成分を持つ場合, グラフ G を時間情報を付与されたノードとエッジからなる部分グラフ g の集合として表す. 式 4.3 に定式化された連結成分とグラフの関係を示す.

$$\begin{aligned} G &= \{g_0, g_1, \dots, g_{n-1}\} \\ g_i &= (V_i, E_i) (0 \leq i < n, V_i \subseteq V) \\ E_i &\subseteq \{\{(u, t_{\langle u,v \rangle}), (v, t_{\langle v,u \rangle})\} \mid u, v \in V_i\} \subseteq E \end{aligned} \quad (4.3)$$

本論文においては, このようにノードやエッジだけでなく部分グラフとして表せる連結成分も併せて表現に組み込む.

4.3 描画スタイル

描画の対象

描画の対象は前節で示したグラフである. ノード, エッジおよびエッジの時間情報とそれらから構成される連結成分が描画の対象となる.

要素の描画規約

ノードの配置規約は自由配置である. エッジの配線規約は折れ線配線であり, 座標系はノード配置における座標系に従う.

エッジの持つ時間情報の配線は局所的に異なる座標系を導入した直線配線であり, 下向き描画として描画する. また, 連結成分の配置規約は自由配置である.

4.4 配置手法

4.4.1 配置手法の概要

本論文では, 連結成分を含めたグラフに物理モデルを同時に適用させる. Eades[23]の物理モデルを採用し, これを基にレイアウトを行う. 以下の4ステップを繰り返し各要素の最適な位置を求める.

1. ノードに物理モデルを適用させる
2. ノードへの適用結果を基に連結成分にも物理モデルを適用させる
3. 連結成分自身に働く力をそれに含まれる各ノードに反映させる
4. ノードの配置を決定する

なお, 物理モデルの適用を終了させる条件は, 全ノードの中で最も移動したノードの移動量とそれが属する連結成分がなすバウンディングボックスの中心座標の移動量の和が閾値を下回った場合とする.

4.4.2 ノード, エッジの配置手法

本論文では, Eades[23]の力指向レイアウトを基にノードを配置する. なお, Eadesのモデルにおけるノード間の斥力は連結成分内のノード間にものみ作用させる.

4.4.3 連結成分の配置手法

概要

連結成分の配置においては, 連結成分内のノードがなすバウンディングボックスを基にこれに与えるべき力を求め, これに含まれるノード群に力を反映させる. 以下にバウンディングボックスの位置形状を基にノードへ反映させる力について述べる.

包含するノード数に応じたレイアウト

読み込むデータによっては, グラフ中に連結成分が複数現れる場合が存在する. これらの連結成分が互いに交差したまま配置されると, 関係構造および時間情報に関して可読性が大きく損なわれると考えられる.

また, 包含するノード数が大きく異なる連結成分同士が隣接して配置された場合, 視覚的に混雑した印象を閲覧者に与え, 図から情報を読み取ることが困難になると考えられる. 本論文では, 連結成分においても物理モデルを適用させ, 成分同士の交差の低減を図る. また, 成分同士の距離を意味付けし視認性の向上を図る.

各連結成分に関して、連結成分を構成するノード群を完全に包含するバウンディングボックスを生成し、成分間の交差の低減を図る。バウンディングボックス同士が交差する場合、交差が完全に無くなるまで互いに斥力を働かせる。このとき、バウンディングボックスの中心座標同士の距離に応じて斥力の強さを変化させる。また、交差する2つのバウンディングボックスにかかる力は交差する他方の連結成分の持つノード数に応じて変化させる。グラフ G 中の交差する連結成分 g_i, g_j に関して、連結成分 g_i に働く斥力を式 4.4 に示す。式中の k_b は斥力の係数であり、また、 $p(g_i), p(g_j)$ は連結成分 g_i, g_j の持つノード群がなすバウンディングボックスの中心座標を表す。

$$F_{g_i}(g_j) = \frac{k_b}{(p(g_i) - p(g_j))^2} \cdot \frac{|g_j|}{|g_j| + |g_i|} \quad (4.4)$$

なお、このモデルでは2つのバウンディングボックスの中心座標が一致する場合、斥力を計算出来なくなる。そのため、中心座標間の距離が閾値を下回る場合、両バウンディングボックスにランダムな方向への力を働かせる。

この手法を用いる際にレイアウト空間に対して空間的な制限を設けた場合、ノードの配置における係数や空間の大きさによっては、手法の適用後も連結成分同士の交差が発生することが考えられる。

要素の時間情報に応じたレイアウト

本論文では、連結成分においても時間情報を定める。連結成分間の時間情報における距離を求め、レイアウトに反映させる。

連結成分に関して、これに含まれる各要素の時間情報を総計した列を本論文においては連結成分の時間情報と定め、他の連結成分との距離を連結成分間に働く斥力に対応付ける。時間情報間の距離は 4.6 節にて説明する。

レイアウト空間中心部へ移動させるレイアウト

連結成分同士が極端に離れた位置に配置された場合、可読性が大きく低下するおそれがある。本論文では、物理モデルを適用させたレイアウト空間を定め、この空間の中心へ向かう力を連結成分に作用させる。このとき、作用させる力の強さは連結成分内のノード数に対応づける。

この力を作用させ、レイアウト空間における中心を提示する像の中心に対応づけることにより、提示される連結成分が像の中心に寄せられるため、グラフの概観把握が容易になると期待できる。

ある連結成分 g に関してレイアウト空間へ向かう力は式 4.5 のように定める。なお、式 4.5 内の k_l は引き寄せる力の係数、 $center_{Area}$ はレイアウト空間の中心座標を表す。

$$F = k_l \cdot |g| \cdot |p(g) - center_{Area}| \quad (4.5)$$

この力は多くのノードを持つ連結成分はレイアウト空間の中央に寄せる. この力を作用させることによって, 図4.1のようにレイアウト空間上の位置から連結成分の持つおよそのノード数を把握することが容易になると考えられる.

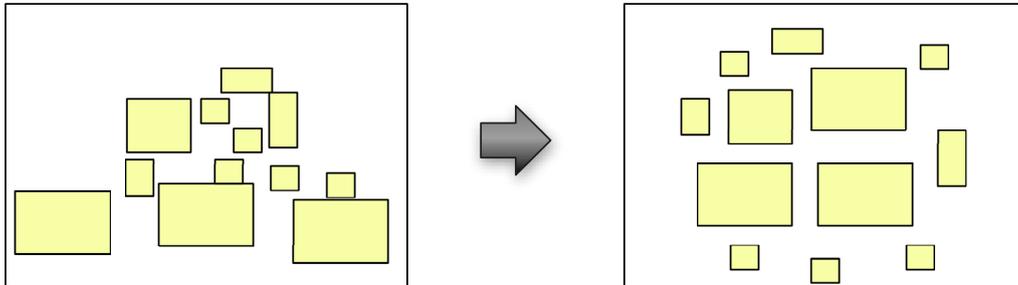


図 4.1: 連結成分をレイアウト空間の中心に寄せる力

バウンディングボックスを縮尺させるレイアウト

連結成分のバウンディングボックスが肥大化しないよう, バウンディングボックスの大きさを縮める力を連結成分内のノードに働かせる. この力は斎藤 [24] の提案するアルゴリズムを一部採用する. 力のかけ方に関して, 図4.2を用いて説明する.

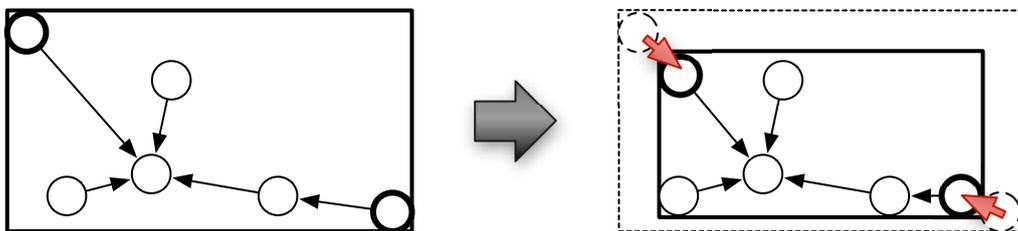


図 4.2: バウンディングボックスを縮める力

図4.2左は, 連結成分のレイアウト結果とそれのバウンディングボックスを表している. また, 力を加える対象としたノードを太枠線で示す. 図4.2右は縮める力の向きを示す. 図4.2左を見ると, 左上に配置されたノードと右下部に配置されたノードはバウンディングボックスに接している. このノードに対して, バウンディングボックス中心に向かって近づくように力をかける.

斎藤のアルゴリズムでは, 理想とするバウンディングボックスの大きさ, および配置が予め与えられている. そのため, ノードにかかる力の向きは, バウンディングボックスに接する各ノードの座標から得る重心としている.

本論文においては対象の連結成分に関して, 理想的なバウンディングボックスの大きさや配置は与えてられていない. 力を導入した目的はバウンディングボックスの肥大化を防ぐ事であるため, 重心でなくバウンディングボックスの中心座標を基にノー

ドにかかる力の向きを決定している. なお, この力に関しては, 閾値を定めこれを下回る場合は求めた力をノードに与えない.

