



Gsharpによるデータの可視化

メタデータ	言語: jpn 出版者: 公開日: 2010-08-12 キーワード (Ja): キーワード (En): 作成者: 小島, 篤博 メールアドレス: 所属:
URL	http://hdl.handle.net/10466/10943

Gsharp によるデータの可視化

大阪府立大学 総合情報センター 小島 篤博

1 はじめに

論文や研究報告などにグラフはつきものですが、みなさんはどのようなソフトでグラフを作成されているでしょうか？フリーウェアでは GNU plot¹が定番ですが、機能面や表現力では力不足の感が否めません。また、Windows 上では Excel のグラフ表示機能を利用することもできますが、科学技術データを可視化する観点からいえば十分な種類の形式のサポートしているとはいえませんし、EPS (Encapsulated PostScript) 形式による出力ができませんので L^AT_EX に取り込むことが容易ではありません。

総合情報センターの画像処理サーバ (kongo.center.osakafu-u.ac.jp) には、Gsharp という可視化 (Visualization) ツールが用意されており、標準的な各種グラフをはじめ、グリッド図、等高線 (contour) 図、ベクトル図などを対話的に作成できます。また、作成した図を PS, EPS, JPEG などの形式で出力することができます。

本稿では、Gsharp を用いてデータを可視化する方法を紹介します。また作成した図を EPS 形式で出力し、L^AT_EX 文書に取り込む方法についても紹介します。

2 入力データ形式

まず、Gsharp に入力するデータの形式について説明します。作成するデータの形式にもよりますが、基本的にはスペース区切りのテキストファイルで、GNU plot や awk などに入力する形式と同じです。例えば、X-Y 平面上にグリッド図を描く場合の入力ファイルは次のようになります。

x 座標	y 座標	値
8	29	1
9	29	2
10	26	1
10	28	4
10	29	5
10	30	4
⋮	⋮	⋮

データを作成する際に、このような形式で出力するようにしておけば Gsharp で処理できるわけです。なお、ファイルに保存する際には、拡張子を '.dat' としておいてください。

3 Gsharp の起動

次に、Gsharp の起動方法について説明します。Gsharp は X11 アプリケーションですので、画像処理サーバ (以下 kongo) に login して起動し、画面だけをネットワーク経由で local 側 (手元) の端末に表示するようにします。このため、端末側では X11 サーバ機能をサポートする必要があります²。総合情報センターのオープンスペースにも、X11 が利用可能な SGI O² が用意されていますのでご利用下さい。

まず、local 側で、kongo からの X11 接続を許可します。UNIX WS の場合は、コマンドラインから以下のコマンドを実行します。

¹Linux などフリーな PC-UNIX ではパッケージとして用意されていますので、簡単にインストールできます。

²UNIX WS で X11 を利用している場合は問題ありません。それ以外の Windows や Mac などの OS では、X サーバソフトを別途インストールする必要があります。例えば Windows 系では Exceed など。

```
% xhost +kongo.center.osakafu-u.ac.jp
kongo.center.osakafu-u.ac.jp being added to access control list
```

Windows や Mac など X サーバソフトを利用している場合は、そちらのマニュアルを参照して設定してください。

次に kongo に telnet で login します。

```
% telnet kongo.center.osakafu-u.ac.jp
Trying 157.16.240.21...
Connected to kongo.center.osakafu-u.ac.jp.
Escape character is '^]'.

IRIX (kongo)

login:username ... ユーザ名
Password: ... パスワード
...
kongo 21%
```

次に、kongo 上で実行する X11 アプリケーションの画面を local 側に表示する設定をします。

```
kongo 22% setenv DISPLAY hostname:0.0
```

ここで *hostname* の部分に、local 側の FQDN を指定してください³。これで準備が整いましたので、Gsharp を起動します。kongo のコマンドラインから、次のコマンドを実行します。

```
kongo 23% Gsharp
```

これで Gsharp が起動します。“Error: Can't open display: *hostname:0.0*” などのエラーが出る場合には、もう一度ここまでの設定を見直してください。

4 データの可視化

Gsharp が起動すると、local 側に図 1 のような画面が表示されます。以下では、2 で示したような X 座標、Y 座標、値の 3 つの列からなるデータファイルを読み込み、X-Y 平面上に 3 次元のグリッド図を作成してみます。

