

3C-5

科学技術論文のための グラフ作成ツールの設計と実現

杉原ミホ¹, 下村秀樹², 並木美太郎², 高橋延匡²¹日立印刷株式会社, ²東京農工大学

1. はじめに

多量の数値データを解析する時、数値だけでは理解しにくい。そこで、データの整理、可視化の手段としてグラフの作成が必要となる。また、科学技術文書の作成において、数式で表されるようなグラフも書く必要がある。当研究室においてもグラフの需要は高い。

そこで、グラフ作成のための専用の言語（グラフ記述言語）を定義し、統合的な文書オブジェクトを出力するための文書整形ソフトウェア「淨書」のグラフ出力機能として、グラフ・文書の一体管理を実現する。本報告では、グラフ記述言語について述べる。

2. 設計方針

2.1 現状の問題点

従来のワードプロセッサなどのグラフ機能では、数式をグラフ化する機能がないため関数グラフを作成するのは困難である。また、多量のデータの入力に手間がかかる。

いくつかの統計処理ソフトウェアでは、機能は豊富であるが、文書との一体管理に問題がある。文書中にグラフを入れる場合は、手書き、または別のシステムで作成し、後から切り張り貼りをしなければならないというものが現状である。

2.2 入力方式

現状の問題を解決するため、グラフ作成ツールの入力として、テキストファイルのフォーマットを定義した。

実験結果をこのフォーマットで計算機から直接出力することにより、多量のデータ入力という手間を省くことができる。また、マウスやタブレットを使用した専用エディタの出力としても可能である。

3. 機能と特徴

グラフ記述言語は、グラフの形式と、データを分離して記述できるという特徴を持っている。データや形式を差し替えることにより、多様なグラフを作成することができる。

また、初版の機能として、次の二つを用意した。

(1) 折れ線グラフ

折れ線の形式として、データのプロットと線によるデータの連結の機能。

(2) 数式の値のプロット

式を記述して、グラフに数式の値をプロットする機能。

4. 設計

4.1 グラフ記述言語の定義

グラフの基本要素を、(1)グラフを書く領域、(2)グラフの形式、(3)データ とし、グラフ記述言語を次のように定義した。

(1) グラフを書く領域の指定

<領域> ::= (領域 <移動量指定>

<縮小率指定>

)

<移動量指定> ::= (移動量 <x方向移動量>

<y方向移動量>

)

<x方向移動量> ::= <数値> {<単位>}

<y方向移動量> ::= <数値> {<単位>}

<縮小率指定> ::= (縮小率 <x方向縮小率>

<y方向縮小率>

)

<x方向縮小率> ::= <数値> {<単位>}

<y方向縮小率> ::= <数値> {<単位>}

<単位> ::= % | mm | cm

ユーザ指定、または上位システムである「淨書」が決定した領域の原点、大きさを基準として原点移動量とサイズ縮小率を指定する。ここで決定する領域は、グラフ全体を描画する領域である。

グラフを書く領域は、ツールの外から与えられた領域の大きさに対して相対的な指定を行なって決定する。このため、紙面領域の大きさの変更に対し、グラフのバランスを保ったまま拡大、縮小ができる。

一度作成したテキストファイルを用い、位置や大きさを変更して、別の文書に出力利用することが可能である。切り貼り作業を解消するだけでなく、拡大、縮小コピーの手間もなくなる。

A Design and Implementation of a Graph Drawing Tool for Scientific Documents

Miho SUGIHARA¹, Hideki SIMOMURA², Mitarou NAMIKI², Nobumasa TAKAHASHI²

¹HITACHI PRINTING CO., LTD, ²Tokyo University of Agriculture and Technology

(2) グラフの形式の指定

```

<形式> ::= ( 形式 {<領域>} )
              {<形式指定>} +
            )

<形式指定> ::= <軸> | <データ表現>
                  | <連結線> | <数式>

ここで指定する領域は、実際にデータが描画される領域（軸の内側）である。(1)で決定した領域を基準に原点移動量とサイズ縮小率を指定する。

<軸> ::= (軸 {<軸指定>} +)
<軸指定> ::= <線> | <軸名> | <目盛り>
<線>では、軸線の修飾（線種、線幅）を指定する。
<軸名>は、文字列実体、文字列の向き、文字の大きさなどを指定する。この時、文字の大きさは文書本文の級数を基準とし、+2級、-2級と符号を付けることにより相対的な指定も可能である。符号を付けない場合は絶対指定となる。<目盛り>では、目盛り範囲、目盛り幅を指定する。

<データ表現> ::= (データ表現
                      <データ表現指定>
                    )
<データ表現指定> ::= <マーク>
<連結線> ::= (連結線 <連結線指定>)

マークの種類、大きさの指定、連結線の修飾を指定する。データのプロットとマークの連結により折れ線グラフの形式となる。

<数式> ::= (数式 <式>)

数式で扱える演算記号として、加減乗除 + - / * を用意した。このほかに sin, cos, log, exp が使用可能である。<式>の記述方法は、次節で例を示す。

```

(3) データの指定

```

<データ> ::= ( データ {<形式>}
                  <データ実体>
                )
<データ実体> ::= (データ実体
                  {<データ実体指定>} +
                )

<データ実体指定> ::= ( {<要素>} +)
<要素> ::= <文字列> | <数値>

```

グラフの形式には、プロット点の種類のようにデータごとに指定したいものがある。そこで、データの中にもグラフの形式を記述できるようにした。これにより、同一領域上に異なったグラフを重ね合わせることが可能である。

さらに、コメント開始記号として“;”を定義している。この記号から改行まではテキスト中のコメントとして扱う。

4. 2 ファイルの記述例

グラフ記述言語は、データの並びである。開き括弧“(”と閉じ括弧“)”の間に、予約語（“領域”、“形式”など）、数値、文字列などの要素を空白文字（スペースや改行）で区切って並べることにより表記される。実際の使用方法として、軸の指定を例に挙げる。（図1参照）

<式>の記述例を図2に示す。例1は、内部の処理系に加算のための関数を用意しておき、関数へ渡す引き数として変数a, bを与える。例2も同様の処理が行われる。また、例からも分かるように、式の評価は内側の式から順に行われる。

```

( 軸
  ( 線
    ( 線種 実線 )
    ( 線幅 太 )
  )
  ( 軸名
    ( 文字列実体 "変化量" "結果" )
    ( 大きさ -2級 -2級 )
  )
)

```

図1 グラフ記述言語による軸指定

例1. a + b

(+ a b)

例2. e^{-x²}

(exp (- (* x x)))

図2 <式>の記述例

4. 3 凈書のソーステキストでの利用方法

凈書のソーステキスト中で、グラフ作成ツールを起動するコマンドに対し、位置、領域の大きさ、本文中の文字級数、ファイル名（グラフ記述言語によるテキストファイル）などを指定する。

5. おわりに

グラフ記述言語の定義とツールの実現により、グラフと文書の一体管理、データ入力の手間の問題は改善されるであろう。今後の課題として、別のグラフの機能、データへの操作（回帰、補間、統計量の表現）など、より科学技術系のグラフに必要な機能を拡張していくことが必要である。

参考文献

鈴木未来子、他：“システム記述言語の拡張言語「画匠」を用いた文書作成方式”，情報処理学会第40回全国大会講演論文集（II），6P-9，1990