

効率的なRTシステム開発および運用のための汎用ビューワコンポーネント

芝浦工業大学
デザイン工学部デザイン工学科
佐々木毅

開発の背景

- RTコンポーネントによるシステム
 - 開発段階
 - それぞれのコンポーネントの動作確認
 - 運用段階
 - センサデータやその処理結果などの表示
 - 異常発生時の原因の早期発見
- 情報の**視覚化**が有効
- **コンポーネントの出力を視覚化するビューワコンポーネントを開発する**

ビューワコンポーネントの設計

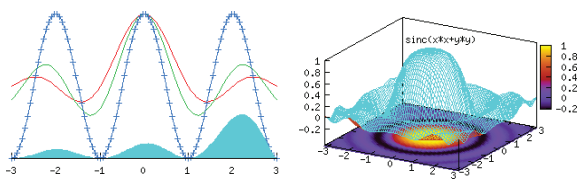
- ビューワコンポーネントに求められる要件
 - 基本的なデータ型のサポート
 - 様々な形式でのデータ表示への対応
 - 容易な使用法の提供

基本的なデータ型のサポート

- 可視化を行うデータは通常、単一の数値よりも数値列であると考えられる
 - 整数及び実数値の可変長1次元配列データ型
TimedShortSeq, TimedLongSeq,
TimedFloatSeq, TimedDoubleSeq
- 異なるコンポーネントの出力を比較するといった使用法を考慮すると、入力ポートは複数必要
 - 十分な数の入力ポートを予め用意→非効率的
 - 入出力ポートの動的な追加・削除機能

様々な形式でのデータ表示への対応

- グラフ描画ツールとして広く用いられているgnuplotを利用

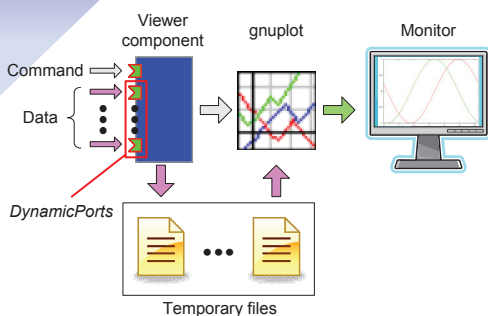


gnuplot homepage: <http://www.gnuplot.info/>より

容易な使用法の提供

- gnuplotのコマンドをそのまま利用可能な構成
 - ユーザは様々な機能を容易に活用することができる
- コンポーネントに入力されるデータは1次元配列データ
 - 入力データをgnuplotが扱える形式へと変換する機能が必要

ビューワコンポーネントの構成



開発したコンポーネント群

- ビューワコンポーネント (GnuplotViewer)
 - gnuplot を用いた汎用ビューワ
- コンソール文字列入力コンポーネント (ConsoleInString)
 - コンソールから入力した文字列をOutPort に出力
- 正弦・余弦関数出力コンポーネント (SinCosFunction)
 - 正弦波、余弦波を出力

GnuplotViewerの使用手順

1. Configurationの設定
 - a. 使用するポートの種類と数を設定する
 - b. 1次元ベクトルデータの区切り位置を設定する
 - c. 周期実行するプロットコマンドを設定する
2. コンポーネントをアクティブ化する
3. 出力を表示したいコンポーネントと接続する
4. 適宜Command ポートからプロット設定などのコマンドを入力する

- 手順1b及び1cは手順2の後に行うことも可能

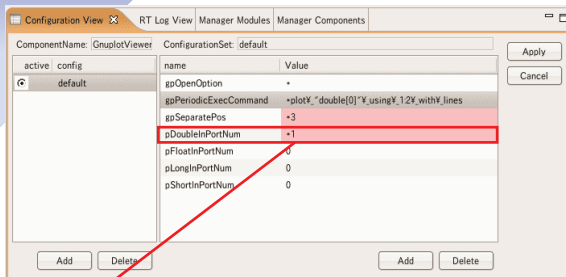
基本的な使い方

-1次元配列データのプロット機能の利用-

- 使用するコンポーネント
 - GnuplotViewer 1つ
 - ConsoleInString 1つ
 - SinCosFunction 1つ
- SinCosFunctionコンポーネントのsin(x)のデータをプロット