4.5 要素の表現手法

ノードの表現手法

ノードは, 一般的な網図におけるノードの表現に則り, 円ないし多角形を用いて表現する.

エッジおよびエッジの持つ時間情報の表現手法

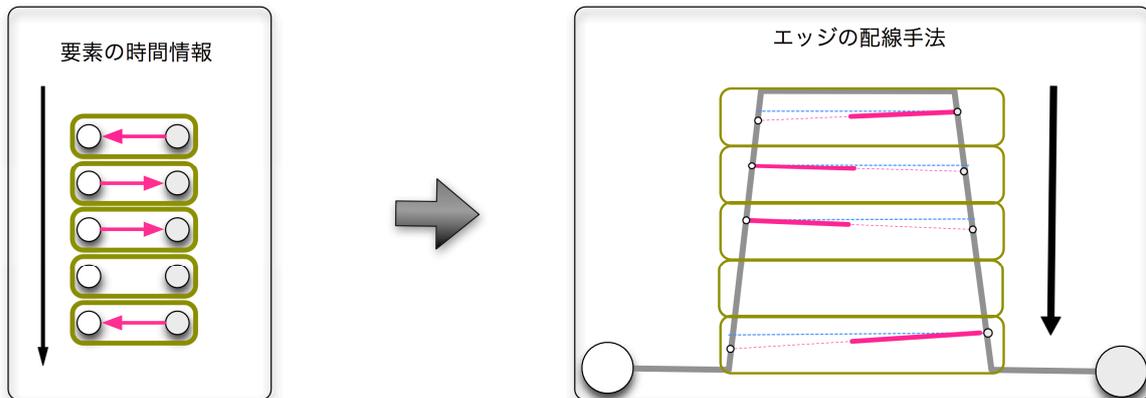


図 4.3: エッジの時間情報の表現

エッジは配線の一部を山型にした線分で表現する. そして, エッジの時間情報は時間軸と対応づけた山型の線分を基に直線で表現する. 図 4.3 に描画手法の概観を示す. 図 4.3 左に入力となるグラフとその時間ステップ (総ステップ数 5) ごとの変化を示す. 各時刻におけるグラフは茶線枠内の白色, 灰色のノードと赤色の有向辺で構成され, 時系列順に並んでいる. 黒線は時間軸を表す.

図 4.3 左で示す変化の流れは以下の通りである. 初期時刻で灰色のノードから白色のノードへの関係が発生している. 次の時刻ではエッジの向きの逆転が発生している. そして更に次の時刻においては, 前の時刻と同様に白色のノードから灰色のノードへの関係が存在する. その次の時刻においては, 関係は削除されている. 続く最新時刻においては, 灰色のノードから白色のノードへの関係が再び発生している.

この入力を基にした表現の概観を図 4.3 右に示す. 淡灰線の折れ線が提案手法を用いて表現されるエッジ自身であり, 赤太線はエッジの時間情報を表す. なお, 茶線枠は図 4.3 左における各時刻におけるグラフとの対応関係を提示するために描画結果と別

に付与した補助的な情報である. 赤色の点線や青色の点線も同様に図 4.4 以降で用いる補助的な情報であり, 描画結果には現れない.

提案手法は図 4.3 のように関係の有無を山型の折れ線で表す. そして, 山型をなす折れ線をなす一部の線分を時間軸に対応づけ, 関係構造の提示と異なる座標系を 2 次元平面上に設ける. 関係における向きの変化などの情報を線分の 2 辺を結ぶように配線した線分で提示する.

エッジの持つ時間情報内の 1 要素の描画手順を追って示す. 各エッジの持つ時間情報に関して順に適用させ, 描画結果を得る. 具体的な描画手順は図 4.4 ~ 4.7 を用いて説明する.

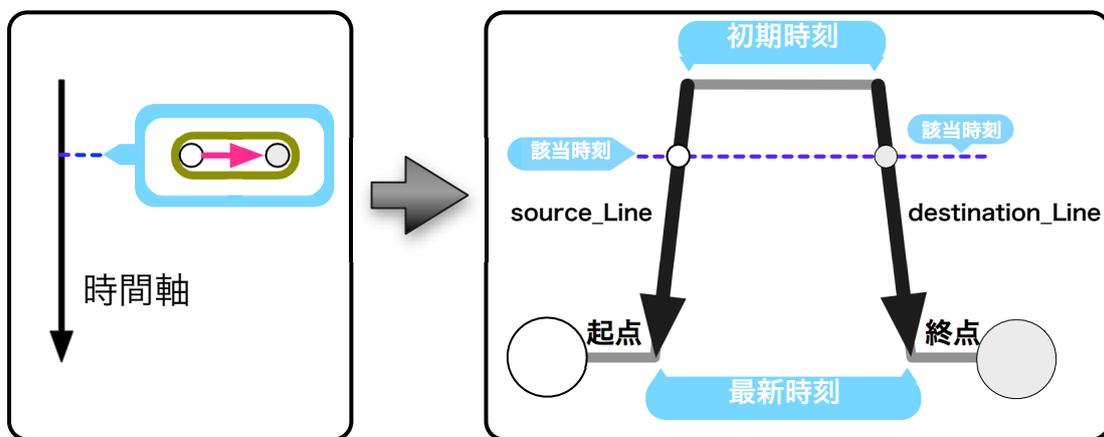


図 4.4: 該当する時刻に対応する位置の決定

図 4.4 に入力となる白色ノードと灰色ノード間の関係を表すエッジの時間情報の 1 要素とその要素の持つ時刻に該当する位置との関係を示す.

図 4.4 左は入力となるエッジの時間情報の一部を表している. 黒線の時間軸に交差する青点線部に該当する時刻に白色ノードから灰色ノードに向けての有向辺が発生したことを表す.

図 4.4 右はエッジの表現において該当する時刻に対応する位置を示した図である. 提案手法においては, 濃灰色の矢印で表現された 2 辺 `source_Line`, `destination_Line` が時間軸に対応づけられる. 辺 `source_Line` は起点側を起点とする順方向のエッジの時間軸に対応し, 時間情報 $t_{\langle \text{白色ノード}, \text{灰色ノード} \rangle}$ の表現に用いる. また, 辺 `destination_Line` は終点側を起点とする逆方向のエッジの時間軸に対応し, 時間情報 $t_{\langle \text{灰色ノード}, \text{白色ノード} \rangle}$ の表現に用いる. いずれも起点, 終点の近傍が最新時刻に対応するよう, 辺の向きを時間軸と対応づける.

時間軸に対応した 2 辺に関して, 時間情報に該当する辺 `source_Line` 上の点 `sp` と辺 `destination_Line` 上の点 `dp` を求め, それぞれの位置を結ぶ線分を求める. 図 4.4 右側の青点線部, 図 4.5 の青点線部がこの線分に該当する. このとき, 時間情報の要素が持つ変化の内容に応じて点 `sp`, `dp` の位置を変化させる (図 4.5). 順方向であれば `dp`, 逆方向であれば `sp` の位置を最新時刻方向 (矢印方向) に向けてずらす. つまり, 有向辺の起点

となる側の端点を固定し, 終点となる側の端点の位置を動かすことにより, 時間情報の要素を表す線分に傾きを持たせる.

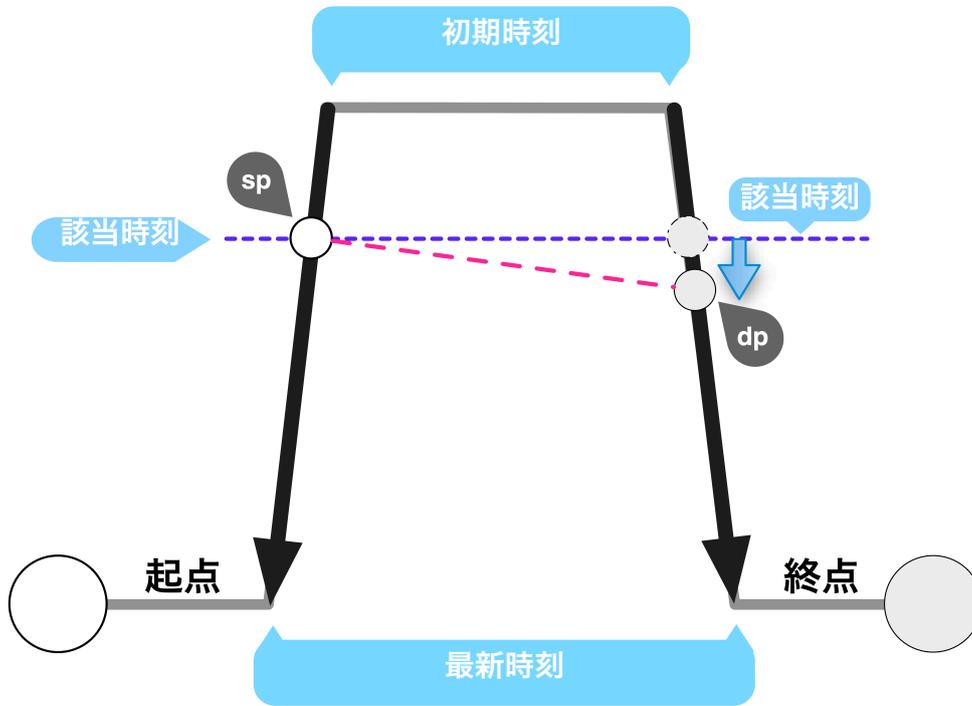


図 4.5: 線分の端点の決定

そして, 先ほど求めた線分に関して, ずらした点を起点にもう一方の点に向けて α ($0 < \alpha \leq 1$) 倍伸ばした線分 (赤太線) を描画する (図 4.6).