4.1 データの読み込み

まず、File メニューから “Open...” を選択し、データファイルを開きます。データファイルの読み込みに成功すると、Gsharp のメッセージ・エリアに次のようなメッセージが表示されます。

```
Import Report: 1376 values read into T1, T2, T3
```

³ここでは必ず FQDN(Fully Qualified Domain Name) を指定してください。短いホスト名だけでは、サーバ側から端末を特定できないためです。

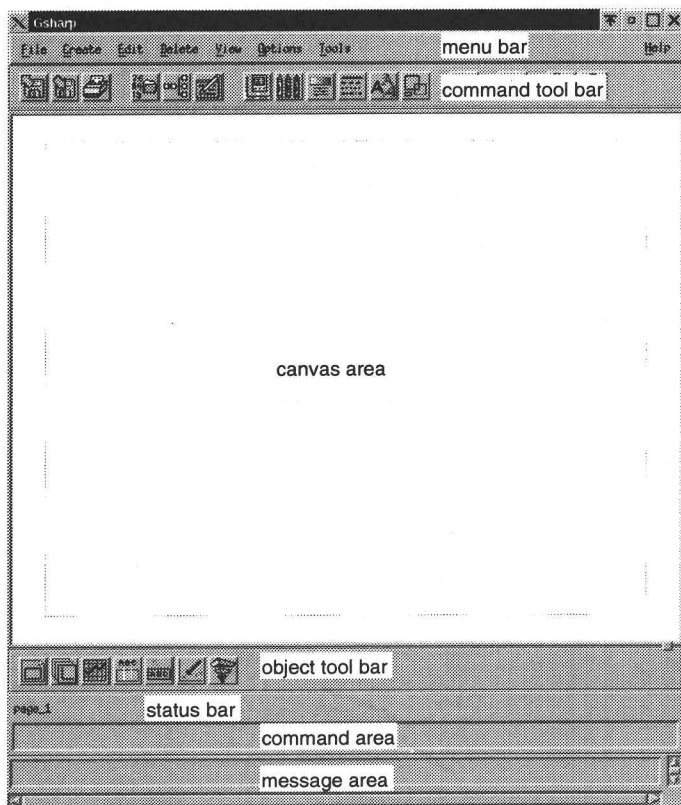


図 1: Gsharp の画面

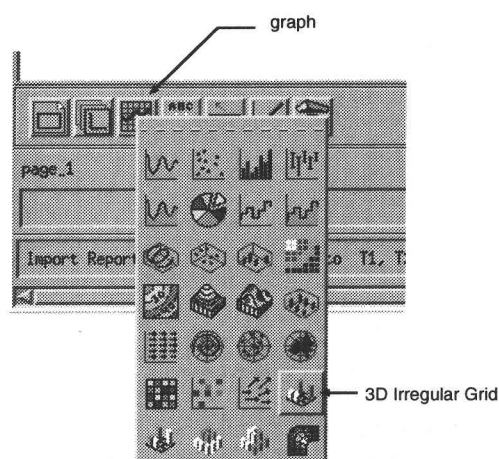


図 2: グラフの作成

これは、データファイルの各列 (X 座標, Y 座標, 値) が、それぞれ T1,T2,T3 という名前で、1376 行分読み込まれたことを示しています。つまり、T1,T2,T3 はそれぞれ 1376 個のデータからなっており、Gsharp ではこのようなデータの集合をデータセットと呼びます。以後、座標軸や値などでデータセットを指定する際には、T1,T2 などデータセット名を用いることになります。

4.2 グラフの作成

では、早速グラフを作成してみましょう。Gsharp のオブジェクト・ツールバー (画面の下側に並んだボタン) にある graph ボタン⁴の上にマウスカースルを移動し、右ボタンを押します。すると、いろいろな種類のグラフアイコンが表示されますので、図 2 のようなグラフ (3D Irregular Grid) を選択します。

作成するグラフのエディタ (図 3) が表示されるので、ここで人差し指のマークがついている必須項目を入力していきます。“Z Data”にはデータ値である T3 を、“X Position”, “Y Position”にはそれぞれ T1,T2 を指定します。データセットの設定は、入力欄の右にある“...”ボタンをクリックして、候補の中から選択します。設定が終わったら、ひとまず OK をクリックします。キャンバスエリアに図 4 のようなグラフが表示されるはずですが、

