

視覚的理解を促すグラフアルゴリズム・デモシステムの開発

Development of Graph Algorithm Demo System for Visual Understanding

稲垣 宏†
Hiroshi Inagaki

1. はじめに

コンピュータ・エンジニアを目指す学生にとって、「アルゴリズムとデータ構造」の知識は必須である。しかし、教科書に沿って、そこに取り上げられている技法を次々と説明していくだけでは、学生の興味を引き付けるのはむずかしい。へたをすると、それらの技法を表面的に暗記するだけになってしまう。そして、各技法の本質的な考え方がわからないまま、「アルゴリズムの授業はむずかしくてつまらない」という印象ばかりが残ってしまう。

本研究では、グラフアルゴリズムを対象とし、アルゴリズムの振る舞いを視覚的に示すことができるデモシステムの開発を行なっている^[1]。このシステムを利用することで、授業において取り上げたグラフアルゴリズムをその場で実行させ、処理過程や出力結果を見ながら、そのアルゴリズムの特徴や優位性について、インパクトのあるわかりやすい説明を行なうことができる。

2. グラフアルゴリズム・デモシステムの概要

2.1 システムの特長

これまでにも、グラフアルゴリズムを視覚化するプログラムがいくつも開発され、公開されている^{[2][3]}。しかし、入力となるグラフを柔軟に生成できない、各種グラフアルゴリズムを統一的に操作できない、表示の制御が十分にできないなど、授業の中での利用を考えた場合には使いづらい面がある。そこで本研究では、実際の授業の中で幅広く利用してもらうことを目指し、多機能で使い勝手のよいデモシステムの構築を行なっている。以下に、その特長を挙げる。

(1) 入力用グラフの規模をユーザが自由に設定できる(乱数を使って自動生成される)。このため、小規模なグラフを使った基本動作の確認から、大規模なグラフを使った処理過程のアニメーション表示まで、広い用途に利用することができる。グラフアルゴリズムの巧妙さを実感するには、大規模な入力グラフに対する振る舞いを視覚化するのが効果的である。

(2) 入力用グラフとしては、一般的なグラフ(有向・無向・重みあり・重みなし)以外に、完全二分木・完全グラフ・一般の二部グラフ・完全二部グラフといった特別なグラフを生成することもできる。

(3) グラフエディタを備えており、入力グラフをユーザが手動で作成したり、自動生成されたグラフの一部を変更したりといった作業ができる。さらに、編集作業を容易にするためのズーム機能も備えている。

(4) 入力用グラフにおけるノード情報やブランチ情報の表示 ON/OFF を必要に応じて切り替えることができる。これ

により、アルゴリズムの全体的な振る舞いを見せたいとき、あるいは、その時点でのノードやブランチの情報を詳しく調べたいとき、といった様々な状況において使い勝手のよいシステムになっている。

(5) アニメーション表示のコントロール機能がある。この機能を使うことで、アニメーション表示の速度を変えたり、一時停止・再開を繰り返したり、ステップ実行したりすることができる。そのため、口頭による説明とうまく同期をとりながら処理過程のアニメーションを見せることが可能である。

(6) 代表的なグラフアルゴリズムの多くを実装しており、継続的に利用することができる。また、各種アルゴリズムを統一した操作で扱うことができるため、教員にとって使いやすいだけでなく、見た目の統一感から、学生にとってもわかりやすい。

(7) 未実装のグラフアルゴリズムを比較的容易に組み込むことができるように設計されており、ユーザが新しいアルゴリズムを追加することも可能である。

2.2 実装済みグラフアルゴリズム

実装済みのグラフアルゴリズムの一覧を以下に示す。基本的なグラフアルゴリズムはほぼ揃っており、実際の授業において有効に利用できるものと考えている。

- ・無向グラフの深さ優先探索
- ・無向グラフの幅優先探索
- ・有向グラフの深さ優先探索
- ・最短経路問題 (Dijkstra 法)
- ・最小全域木問題 (Prim 法)
- ・最小全域木問題 (Kruskal 法)
- ・ネットワーク流問題 (Ford-Fulkerson 法)
- ・関節点の検出
- ・有向グラフの強連結成分分解
- ・最大マッチング問題
- ・割り当て問題

3. グラフアルゴリズム・デモシステムの実行例

本システムは、Java アプレットとして構築されているため、特別なインストール作業は不要で、だれでも気軽に利用することができる。

図1に示すのは、大規模な有向グラフにおける深さ優先探索アルゴリズムの実行例(処理過程アニメーションの一場面)である。アニメーション中で使われているノード色の意味などは、下部のコメント領域に表示される。

入力用グラフの生成は、乱数による自動生成以外に、グラフエディタ機能を使って、ユーザが手動で作成・編集することもできる。グラフエディタを使った編集作業の様子を図2に示す。

以降に、実装済みのグラフアルゴリズムの実行例をいくつか示す。図3は、大規模グラフに対して導出した最小全

† 豊田工業高等専門学校, Toyota College of Technology

域木である。なお、グラフの一部を拡大表示する機能および個々のノードやブランチがもっている情報を表示する機能を使うことで、グラフの一部分の詳細なチェックもできる(図4)。

図5には、Ford-Fulkerson法を利用した最大マッチングの導出過程アニメーションの一場面を示す。さらに、図6には、大規模なグラフに対して導出した関節点の様子を示す。こうしたデモを見せることで、教科書の説明で用いられる小規模なグラフでは実感できない、そのアルゴリズムの有用性や巧妙さといったものを伝えることができると考えている。