これを以てエッジにおける時間情報の 1 要素の描画が完了する. このように線分のみで時間情報を表現する事により, 形状から時刻およびその時刻におけるエッジの向き
の把握が期待できる. 図 4.7 に入力となった時間情報と表現手法との対応を示す.

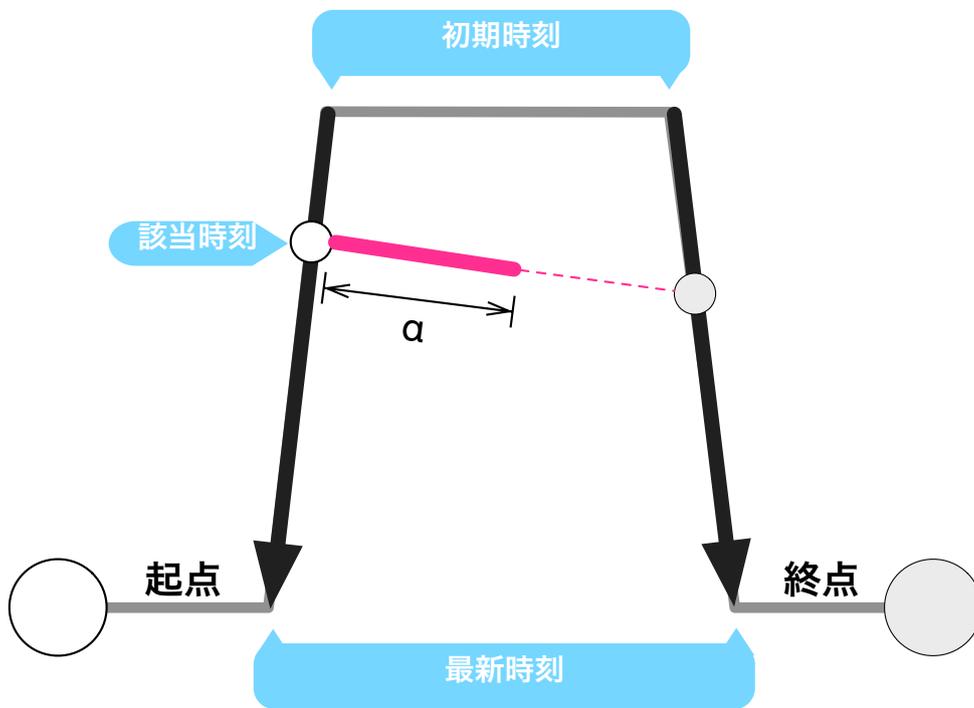


図 4.6: 線分の描画

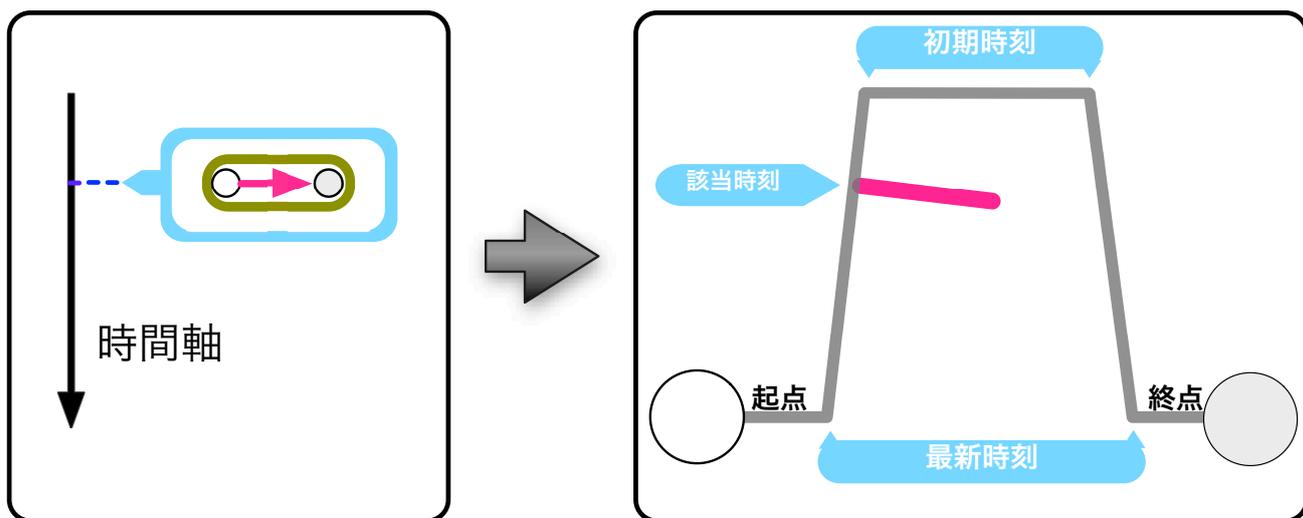


図 4.7: 描画された1本のエッジの時間情報

しかしながら、この手法はエッジの交差に伴い、時間情報同士の交差が頻繁に発生する。そのため、可読性に難があった。可読性の低下に対し、描画位置をいずれかの端点の方向へずらすなどの工夫をとったところ、時間情報同士の交差はある程度の低減が図れた。ただ、描画位置をずらした場合、ずらした側の端点に時間情報が提示され、もう一方の端点の周辺に時間情報が表現されなくなる。そのため、エッジの時間情報を視覚的に探索する機会が増大し、閲覧者の把握が困難になった。そこで、開始位置をずらすだけでなく、エッジの両端に時間情報を提示するよう改善を図った。図 4.8 に改善した表現を示す。

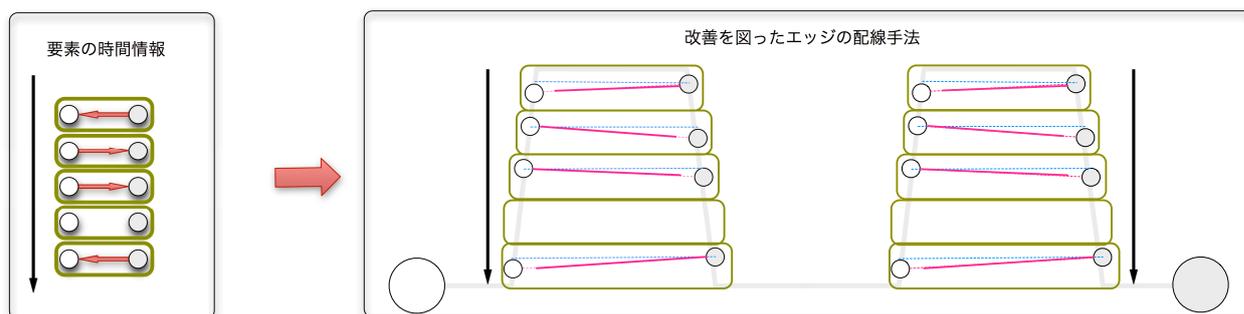


図 4.8: 時間情報の表現の改善

両端に配置された時間情報は互いに同じ形状をとる。ノードに関して、自身に伸びる線分の本数や線分の分布から各エッジにおける時間あたりの接続数の分布が把握できる。また、多数のノードと頻繁に接続しているノードは自身の近傍に線分が多く描画される。そのため、そのようなノードは他のノードより大きく見えるようになる。

エッジ表現より読み取りが期待できるパターン

図に表現手法によって読み取りが期待できるパターンを示す。なおそれぞれの時間情報の表現に関して、向きを強調するために時間情報を表す線分の起点に応じて異なる色を割り当て表示している。表現に関して、双方向から線分の伸びている場合、双方からの接続があったことを示す。また、同じ傾きを持った線分同士の間隔が狭い場合、これはその線分の群が表す時間帯において時系列に関して密な接続があったことを示す。

図上部の場合、複数のパターンが示されている。図上部に関して、図左から右に向かっている線分がまばらに配線されているものの、もう一方方向の線分が存在しない。よってグラフ初期においては、時系列において双方向からの接続が疎である。図中部に関して、図左から右に向かっている線分が隙間なく密に配線されている。これより、グラフ中期においては時系列において一方向からの接続が密であることが読み取れる。また、図下部に関して、図の左右から共に隙間なく多くの線分が配線されている。これより、最新時刻では時系列において双方向からの接続が密になったことが読み取れる。図この