あとは、基本的にこのグラフの各パラメータを調整して、望ましい形にしていくことになります。このためには、Gsharp 特有の階層的なオブジェクト (構成要素) を理解しておく必要があります。

まず、コマンド・ツールバー (メニューバーの下に並んだボタン) の中にある Hierarchy Browser ボタンをクリックします。ここで表示されるのが、グラフを構成するオブジェクトの階層を表示する Hierarchy Browser (図 5) です。一番左側にある page オブジェクトを最上位として、viewport (視点) オブジェクト、domain (可視化範囲) オブジェクト、そしてその下には 3 つのサブツリーがあり、それぞれ graph オブジェクト、axis (座標軸) オブジェクト群、legend (凡例) オブジェクトが並びます。上位のオブジェクトのプロパティ (属性) を変更すると、下位のオブジェクトにも影響を与えます。例えば viewport オブジェクトを変更すると、下位のすべ

⁴それぞれのボタンの上にマウスカースルを移動し、少しの間動かさずにおくと、ボタンの機能が表示されます。

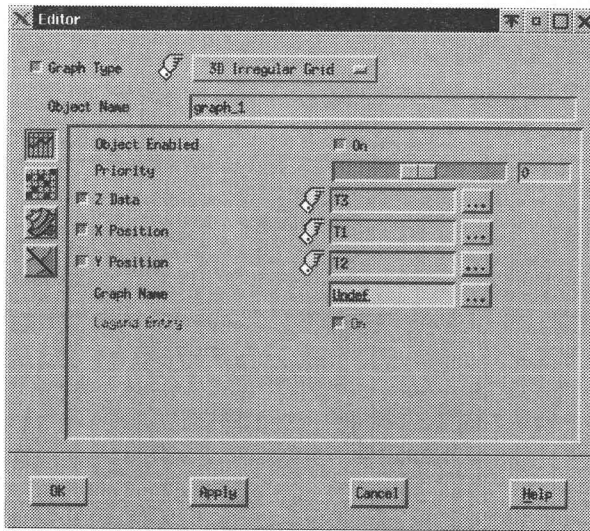


図 3: グラフエディタ

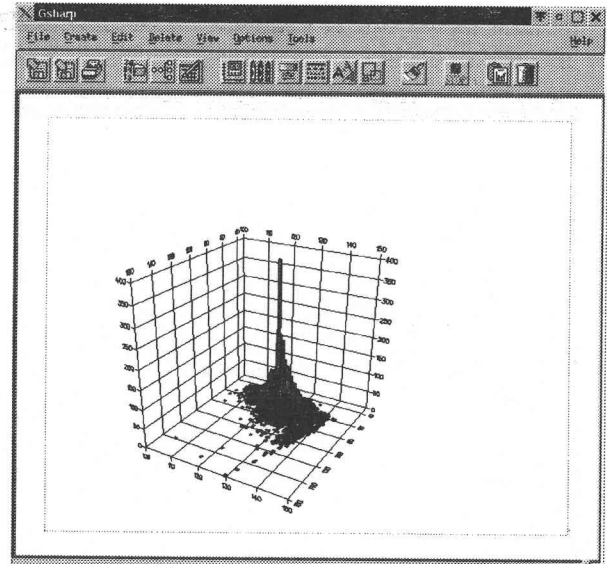


図 4: 3D グリッド図

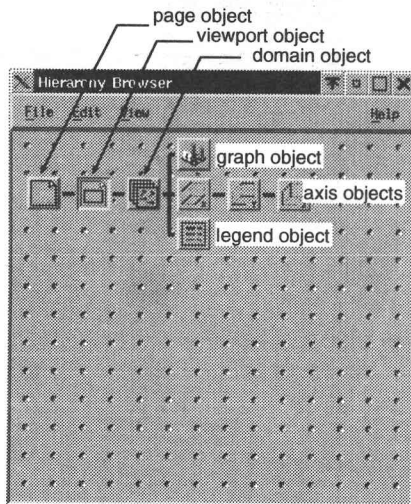


図 5: Hierarchy Browser

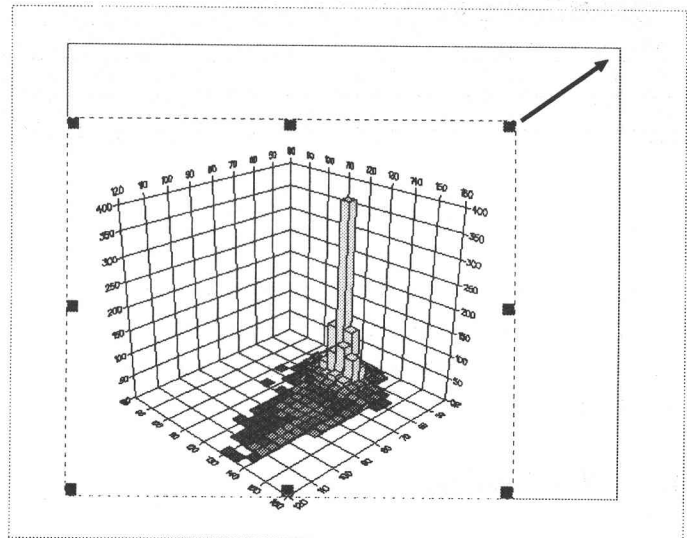


図 6: リサイズ

でのオブジェクトに対するの視点が変わります。

可視化範囲の設定

