

WWW 上におけるアルゴリズムアニメーションシステムの構築*

3 L-7

関下 浩正†

池田 諭‡

中森 眞理雄§

東京農工大学大学院 工学研究科¶

1 はじめに

アルゴリズムアニメーション (AA) は、アルゴリズムの振舞いを視覚的に表現する手法で、アルゴリズムの動作の理解や確認を容易にすることができる。このことから、アルゴリズムの教育やデモンストレーションに対して有効であると言える。これまで、さまざまなアルゴリズムアニメーションシステムが設計されてきたが、システムの記述言語、アルゴリズムの記述言語などにより、動作環境が制限され、使用目的が限られてきた。

筆者らが設計したシステムは、WWW 上で動作し、さまざまなアルゴリズムの実装を容易に行なうことができるので、インターネットに接続された環境であれば、特定の OS やプログラミング言語に依存せずに、ブラウザだけで見ることが可能になる。そのため、教育やデモンストレーションで利用するためのアニメーションの作成に有効である。

2 システムの基本設計

本システムは、アルゴリズムに注釈をつけて、それに対応するアニメーション処理を逐次的に実行する。アルゴリズムの実行部分 (アルゴリズム実行部) とアニメーションの表示部分 (アニメーション表示部) の間はデータ (アニメーションデータ) によって処理の受け渡しを行なう。

システムの構成を図 1 に示す。図 1 の左側の部分がアルゴリズムの実行に関する部分で、右側の部分がアニメーションの実行に関する部分である。それぞれ独立したアプレットであり、同一ページ上で実行され、アルゴリズムが実行されるとアニメーション関数からアニメーション用のデータが出力され、アプレット間

の通信によりアニメーション制御部にデータが送られる。アニメーション実行部はこのデータにしたがって逐次的にアニメーションを実行する。

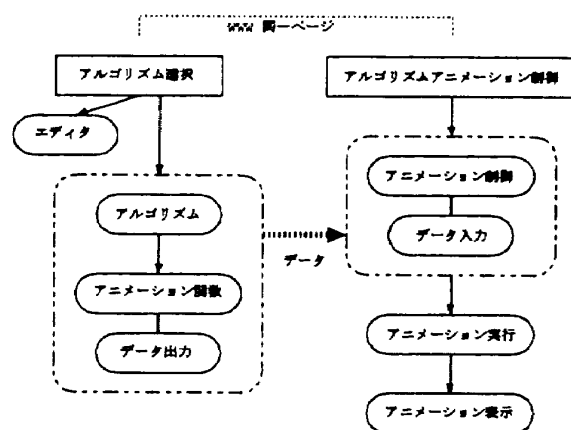


図 1: システムの構成

本システムでは、複数のアルゴリズムを同時にいくつかのウィンドウで表示する Multiple Algorithms (MA) と、あるアルゴリズムに対して表現の異なるアニメーションを同時に表示する Multiple Views (MV) を Java のスレッド機能によって実現している。

3 AA の実装方法

アニメーション設計者は AA の表現方法に専念し、MA や MV を実現するための労力は使うべきではない。本システムでは、MA や MV について考えることなくアニメーション設計を行なうことができる。

アニメーション設計者は、まず具体的なアニメーションを設計し、Java でプログラムを記述する。これには Java の知識があれば、作業は容易である。次に、アニメーションの初期状態を指定するデータと、アルゴリズムのイベントを表すデータを設計し、これらのデータをアルゴリズムのプログラムから出力するためのアニメーション関数を記述する。最後に、設計

*Algorithm Animation System on WWW

†Hiromasa Sekishita

‡Satoshi Ikeda

§Mario Nakamori

¶Tokyo A&T University

したデータとアニメーションの対応をアニメーション実行部に記述する。

これらの作業を終えていけば、アルゴリズムをアニメーション化するには、適切なアニメーション関数をアルゴリズムのイベントとしてプログラムに挿入するだけである。

3.1 ソート（整列）の AA の実装例

ソート（整列）に関する AA の実装例を説明する。ソートでは、並べ替えられる要素の値を高さとした長方形で表し、比較対象の要素を強調して表示する。また、要素の交換が行なわれる場合には交換される様子をわかりやすく連続的に表示する。