エッジ表現に関して、2 ノード間には3つのパターンがそれぞれ存在することが読み取れる。

図中央のエッジ表現は、図上部から下部まで一方向の線分がまばらに描画されている。特に初期時刻に対応する位置においては線分間の間隔が密であるものの、最新時刻側に向かうに従って、間隔が開いている様子が確認できる。これより、図中央のエッジは時系列において一方向からの接続が発生し、それが徐々に接続頻度が下がっていることパターンを有することが読み取れる。

図下部のエッジ表現は線分の本数が少なく、また、それらの間隔がまばらである様子が確認できる。図下部のエッジは時系列において双方向からの接続が疎であり、ほとんど接続が発生していないパターンを有することが読み取れる。このように、線分の本数およびそれらの間隔の疎密、線分の起点と傾きより、様々なパターンの読み取りが期待できる。

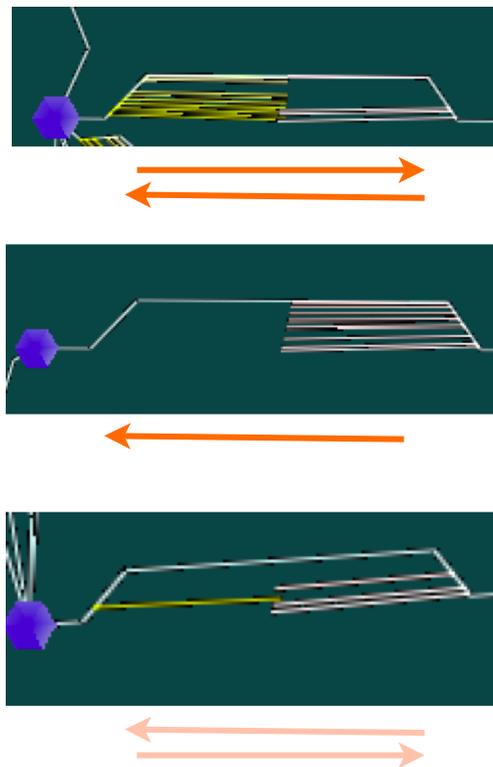


図 4.9: 読み取りが期待できるパターン

また、交差箇所の個数および発生箇所からもパターンを読み取りが期待できる。図 4.10 に一例を示す。図 4.10 に関して、図中央にて双方向から伸びた線分が2カ所交差していることが見て取れる。また、交点を見ると図左から右に向かつての線分が図右から左に向かつての線分の上部に重なっていることがわかる。これより、この2ノード間に関して、図左からの接続に反応して逆方向の接続が発生する、いわゆる実世界における2点間の行き来を表す接続関係のパターンが散発している様子が読み取れる。



図 4.10: ”行き来”が発生したパターン

連結成分の表現手法

連結成分は、構成する要素を完全に包含するバウンディングボックスを描画し、これを提示する。

4.6 時間情報間の距離

本節では、4.4.3 節でのレイアウトに用いる時間情報間の距離について述べる。本論文では、動的時間伸縮法を用いて時間情報間の距離を定める。動的時間伸縮法とは、2つの数値列間の距離を図るための手法である [25]。この手法の特徴として、以下の2点が挙げられる。

- それぞれの数値列に関して、両者に共通する要素が無い場合でも距離を求められる。
- それぞれの数値列が持つ要素の総数が異なる場合でも距離を求められる。

この手法は数値列の要素間の距離の和から数値列間の距離を算出する。

本論文で扱う時間情報は時刻と変化の内容の組を要素とする列である。つまり時間情報もまた数値列として扱うことが出来る。数値列として扱い手法を適用させるにあたり、時刻に関して正規化を行う。総計して得たグラフにおける初期時刻から最新時刻までの時間長を 1 とし、それぞれの時刻を (0,1) の区間内の数値に置き換える。時間情報内の各要素を数値に置き換えて得た数値の列を数値列とし、手法に適用させる。

動的時間伸縮法を用いた、時間情報 x, y における距離関数 gwt は [26] に倣い記述すると以下のように表せる。なお、 $x[n]$ は数値列 x の n 番目の要素を表し、 $x[n, :]$ は数値列 x に関して n 番目の要素から数値列の終端まで順に取り出した数値列を表す。

$$gwt(x, y) = gwt_0(x[0], y[0]) + \min(gwt(x[1, :], y), gwt(x, y[1, :]), gwt(x[1, :], y[1, :])) \quad (4.6)$$

なお、数値列内の要素 a, b 間の距離関数 gwt_0 は以下のように設定した。

$$gwt_0(a, b) = |a - b| \quad (4.7)$$

動的時間伸縮法による時系情報間の距離の算出

動的時間伸縮法による距離の算出は、動的計画法の手法を用いて行われる [26]. $x = \langle 0.0, 0.1, 0.3, 0.9 \rangle, y = \langle 0.0, 0.5, 0.7 \rangle$ が与えられた場合、 $gwt(x, y)$ は表 4.1 のとおりになる.

| $y \backslash x$ | 0.0 | 0.1 | 0.3 | 0.9 |
|------------------|-----|-----|-----|-----|
| 0.0 | 0.0 | 0.1 | 0.3 | 0.9 |
| 0.5 | 0.5 | 0.4 | 0.2 | 0.4 |
| 0.7 | 0.7 | 0.6 | 0.4 | 0.2 |

表 4.1: 差分テーブル

$$\begin{aligned}
 gwt(x, y) &= gwt_0(x[0], y[0]) + (gwt_0(x[0], y[1]) + (gwt_0(x[1], y[2]) + (gwt_0(x[2], y[3])))) \\
 &= 0.0 + 0.1 + 0.2 + 0.2 \\
 &= 0.5
 \end{aligned}$$

それぞれの時間情報中の要素同士の距離を求め、これらを差分テーブル 4.1 と呼ばれるテーブルに挿入する. そして、テーブル左上部を起点とし右下部を終点としたパスにおいて、通過したセルの値の合計を最小とするパスを求める問題として解く.

要素aの時間情報 要素bの時間情報

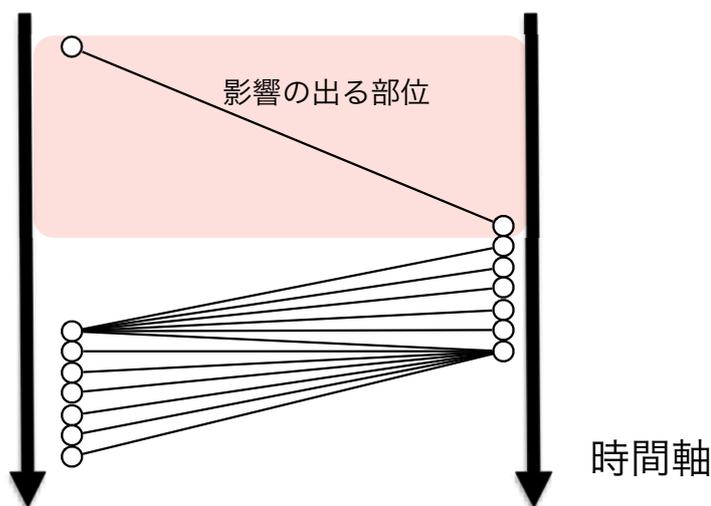


図 4.11: 動的時間伸縮法の算出イメージ

なお、この手法を用いた場合、一方の数値列に関して、他の要素との距離が大きくなる要素が含まれる場合、距離関数はその要素を大きく評価するおそれがある. 図 4.11

に2つの時間情報に対して動的時間伸縮法を適用させた場合のイメージを示す。時間軸上に沿って配置された点は時間情報の要素を表す。そして、それらを結ぶ線分の傾きを数値化した値の総和が数値列間の距離に相当する。図4.11において赤く囲った箇所は要素aが持つ一部の要素が他の要素との距離が大きく離れているため、距離関数がそれらの距離を大きく評価していることを示す。

第5章 実装

提案手法を実装したツールについて述べる.

5.1 ツールの概要

実装したツールは時系列有向グラフに変換できるデータを基に, 提案手法を用いた可視化結果をユーザに提示する. ユーザは可視化結果を閲覧し, インタラクションを通して, データの分析を行う. このツールはビューとコントローラから構成される (図 5.1).

