

CONTENTS

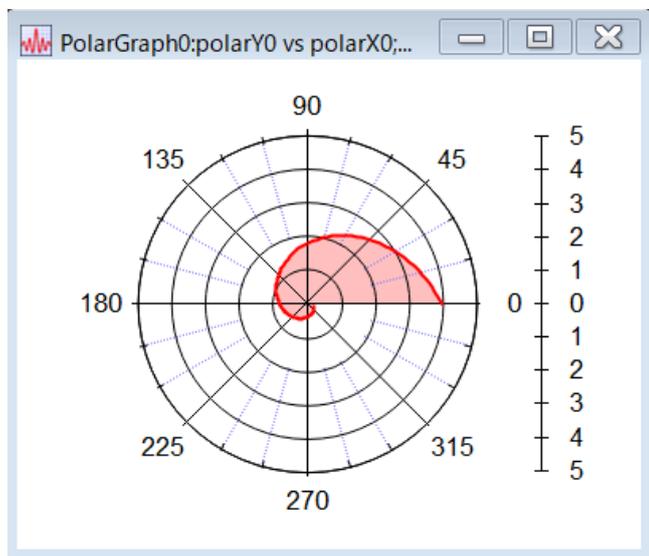
サンプルの Experiment – New Polar Graph Demo	2
Polar Graphs パネル.....	4
極座標グラフの作成.....	4
極座標グラフの更新.....	5
「Main」 タブ.....	5
「Range」 タブ	7
「Axes」 タブ.....	9
「Ticks」 タブ.....	11
「Labels」 タブ	12
Append Image ダイアログ.....	15
Polar Error Bars ダイアログ	16
「Radius Error Bars」 タブ	17
「Angle Error Bars」 タブ.....	19
Polar Layers ダイアログ.....	21
Axes, Grid, Labels, and Fills to Zero on One Drawing Layer の詳細.....	21
Multiple Drawing Layers (old method) の詳細.....	22
極座標のカーソル.....	22
極座標グラフの名前の変更.....	23
再作成マクロとして極座標グラフを保存	24
Polar Graphs パッケージから極座標グラフを切り離す	24

サンプルの Experiment – New Polar Graph Demo

メニュー **File** → **Example Experiments** → **Graphing Techniques** → **New Polar Graph Demo**

この Experiment は、極座標グラフ (Polar Graph) を作成するデモです。

極座標グラフは Igor Pro の標準機能ではありませんが、「New Polar Graph」プロシージャを使って作成することができます。

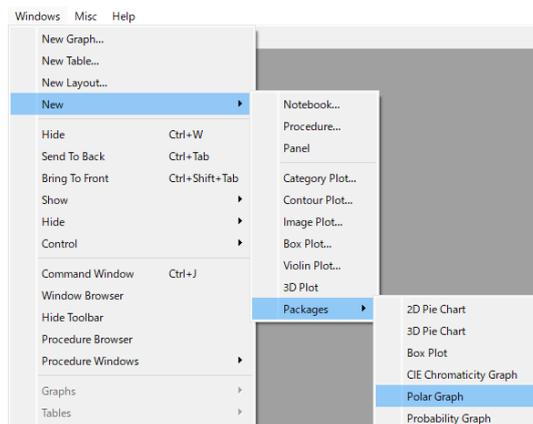


以下、新しい Experiment を作成したところから作業を始めます。

極座標グラフを作成する最初のステップは、メニュー **Windows** → **New** → **Packages** → **Polar Graph** を選択して、現在の Experiment に Polar Graph パッケージをロードすることです。

このメニューを選択すると、プロシージャウィンドウに次の行が追加されます：

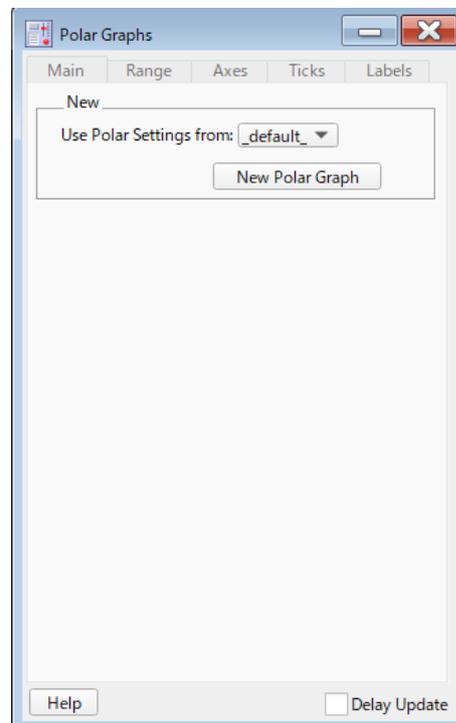
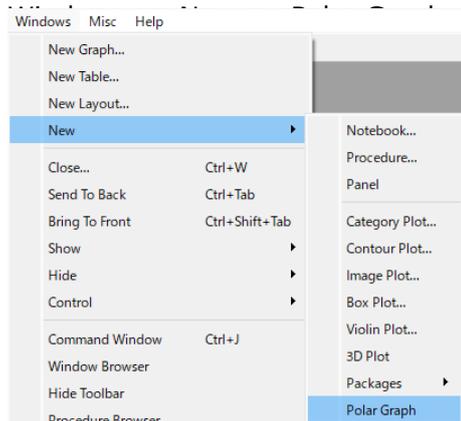
```
#include <New Polar Graphs>
```