まず、グラフの可視化範囲 (domain) を調整してみます。Hierarchy Browser 上の domain オブジェクト (ここでは domain_1 という名前になっています) をダブルクリックすると、エディタが表示されますので、ここで各座標軸の範囲を設定します。Apply ボタンをクリックすると、現在の設定値をグラフに反映します。また、OK ボタンをクリックすると、設定値が反映されるとともにエディタも消えてしまいます。

グリッドの大きさ

次に、一見してグリッドの目が細かすぎるので、分割数をもう少し小さくしてやります。今度は graph オブジェクト (graph_1) をダブルクリックし、エディタを表示します。これは最初にグラフを作成したときのエディタ (図 3) ですが、左側に並んでいるボタンをクリックすることで、他のいくつかの項目についても設定することができます。このうち、Gridcell ボタンを押すと、X,Y それぞれのグリッドセルの分割数などを設定す

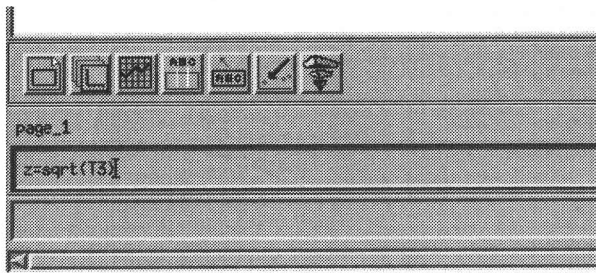


図 7: データセットの定義

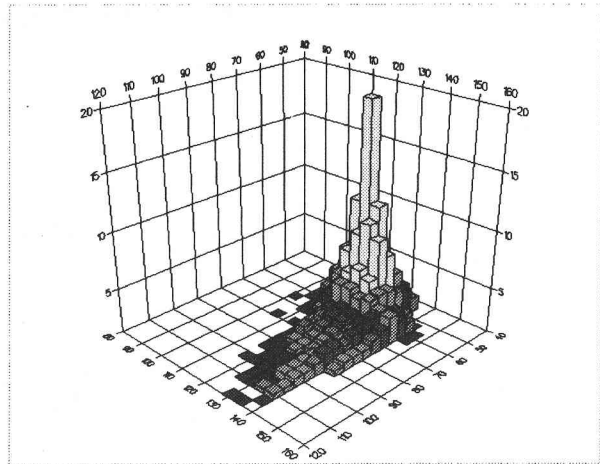


図 8: データセットを置き換えたグリッド図

る画面に切り替わります。これも Apply ボタンを随時クリックし、実際のグラフを見ながら最適な設定値を決めてやります。

視点の設定

グラフを見る視点は、viewport オブジェクト (viewport.1) で設定します。グラフの回転や俯角、各軸の伸縮比など細かく設定することができます。

移動・リサイズ

キャンパスエリア上のグラフをクリックすると、境界上の 8 点に黒い四角 (リサイズハンドル) が表示されます。この状態でグラフ領域内をドラッグすると、位置を移動することができます。またリサイズハンドルをドラッグすることで、グラフをリサイズすることができます (図 6)。