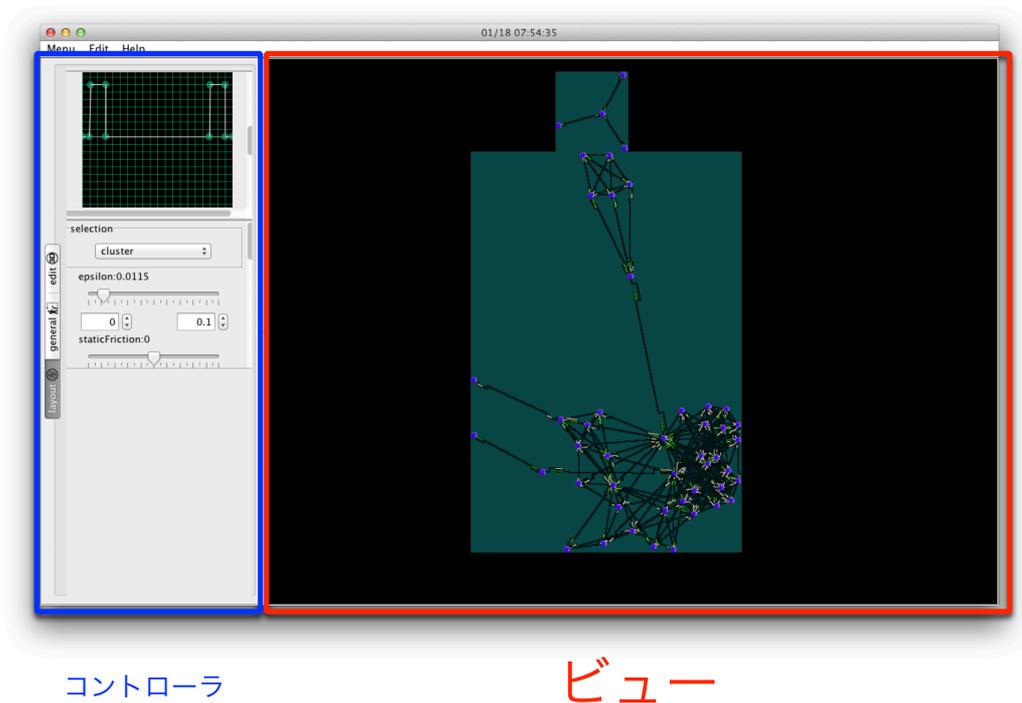


図 5.1: アプリケーションの概観

5.1.1 構成

ツールは Java 6 を開発言語としており、描画においては OpenGL¹を利用している。また、json データの読み込みの関して、パーサとして JSONObject²を利用している。時刻情報を列に持った csv,tsv ファイルないし json ファイルを指定することにより、システムはファイルをパースし、ビューにグラフを提示する。

5.2 機能説明

提示されるビューに関して、ユーザはいくつかのレイアウト、および描画に関するパラメータを調整できる。これにより、ユーザはデータから得たグラフの特徴やユーザの興味に応じて適切なビューを閲覧することが可能となる。

レイアウトパラメータの操作

連結成分やノード間の斥力、エッジのばね係数などグラフレイアウトにおけるレイアウトパラメータを操作できる。また、分析したい情報に合わせて、配置アルゴリズムを変更できる。

ビューパラメータの操作

ツールのビューは OpenGL を用いて 3 次元空間の描画空間を生成し、その上の 2 次元平面上に描かれたビューを提示している。ユーザは 3 次元空間上のカメラ座標を操作し、視点の移動やビューの拡大縮小を行うことが出来る。

ラベル表示の制御

ビュー上のノードを選択することにより、選択した要素のラベルがユーザに提示される。また、コントローラ内のチェックボックスを選択することにより、全ノードのラベルを提示するか選択したノードのラベルのみを提示するかを決定できる。グラフを分析する場面においてラベルは重要な分析のきっかけとなる。しかしながら、グラフの規模が増大すると、要素のラベルが他の描画要素と交差してしまう場合が発生する。この機能はそのような交差を回避するために用意されている。

¹OpenGL : <http://www.opengl.org/>

²<http://json.org/java/>

エッジ描画のパラメータの操作

4節で説明した,エッジの描画に関わるパラメータ,およびエッジの持つ時間情報のパラメータの調整を行うことが出来る. エッジの節点の位置を操作するパネル(図5.2上部)を用いることにより,提示されるエッジの形状(図5.2下部)を変形させることが出来る. 形状を変化させることにより,注目する時間変化を強調させ,ユーザの興味に応じたビューを提示することが可能となる.

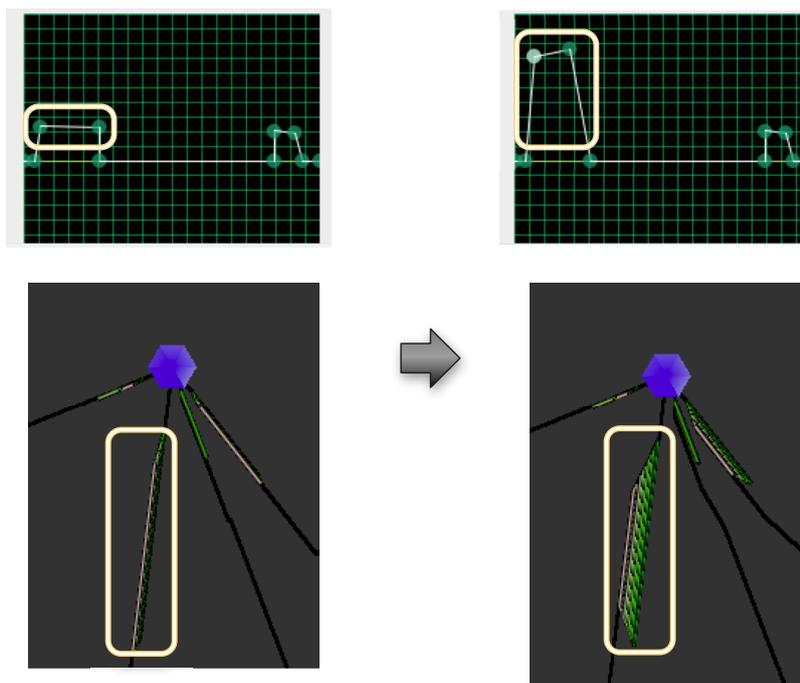


図 5.2: エッジ形状の操作

形状だけでなく,エッジの持つ時間情報に割り当ててる色を操作することも可能である. 初期時刻,最新時刻に対応させる色をエッジの向きごとに指定できる. いずれかのエッジの向きに関して,初期時刻と最新時刻にそれぞれ異なる色を割り当てた場合,その他の時刻に対応する色は2色を線形で補完した色となる. 異なる色を割り当てることにより,形状だけでなく,色情報の分布から変化の発生頻度の推移を把握することが容易となる.

第6章 ケーススタディ

本章では、具体的な使用例を示し、得られた知見とその議論を行う。

6.1 アクセスログの可視化

サイト来訪者のページ遷移の時系列有向グラフを用いた。2009年4月9日より同年5月6日までの履歴を利用した。

アクセスログからのグラフの抽出

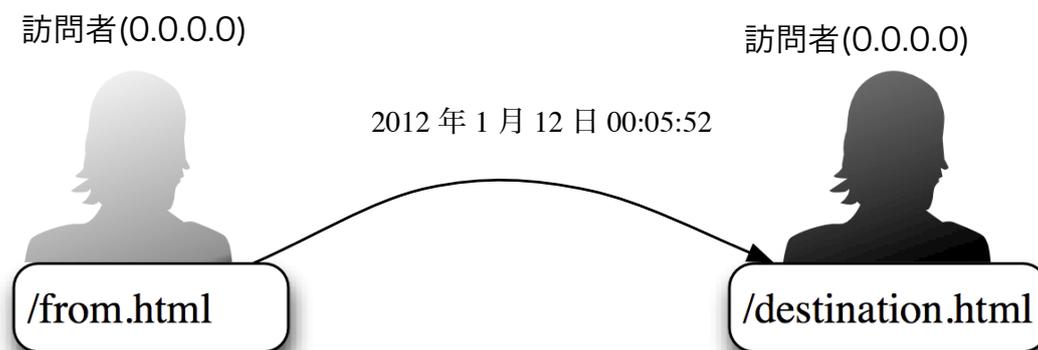


図 6.1: ページ遷移のイメージ

アクセスログに記載されているリクエストヘッダーおよびファイル名の情報とリファラ情報からページ間の遷移を抽出する。

```
0.0.0.0 - - [12/Jan/2011:00:05:52 +0900] "GET /destination.html HTTP/1.1" 200 6556  
"http://webpage/from.html" "Mozilla/5.0 (Windows; U; Windows NT 5.1; ja; rv:1.9.0.8)  
Gecko/2009032609 Firefox/3.0.8 (.NET CLR 3.5.30729)"
```

このようなアクセスログの場合、「訪問者”0.0.0.0”が2012年1月12日 00:05:52に Web ページ”http://webpage/from.html”からサイト内の Web ページ”/destination.html”へ遷移した」という情報を抽出する。

今回は、問い合わせに成功しており、かつリファラ情報、およびリクエストしたファイルが Web ページであったデータのみを選び出し、抽出を行った。

なお、収集期間中に訪問されなかったページは、アクセスログに記載される事はないため可視化結果に現れない。

また、留意すべき点として、「戻る」「進む」などのブラウザのキャッシュを利用した閲覧行動が挙げられる。一般的な Web ブラウザを用いて Web サイトをブラウジングする場合、ブラウザは一度訪れたページをキャッシュに保存する。これは利用者が再度同じページに訪れることを予期し、そのような場合が発生したときページの読み込みを高速化させるための工夫である。キャッシュ内のページを閲覧している場合、サーバ側に特別な工夫が無い限り、ユーザの閲覧行動はアクセスログに記録されない。