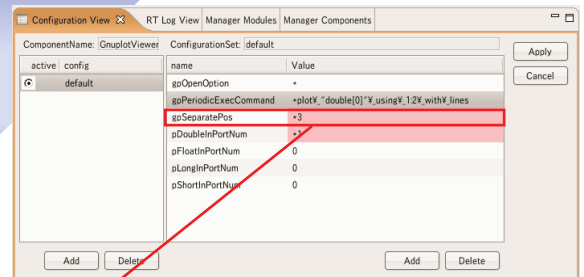
SinCosFunction → $x_0 \sin(x_0) \cos(x_0) x_1 \sin(x_1) \cos(x_1) \dots$
 (データはTimedDoubleSeq型)

Configurationの設定



- ポートの種類と数: TimedDoubleSeq 1つ

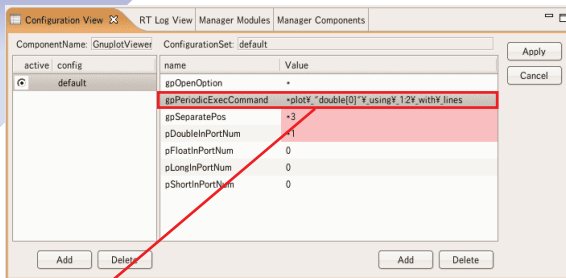
Configurationの設定



- データの区切り位置: 第3列で区切る

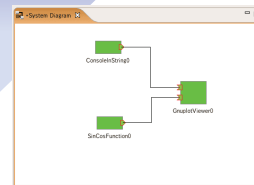
$x_0 \sin(x_0) \cos(x_0) x_1 \sin(x_1) \cos(x_1) \dots$
 $x_0 \sin(x_0) \cos(x_0)$
 $x_1 \sin(x_1) \cos(x_1)$
 \dots

Configurationの設定

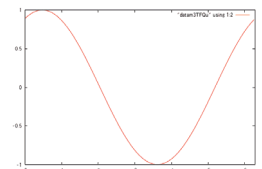


- 周期実行コマンド: x軸を第1列のデータ(x)、y軸を第2列のデータ(sin(x))として実線で表示
 - plot "double[0]" using 1:2 with lines

コンポーネントのアクティブ化と接続



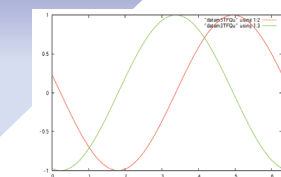
コンポーネントの接続図



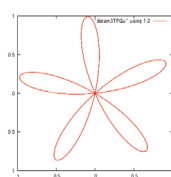
プロット結果

- ConsoleInStringコンポーネントを使って
GnuplotViewerコンポーネントに直接コマンドを送ることも可能
 - 上記のプロット結果も表示範囲を調整して見やすくしたもの

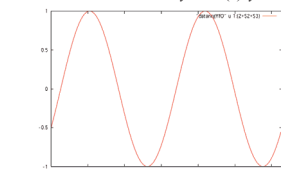
その他のプロット例



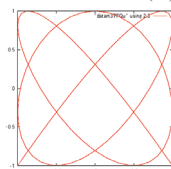
2つの系列を同時に表示 $y = \sin(x)$, $y = \cos(x)$