4.3 データセットの定義

Gsharp で利用できるデータセットは、データファイルから読み込んだものだけではありません。ここでは数式を使って新たなデータセットを定義してみます。Gsharp のキャンパスエリアの下側に、コマンドを入力できる部分 (コマンドエリア) があります。ここに “ $z=\text{sqrt}(T3)$ ” と入力します (図 7)。

これは、 $T3$ の平方根 (sqrt) を要素とする新たなデータセット z を定義することを意味しています。sqrt の他にも多くの関数が利用できます。図 8 は、新たに定義した z を使って、グリッド図のデータ値を置き換えたものです。

5 印刷とファイル出力

File メニューから “Print Setup...” を選択すると、Print Setup ダイアログが表示されます。ここでは、プリンタへの出力と、EPS などの形式によるファイルへの出力とを行うことができます。

5.1 プリンタ出力

プリンタで印刷する場合には、Output コマンドグループ内の “Command:” をチェックし、印刷コマンドを設定します。デフォルトでは “lpr -Psv1p92ps” となっていますので、総合情報センターのモノクロプリンタへ出力する場合はこのままで結構です。また、Device リストボックスにより出力形式を選択することができます。印刷の場合は、プリンタに合わせて “PostScript, A4 Paper”, “Color PostScript, A4 Paper” のいずれかを選んでください。設定が完了したら、Print ボタンをクリックすると印刷されます。

研究室などのプリンタへ出力したい場合は、次の項目で述べる方法で一旦 PS (PostScript) 形式でファイルへ出力したあと、そのファイルを研究室側のマシンに転送してからプリンタに出力してください。

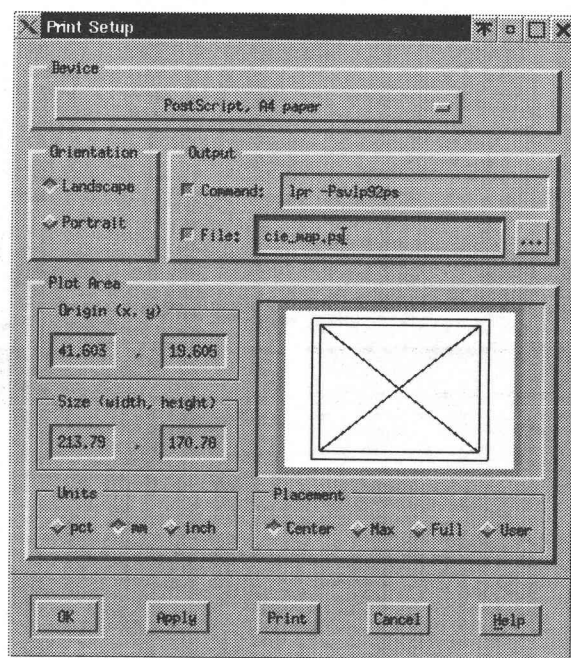


図 9: Print Setup ダイアログ

5.2 ファイル出力

ファイルに出力する場合には、まず Output コマンドグループ内の“File:”をチェックします。このとき、“Command:”のチェックは外しておいてください。次にファイル名を入力しますが、拡張子はファイル形式に応じて与えます。ここでは EPS 形式で出力すると仮定して、“cie_map.eps”というファイル名を使うことにします⁵。

ファイル形式は Device リストボックスから選択します。EPS 形式にはカラーとモノクロの2種類 (“Encapsulated Color PostScript, 100x100 mm”, “Encapsulated PostScript, 100x100 mm”)があり、それぞれ必要なものを選んでください。この他にも、TIFF や JPEG, Raster PostScript などがサポートされています。設定が完了したら、Print ボタンをクリックすればファイルへ出力されます。

5.3 Gsharp 形式での保存

Gsharp の各種オブジェクトを保存するためには、File メニューから“Save...”を選びます。ここでまず、File Type リストから“GSL Script”を選びます。これは読み込んだデータに対してどのような表示処理を行うかを記述したコマンドテキスト (Gsharp スクリプト) の形式です。次にファイル名を設定しますが、拡張子は“.gsl”としておいてください。OK ボタンをクリックすると、指定したファイルに保存されます。