アクセスログへの適用結果

Web サイトのアクセスログを参考に、コンテンツの配置について検討することを目的に分析を行った。アクセスログを適用させたところ、図 6.2 を得た

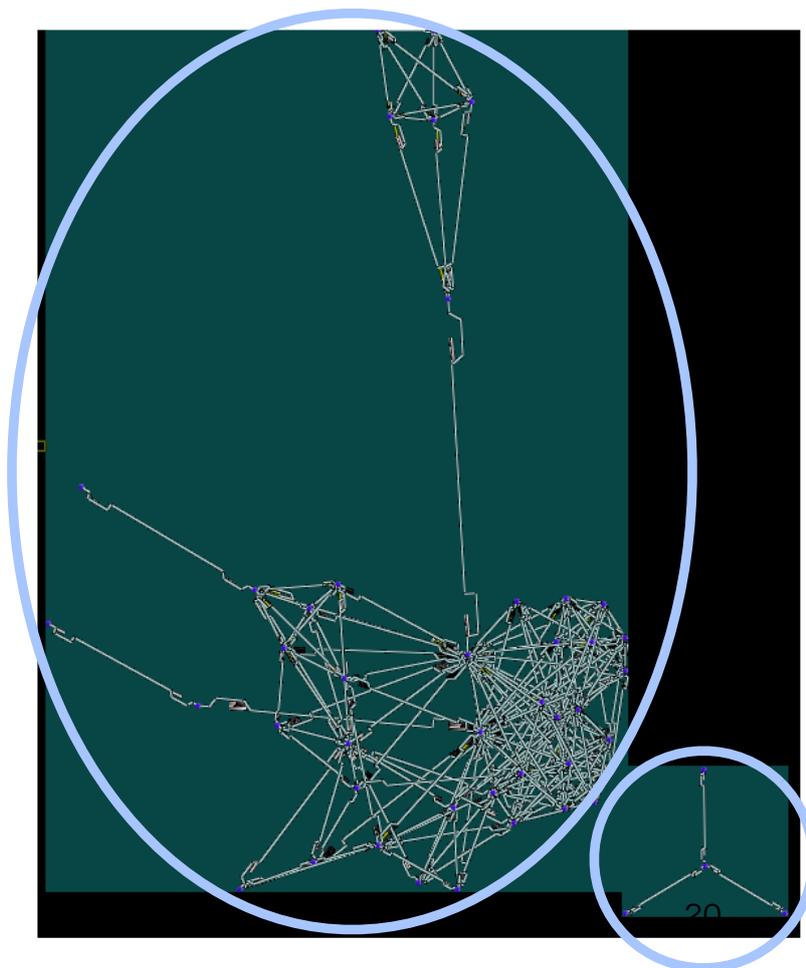


図 6.2: 適用結果

まず、2つの連結成分が存在することが確認できる。ここで、ノード数の少ない連結成

分(図 6.3)に注目する. エッジの時間情報の本数より, この1度 Web ページを通過した

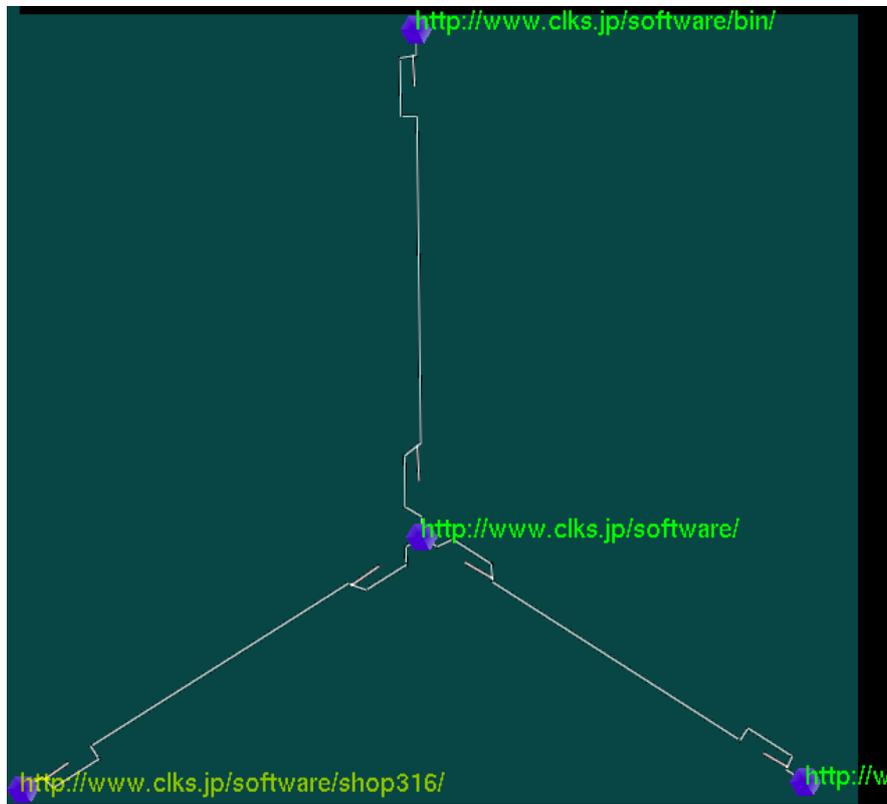


図 6.3: /software/ディレクトリ化の連結成分

連続するアクセスはそれぞれ1度のみ発生したことが分かる. また, エッジの時間情報の位置と形状より, 4月下旬に中央のページに向けてのアクセスが発生したことが分かる. それぞれのノードのラベルを確認したところ, いずれも”/software/”であることが分かった. コンテンツのページであるのにも関わらず連続したアクセスが少ないことより, 「”/software/”ディレクトリ下のページは直帰率が高いのではないか」という次の分析に向けた知見を得た.

続いて, もう一方の連結成分に目を向けた. グラフの形状よりグラフの全要素数に比べて, クリークが大きいことが確認できる. クリークが存在することより, web サイトを巡回する訪問者が存在することが読み取れる. このクリークをなすエッジに関して, 両方向から頻繁にアクセスが存在するのかまたは一方向からのアクセスのみ存在するのかを調べるため, エッジの形状を変化させ, 線分を縮小させる倍率 α を高めた. この操作により図 6.4 を得た. エッジの山型の線分付近にまとまった線分が存在するか否かで判断できる. 多くのノードにおいてエッジの山型の線分付近にまとまった線分が存在することが確認できた. そのようなエッジを持つノードに関して, それぞれのラベルを調べたところ, ”cs001.html”, ”cs002.html”など連番のページを表すノードであった(図 6.6). 連番ページに関して, エッジの時間情報を眺めたところ, 双方向からのエッジが多数出現していることを確認できた. また, 異なる連番に属するページとのエッ

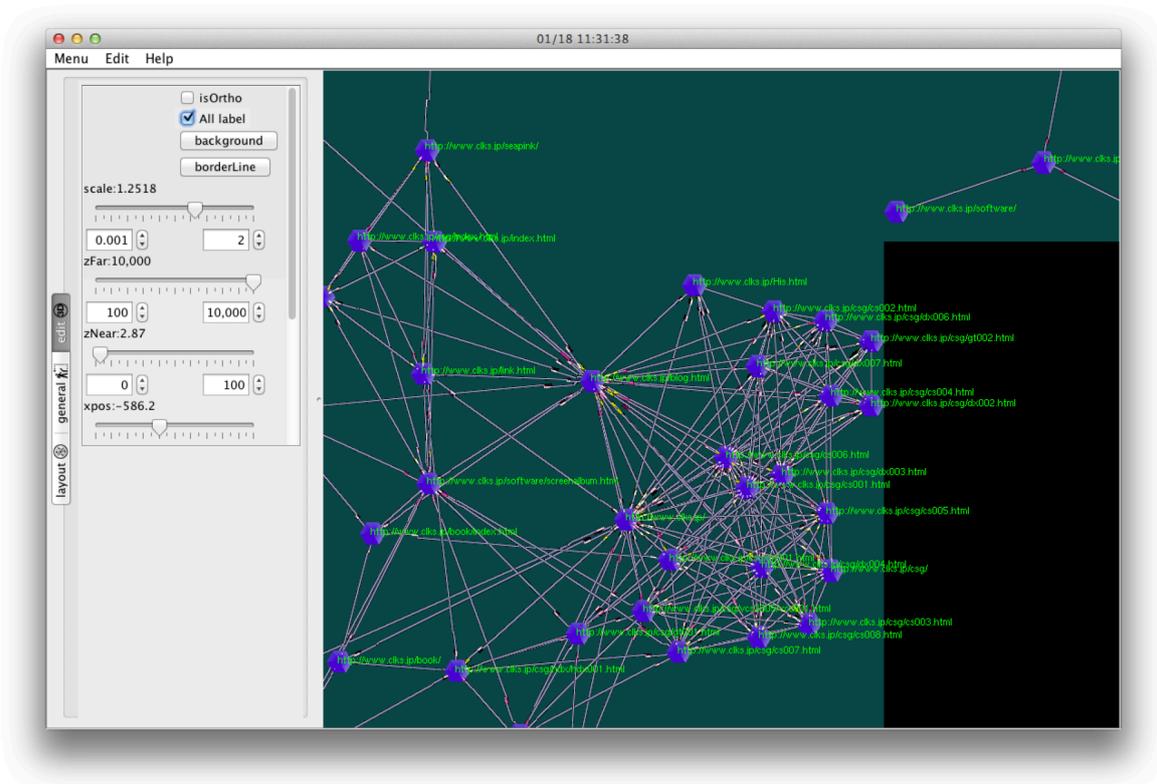


図 6.4: エッジの形状を変化させた図

ジの時間情報は一方向のみの線分で構成されている。親階層のページとのエッジは双方向に線分が存在するものの、線分の本数およびそれらの間隔は連番ページのそれらと比べ、いずれも本数が少なく、また間隔も大きい。これより、同じ連番に属するページ同士に関して、双方向からのアクセスが頻繁に発生しているという情報が読み取れる。