次に Polar Graphs コントロールパネルが表示されます。

この include 行が、今回の Experiment で極座標グラフパッケージを利用可能にします。

これは、メニューに次の項目を追加します：



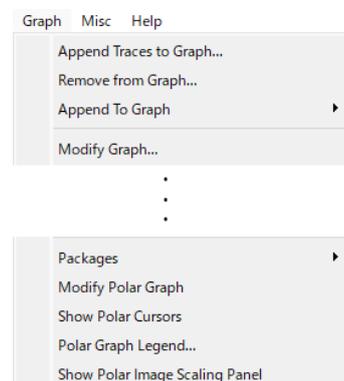
また、コントロールパネルで New Polar Graph ボタンをクリックし、新しいグラフウィンドウを表すると、Graph メニューに次の項目が追加されます：

Graph → Modify Polar Graph

Graph → Show Polar Cursors

Graph → Polar Graph Legend

Graph → Show Polar Image Scaling Panel



Graph → Modify Polar Graph と Windows → New → Polar Graph を選択することで、Polar Graph コントロールパネルをいつでも呼び出すことができます。

Polar Graphs パネル

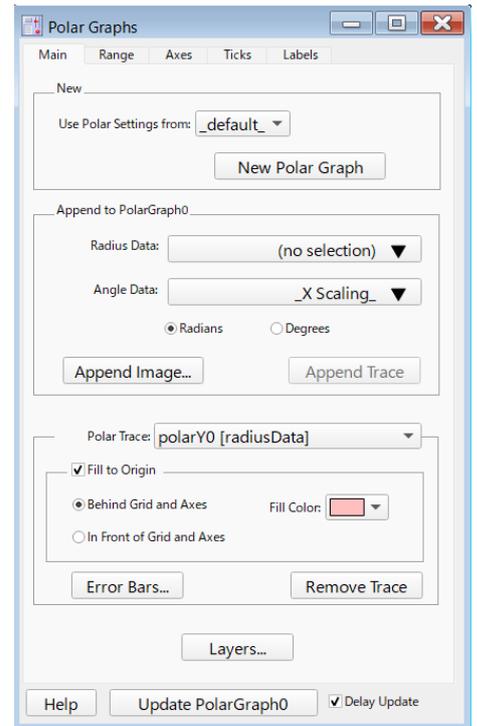
極座標グラフの作成と編集を行うためのメインインターフェースは、Polar Graphs パネルです。

右の図は、極座標グラフがすでに作成され、極座標トレースが追加されている場合の Main タブの内容を示しています。

パネルは、5つのタブで構成されています。

パネルは、左から右（タブ）そして上から下（各タブ内）の順序で使うように構成されています。

より一般的な設定や重要な設定は、Main タブにあり、最も詳細な設定は、右端のタブの最下部にあります。



極座標グラフの作成

極座標グラフを作成するには、半径と角度のデータが必要です。ウェブの内容が半径のデータとなります。

角度データはラジアン単位、または度単位で指定でき、別のウェブの内容から取得することも、半径ウェブの X スケーリングから取得することもできます。

簡単な例を示します。

コマンドラインで次を実行します。

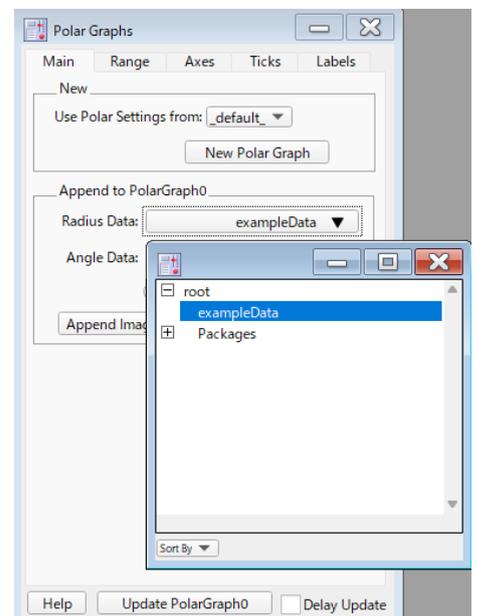
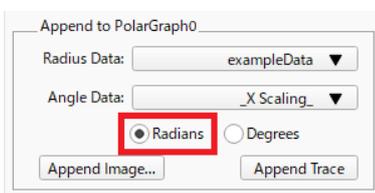
```
Make/N=100 exampleData = p/100  
SetScale x 0, 2*PI, "rad", exampleData
```