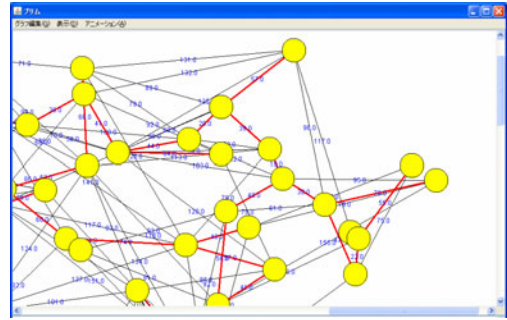


図4. ズーム機能・データ表示機能

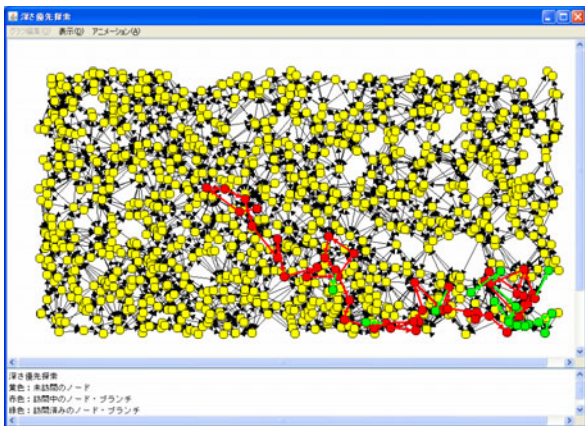


図1. 有向グラフの深さ優先探索(処理過程)

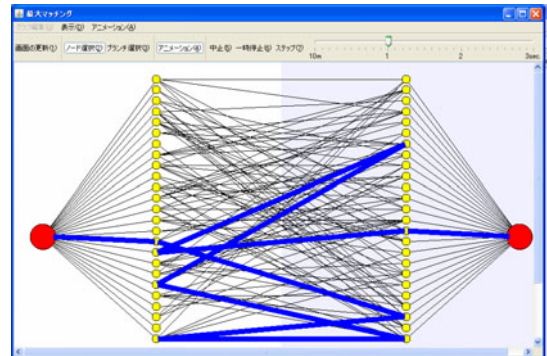


図5. 最大マッチング(処理過程)

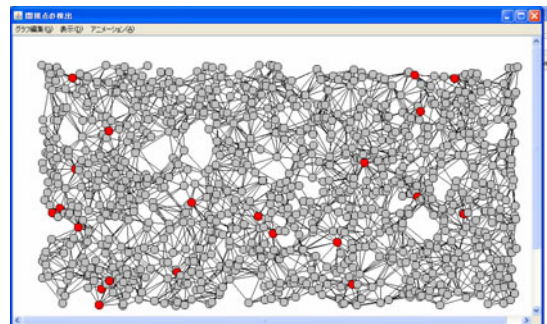


図6. 求めた関節点

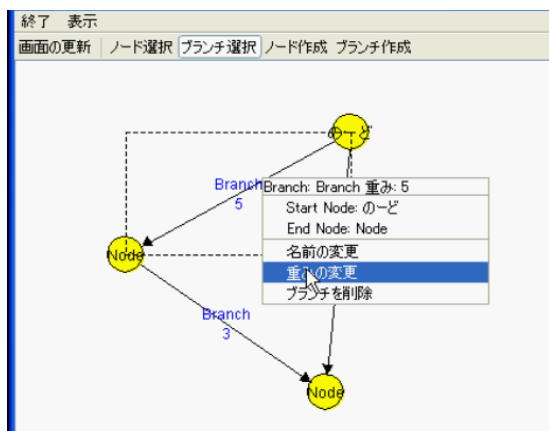


図2. グラフエディタによる編集作業

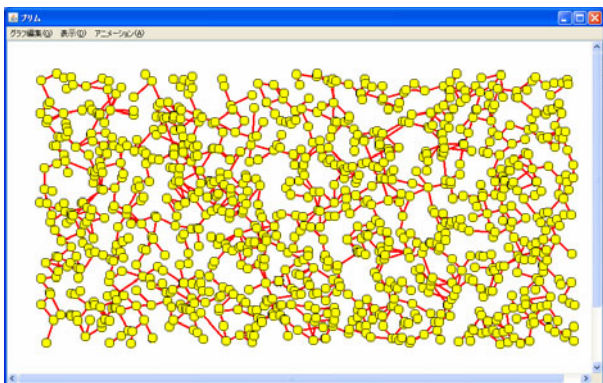


図3. 求めた最小全域木

4. おわりに

本稿では、グラフアルゴリズムの授業支援を目指して開発を進めてきたグラフアルゴリズム・デモシステムについて、その特長や実行の様子を紹介した。授業の中で、口頭や板書による説明の補助教材として利用することで、インパクトのあるわかりやすい説明が可能になると考えている。

また、本システムは、自学自習用のe-ラーニング教材として活用することも視野に入れて設計を行なっている。現在、本システムを利用したグラフアルゴリズム学習サイトを構築中である。

参考文献

- [1] 内藤智唯, 稲垣 宏: グラフアルゴリズムを対象とした教育支援システムの開発, 情報処理学会第68回全国大会, 2006年.
- [2] アルゴリズムデータベース, <http://www-or.amp.i.kyoto-u.ac.jp/algo-eng/db/>
- [3] 陳 致中: Program Demo, http://rnc.r.dendai.ac.jp/chen_lab/content_program.html