連番ページに何らかの広告を掲載した場合、他の連番ページにアクセスした訪問者はいずれ広告を掲載したページにアクセスする可能性があるとして強く考えられる。よって、「連番ページに特定の広告などを掲載する場合、全てのページにそれを掲載する必要は無いのではないか」という運営上の仮説を得た。

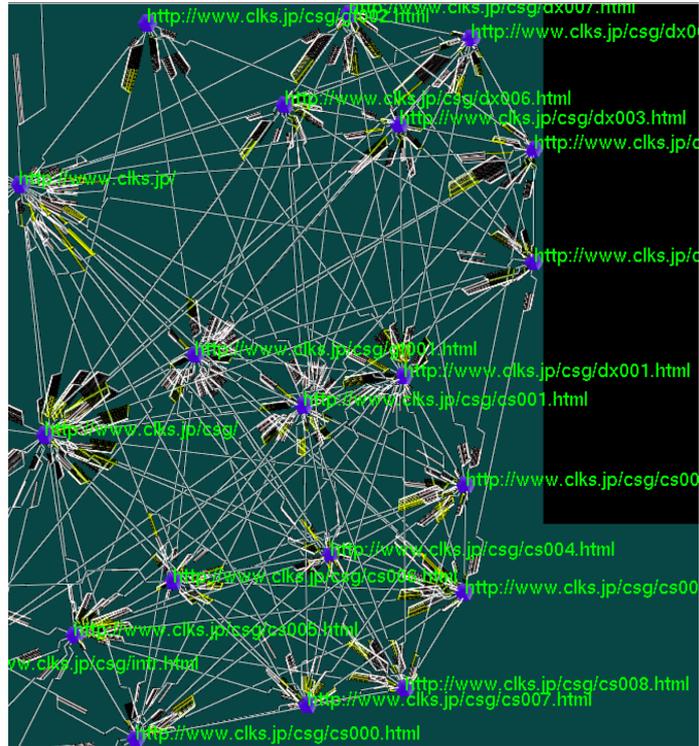


図 6.5: 連番ページ群を囲った図

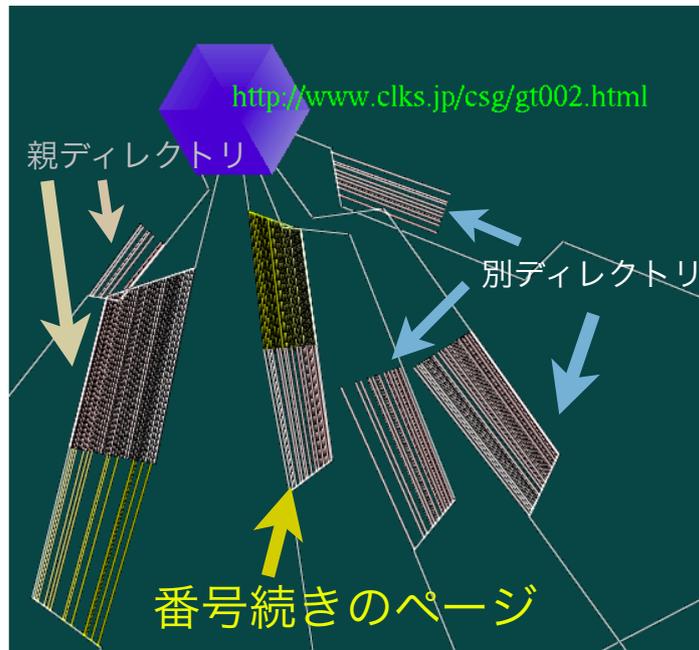


図 6.6: エッジの接続先の属性ごとに異なる線分のパターン

第7章 議論

7.1 考察

7.1.1 局所的な座標系の導入に関して

本論文では、エッジの時間情報を局所的な座標系を導入した2次元平面上で線分を用いて表現した。このように局所的に別の座標系を導入し、関係構造と時間情報を同じ空間に配置することにより、従来の関係構造の提示手法をそのまま利用しつつそれぞれの時間情報を提示することが可能となる。しかしながら、同時刻の変化を見ると言ったようなタスクを提案手法の表現のみで実現することは困難であると考えられる。これは、導入した局所的な座標系においてそれぞれの時間軸が統一されていないため発生する問題である。提案手法を実装したツールを用いてそのようなタスクをこなす場合、時刻の指定などの操作に応じてインタラクティブに一部の描画要素における描画属性を切り替えるなどの処理を加える必要があると考えられる。

7.1.2 時間情報の表現の可読性に関して

エッジの時間情報の表現手法では、線分の開始位置を変化の発生した時刻に対応付け、線分の傾きと長さを用いてエッジの向きを表現する。手法の提案段階においては、エッジの向きの表現に傾きを用いることにより、傾いた線分同士の交差から関係構造の変化パターンの発見が容易になると期待した。しかしながら、表現を実装したツールを用いて手法を適用させたところ、期待した効果が得られなかった。期待した効用に関しては1本の線分において発生しうる交差を1回に収めることによりはじめて得られると考えられる。また、描画においても交差した箇所を別の表現に置き換えるなどの工夫が必要になると考えられる。

その一方で、手法を用いることにより、時間ステップ数が多いグラフにおいても、時系列に関して情報を圧縮させる処理を行うことなく全時刻の変化を提示することが可能となった。全時刻の情報を提示するため、任意の時間帯に関して時系列における疎密が提示可能となり、関係構造の変化するパターンの発見に有用となると考えられる。

しかしながら、エッジごとにそれぞれ異なる座標系を導入したため、エッジ同士の交差が可読性の低下を引き起こしている。これに対し、時間情報をエッジの両端に配置することで時間情報同士の交差の低減を図ったが、同じ情報が図中に重複してしまい、空間あたりの記述できる情報量が低下する問題も新たに生んでいる。

7.1.3 無向辺を含むグラフへの適用に関して

本論文では有向辺に着目し, その変遷を視覚的に提示する手法を提案した. この手法の適用範囲は有向グラフのみならず無向グラフ, 有向辺と無向辺の2種類を含むグラフにおいても適用可能である. 無向辺を含むグラフ表現の例として, UML におけるクラス図などが挙げられる.

有向辺は, 提案手法をそのまま用いて表現し, 無向辺は, 有向辺描画におけるパラメータを変化させることにより実現可能となる. 提案手法において, 有向辺の向きは線分の傾斜と線分の方の端点の位置を用いて表現している. 無向辺の時間情報は, 線分の傾斜を無くし, 線分の両端をエッジを表す折れ線に結びつけるよう描画パラメータを変化させた有向辺の描画手法を用いて表現することが出来る (図 7.1 右). 無向辺を含むグラフへの適用が可能となることにより, より複雑な関係構造の変遷を視覚的に提示することが可能になると考えられる.

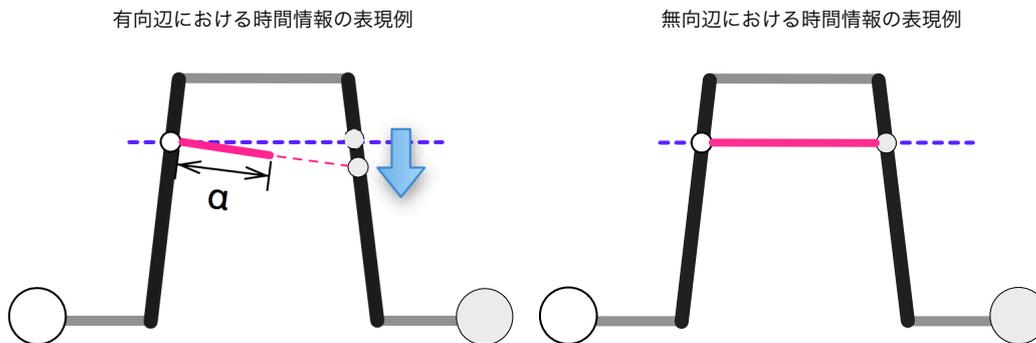


図 7.1: 提案手法を用いた無向辺の描画

7.2 今後の課題

7.2.1 エッジの形状の自動的な調整

本研究においては, エッジの形状を山型としている. この形状としたことにより, 時間情報間の交差が発生した場合にそれらの可読性が大きく減じられる結果を引き起こしている.