ここで注意することは、GSL 形式で保存されるのはあくまで読み込んだデータの料理法であって、材料であるデータは一緒には保存されません。したがって Gsharp を再起動して作業を継続する場合には、まずデータを読み込んでから、この GSL ファイルを読み込む必要があります。

6 L^AT_EX での利用

L^AT_EX 文書の中に図を貼り付ける際には、EPS がよく利用されています。これまで述べてきたように、Gsharp では EPS 形式のファイルを出力できますので、これを実際に L^AT_EX 文書に取り込んでみます。

まず eclepsf.sty というスタイルファイルを入手し⁶、お使いの L^AT_EX システムのスタイルファイル検索パスの通ったディレクトリに置きます。L^AT_EX 文書に eclepsf.sty を読み込むための設定は、最初の行の documentstyle のオプションとして eclepsf を指定します。

⁵デフォルトでは“gsharp.ps”というファイル名になっており、ホームディレクトリ配下に作成されます。

⁶ほかのいくつかの ecl*.sty というファイルと一緒に、eclsty.tar.gz という形で配布されています。大阪府立大学の ftp サーバにもありますし、archie や検索エンジンでもすぐ見つかると思います。

```
\documentstyle[eclepsy]{jarticle}
```

図を挿入する場所では，次のような形式でファイル名やサイズを指定します．

```
\epsfile{file=cie_map.eps,width=10cm}
```

通常は figure 環境の中に取り込む場合が多いと思いますが，その場合には次のようになります．

```
\begin{figure}[tb]
  \centering
  \epsfile{file=cie_map.eps,width=150mm}
  \caption{グラフの例}
  \label{fig:example1}
\end{figure}
```

そして，実際に取り込んだ結果が図 10 です．

7 おわりに

Gsharp によるデータの可視化の方法と，その利用の一例として L^AT_EX への取り込みについて解説しました．Gsharp には，他にも今回紹介しきれなかった多くの機能がありますので，様々な研究分野のデータを可視化するお役に立てるのではないかと思います．ぜひ一度お試し下さい．なお，今回 Gsharp を利用する上で日本語簡易マニュアルが大変参考になりました．総合情報センターに置いてありますので，ご利用の際にはお問い合わせください．

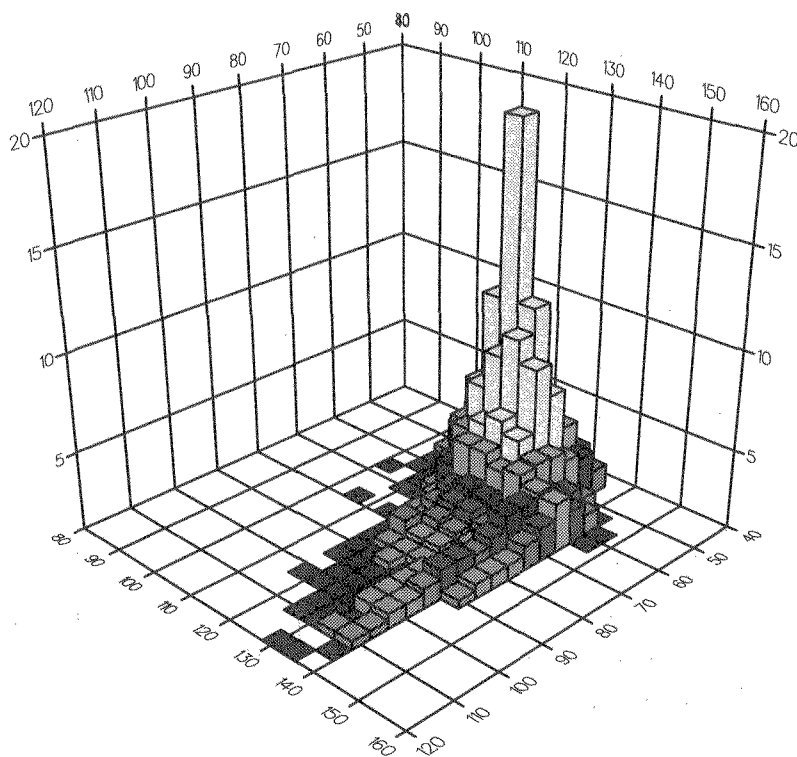


図 10: グラフの例