極座標プロット $r = \sin(5\theta)$

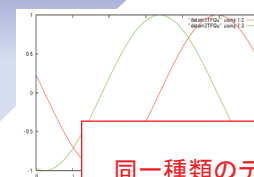


系列同士の演算 $y = 2\sin(x)\cos(x) = \sin(2x)$

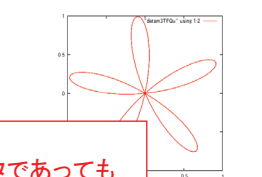


媒介変数表示 $x = \sin(5t)$, $y = \cos(6t)$

その他のプロット例

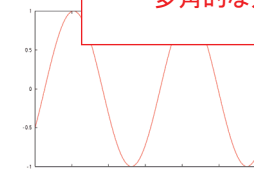


2つの系列を

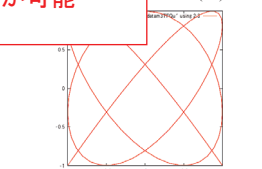


$r = \sin(5\theta)$

同一種類のデータであっても多角的な分析が可能



系列同士の演算 $y = 2\sin(x)\cos(x) = \sin(2x)$

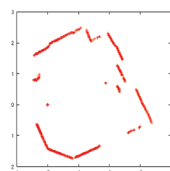
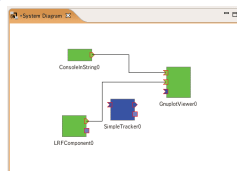


媒介変数表示 $x = \sin(5t)$, $y = \cos(6t)$

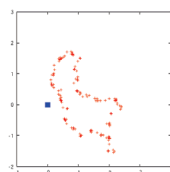
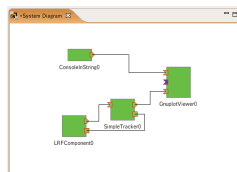
GnuplotViewerの実用例

-レーザレンジファインダのデータのプロット-

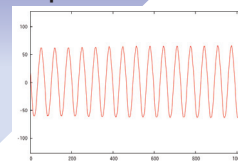
スキャンデータ
の表示



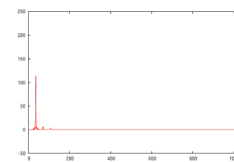
トラッキング
結果の表示



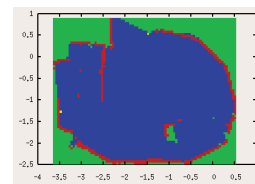
GnuplotViewerの実用例



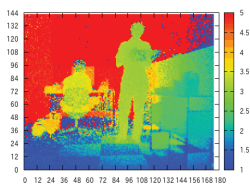
マイクから入力した音声データの表示



音声データのフーリエ解析結果の表示

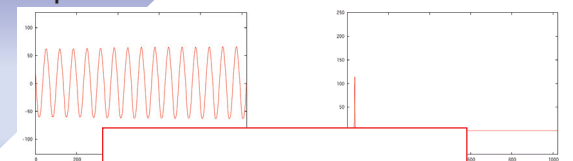


レーザレンジファインダによる
occupancy gridの表示

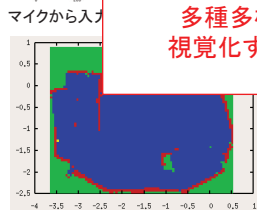


3次元距離測定カメラ(SwissRanger
SR-3000)の距離画像の表示

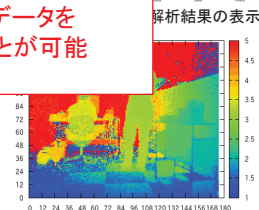
GnuplotViewerの実用例



多種多様なデータを
視覚化することが可能



レーザレンジファインダによる
occupancy gridの表示



3次元距離測定カメラ(SwissRanger
SR-3000)の距離画像の表示

まとめ

- RT コンポーネントの出力データを可視化するための汎用ビューワコンポーネントの開発
 - コンポーネントに対する要求事項と設計指針
 - 基本的なデータ型のサポート
 - 様々な形式でのデータ表示への対応
 - 容易な使用法の提供
 - コンポーネント群の実装
 - 動的入出力ポート
 - gnuplotを利用
 - ビューワコンポーネントの動作確認
 - 多角的な分析
 - 多種多様なデータの可視化