これに対し本論文では, エッジの山型となる部分を2つに増やしそれぞれをノードの傍に寄せて配置し交差数の低減をはかった. また, ツールに関して, ユーザはインタラクティブにエッジの形状を変形できるように実装を行った. この副次的な効用として, ユーザは手動で発生した描画要素の回避を図ることが可能となった.

しかしながら, 改善を図った提案手法を用いても時間情報間の交差を完全に回避することは難しい. 今後は, 描画される時間情報の周辺に描画要素が存在するか否かを描画規約に組み込み, 自動的に交差を回避する手法を導入したい.

7.2.2 クリークへの対応

提案した表現はエッジ同士の交差が発生すると可読性が大きく低下する。既存のレイアウト手法を用いることにより、交差数の低減を図ることは出来るが、構成するノード数が5以上のクリークの場合、それらのエッジを完全に回避させるレイアウトは2次元平面上で得られないことは一般に知られている。今後、クリークのレイアウトにおけるエッジ同士の交差を低減させるレイアウト手法や描画手法を導入し、問題の解消を図る必要がある。本手法との併用が可能な手法が実現されれば、時系列においても関係構造においても密なグラフの分析が容易になると考えられる。

7.2.3 動的なデータ追加への対応

本論文で提案する手法は、対象とするデータが予め揃っていることを前提としてレイアウトおよび描画を行う。しかしながら、現実の分析対象となるデータの多くはリアルタイムにその構造を変化させている。実利用を考慮すれば、蓄積されたデータだけでなく、リアルタイムに変化するデータに対応させることは必要となる。

力指向レイアウトにおいては、エッジが1つ追加されるだけでも、全体のレイアウトに大きな影響が及ぶ。そのため、特定の要素に対して加わる力ないし1計算ステップにおける移動量に制限をかけることにより閲覧者のメンタルマップを破壊する事無くレイアウトを更新することが可能になると考えられる。制限のかけ方に関して、要素の追加後にそれ自身へ制限をかける、要素の追加前に対象データのリンク予測を行い、次の時刻にエッジが追加される可能性の高い要素へ制限をかけるなどの手法が考えられる。

7.2.4 サブグラフの同型判定

関係構造の類似性を手がかりに分析を行う場面がしばしば発生する。あるサブグラフと同じように構造が変化したサブグラフを探索するタスクに置き換えることが出来る。計算機が予め任意のサブグラフに関して同型判定を行いその結果を視覚的に提示することにより、同じような変化のパターンを辿った情報の発見が容易になると考えられる。

第8章 結論

本論文では、時系列有向グラフを対象として、時間経過による要素の追加と削除の履歴を視覚的に提示する手法を提案し、これを開発した。

提案手法は配線したエッジの辺上に局所的な座標系を導入し時間軸と対応づけ、要素の追加や削除を線分で表現する可視化手法である。また、提案手法は各要素における変化が発生した時刻を用いて、それぞれの距離を導出し、これをレイアウトに反映させる特徴を持つ。提案した手法を基にビューワを作成した。アクセスログより抽出した訪問者の Web ページの移動から生成される有向グラフを用いてケーススタディを行った。ケーススタディを通して、可視化手法が関係構造の時間的特徴の発見に有用であることを示した。提案手法により、エッジの向きの変化などを視覚的に提示でき、これまで困難であった有向グラフが変化する過程を分析することが可能になった。

謝辞

本論文を執筆するにあたり 三末 和男先生、田中二郎先生、高橋伸先生、志築文太郎先生より丁寧な指導、貴重なご意見をいただきました。心より感謝申し上げます。また、筑波大学システム情報工学研究科インタラクティブプログラミング研究室の方々には、ゼミなどを通じ多くの意見をいただきました。深く感謝いたします。最後に、物心両面に渡り長き学生生活を支えてくださった家族、大学院での生活を実りあるものにしてくれた友人、そしてお世話になった全ての方々に心より感謝いたします。

参考文献

- [1] 出原栄一. 図の体系-図的思考とその表現. 日科技連, 1986.
- [2] 杉山公造. グラフ自動描画法とその応用. 計測自動制御学会, 1993.
- [3] Duncan J. Watts. スモールワールド - ネットワークの構造とダイナミクス. 東京電機大学出版局, 2006.
- [4] Noah Iliinsky Julie Steele. ビューティフルビジュアライゼーション. オライリージャパン, 2011.
- [5] 大澤幸生. チャンス発見の情報技術 - ポストデータマイニング時代の意志決定支援. 東京電機大学出版局, 2003.
- [6] James Moody, Daniel McFarland, SkyeBender - deMoll. Dynamic network visualization. *American Journal of Sociology*, Vol. 110, No. 4, pp. 1206–1241, 2006.
- [7] Lucian Voinea, Alex Telea, and Jarke J. van Wijk. Cvsscan: visualization of code evolution. In *Proceedings of the 2005 ACM symposium on Software visualization*, pp. 47–56, 2005.
- [8] 鈴木祐太, 古川園智樹, 青山希, 井庭崇. 動的ネットワークの可視化ツールの構築. 情報処理学会ネットワーク生態学シンポジウム, 2006.
- [9] 豊田正史. インタラクティブな動的グラフィックレイアウト手法を用いたウェブグラフ発展過程の可視化. 第14回インタラクティブシステムとソフトウェアに関するワークショップ, pp. 143–144, 2006.
- [10] Nagayoshi Nakazono, Kazuo Misue, and Jiro Tanaka. Nel2: network drawing tool for handling layered structured network diagram. In *Proceedings of the 2006 Asia-Pacific Symposium on Information Visualisation - Volume 60*, 2006.
- [11] Cesim Erten, Stephen G. Kobourov, Vu Le, and Armand Navabi. *Simultaneous Graph Drawing: Layout Algorithms and Visualization Schemes*. Springer, 2004.
- [12] 豊田正史, 喜連川優. Webrelievo: ウェブにおけるリンク構造の発展過程解析システム. 第12回インタラクティブシステムとソフトウェアに関するワークショップ, 2004.

- [13] Ed H. Chi, James Pitkow, Jock Mackinlay, Peter Pirolli, Rich Gossweiler, and Stuart K. Card. Visualizing the evolution of web ecologies. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 400–407, 1998.
- [14] 伊藤正彦, 豊田正史, 喜連川優. Timeslices: 時系列情報をもつウェブグラフの対話的3次元可視化. 第17回インタラクティブシステムとソフトウェアに関するワークショップ, pp. 141–142, 2009.
- [15] 石原正樹. 動的ネットワークの成長過程と差分の可視化手法. 筑波大学大学院博士課程システム情報工学研究科修士論文, 2007.
- [16] Chaomei Chen. Citespace ii: Detecting and visualizing emerging trends and transient patterns in scientific literature. *J. Am. Soc. Inf. Sci. Technol.*, Vol. 57, No. 19, pp. 359–377, 2006.
- [17] Yarden Livnat, Jim Agutter, Shaun Moon, and Stefano Foresti. Visual correlation for situational awareness. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, pp. 13–. IEEE Computer Society, 2005.
- [18] 伊藤正彦, 赤石美奈. 3次元可視化による史料データにおける人間関係構造変化の俯瞰. 第82回人工知能基本問題研究会, pp. 31–36, 2011.
- [19] Steffen Hadlak, Hans-Jorg Schulz, and Heidrun Schumann. In situ exploration of large dynamic networks. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 17, No. 12, pp. 2334–2343, 2011.
- [20] 伊藤隆朗, 三末和男, 田中二郎. タワーマップ:2部グラフ構造と量的情報を同時提示する3次元可視化手法. 情報処理学会第72回大会, 2010.
- [21] 杉山浩平, 本田治, 大崎博之, 今瀬真. ネットワーク分析手法による日本企業間の取引関係ネットワークの構造分析. 日本社会情報学会誌, Vol. 11, pp. 45–56, 2007.
- [22] 三末和男. ネットワークの可視化技術-大規模ネットワークと動的ネットワークへの挑戦. 電子情報通信学会, Vol. 92, No. 2, pp. 112–117, 2009.
- [23] Peter Eades. A heuristic for graph drawing. *Congressus Numerantium*, Vol. 42, pp. 149–160, 1984.
- [24] 斉藤和佳子. Lmntal階層グラフの可視化手法の提案と実装. 早稲田大学大学院基幹理工学研究科修士論文, 2009.
- [25] J. B. Kruskal and M. Liberman. The symmetric time warping algorithm:from continuous to discrete. *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison*, pp. 125–161, 1983.

- [26] 吉川昂伯. 長大な時系列データの類似検索手法に関する研究. 筑波大学大学院 博士課程システム情報工学研究科修士論文, 2007.