図 2 にアニメーション関数を挿入したプログラムを示す。プログラム中の ANmark, ANcomp, ANswap がアニメーション関数であり、ANmark が要素に印をつける、ANcomp が要素の比較、ANswap が要素の交換を意味している。同様にして、挿入ソート、バブルソート、クイックソートなども簡単にアニメーション化することができる。

```
for (i=0; i<N-1; i++) {
  min=a[i]; minI=i; sort.ANmark(i,i);
  for (j=i+1; j<N; j++) {
    sort.ANcomp(j);
    if (a[j]<min) {
      sort.ANmark(i,j); min=a[j]; minI=j;
    }
  }
  sort.ANswap(i,minI); a[minI]=a[i]; a[i]=min;
}
```

図 2: プログラム例（選択ソート）

4 システムの機能

実行画面を図 3 に示す。ユーザは最初にアニメーション制御部で、アニメーションを表示させるためのウィンドウを開く。次に、アルゴリズム選択部で、登録されているアルゴリズムの中から実行させたいアルゴリズムを選択すると、そのアルゴリズムに応じて、アルゴリズムに対する入力データを生成するためのエディタが表示される。エディタでデータを設定し、もう一度アルゴリズムを選択すると、ウィンドウにアルゴリズムの初期状態が表示される。アニメーションの開始や停止、再開、ステップ単位の実行、アニメーション速度の調整などはアニメーション制御部から行なう。また、アルゴリズムに対して異なった側面を表

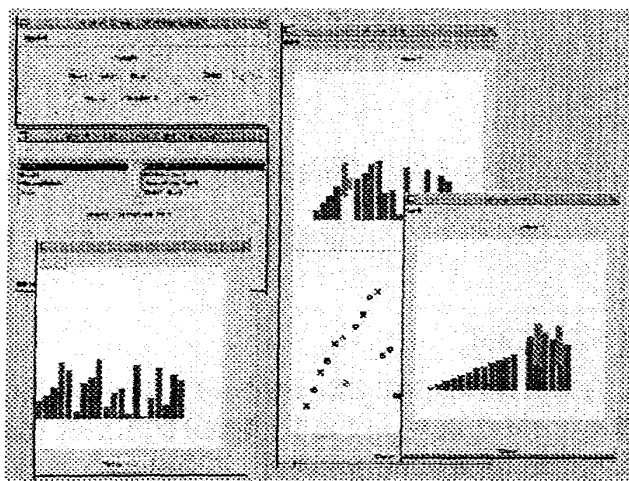


図 3: システムの実行画面（appletviewer で実行）

現する View の選択は、各ウィンドウの中で行なうことができる。

5 おわりに

本稿では、WWW 上における AA システムについて述べた。WWW 上で動作するシステムを実現したことによって、アルゴリズムの教育などにおいて利用価値が高まると考えられる。また、AA の実装に関して、MA や MV について考えることなく、容易にさまざまな AA を実現できる。

今後の課題は、描画する図形やアニメーションの動作のオブジェクト化、GUI によるアルゴリズムとアニメーションのマッピングなどによって、アニメーション化の作業をより軽減することである。

参考文献

- [1] G.-C.Roman et al.: "A Taxonomy of Program Visualization Systems", Computer, Vol.24, No.12, pp.11-24, 1993.
- [2] 曾田忠之他: "アルゴリズム研究のための環境におけるアルゴリズムアニメーション", 情報研報 AL-93-32, pp.73-80, 1993.
- [3] 井上勝行, 魚井宏高, 首藤勝: "能動的学習を支援するためのアルゴリズムアニメーションの拡張法に関する研究", 情報研報 97-CE-43, pp.9-15, 1997.