// 半径データを設定

// ラジアンで角度データを設定

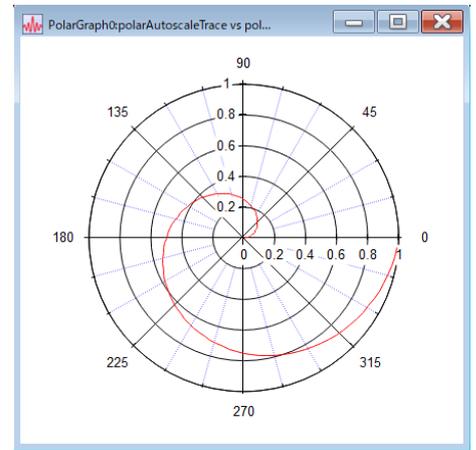
以下の手順に従って、極座標グラフを作成することができます。

1. Polar Graph パネルの Main タブにある New Polar Graph ボタンをクリックします。
2. exampleData を半径データ (Radius Data) として、X スケーリング (_X Scaling_) を角度データ (Angle data) として選択します。
3. ラジアン (Radians) ラジオボタンを選択します。



4. Append Trace ボタンをクリックします。

グラフウィンドウに、極座標グラフが表示されます。

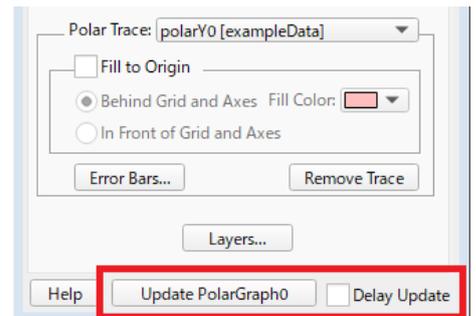


極座標グラフの更新

更新が行われるたびに、極座標グラフのコードでグラフを更新するか、または Delay Update チェックボックス（パネルの Main タブの下部/デフォルトではオフ）で更新を制限するかを選択できます。

一部の設定変更（例えば、ズームインなど）では、意図的に Delay Update が無視されます。

処理速度の遅いコンピューターの場合や、極座標グラフの再描画を避けたい場合は Delay Update チェックボックスを使うことを推奨します。



「Update <グラフの名前>」ボタン（例：“Update PolarGraph0”）は、Delay Update チェックボックスの状態に関わらず、最前面の極座標グラフを完全に再描画するために使われます。

Delay Update をオフにしても、極座標グラフのコードが認識できないイベントがいくつかあり、グラフの更新が必要になる場合があります。

その場合は、Update ボタンをクリックすると更新できます。

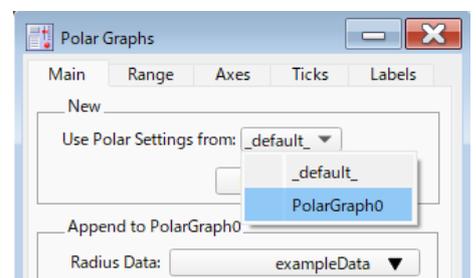
「Main」タブ

Main タブは次の機能を持ちます：

- 新しい極座標グラフを作成
- 極座標グラフに半径/角度データを追加
- 極座標グラフに画像データを追加
- 極座標グラフのトレースを修正（塗りつぶし、エラーバー）
- 極座標グラフからトレースや画像を削除
- 極座標グラフの要素を描画するために使われる描画レイヤーを変更

新しい極座標グラフをデフォルト設定で作成するか、「Use Polar Settings from:」ポップアップで選択した別の既存の極座標グラフの設定を使って作成することができます。

プロットされているトレースは設定の一部とはみなされません。



新しく作成した極座標グラフには、トレースや画像が含まれていません。

半径と角度のデータを指定し、Append Trace ボタンをクリックすることで、極座標のトレースを追加できます。

極座標グラフに複数のトレースを1つずつ追加することができます。

角度データは、個別のウェーブまたは半径ウェーブの X スケーリングから取得できます。

Append Image ボタンをクリックすることで、極座標グラフに画像を追加することもできます。

New Polar Graph パッケージは、半径と角度のデータに極座標-直交座標変換を適用した「シャドウウェーブ」を作成し、その XY シャドウウェーブを通常のグラフトレースとしてグラフに追加します。

極座標-直交座標変換は、ラジアンと度数のラジオボタンの状態を記憶し、適用します。

追加されたトレースは、「Polar Trace:」ポップアップメニューにリスト表示されます。

トレース名は Y シャドウウェーブの名前です。

便利のように、Radius Data ウェーブの名前が [] 内に表示されています。

この例では、半径ウェーブの名前は exampleData、極座標トレースの名前 (Modify Trace Appearance ダイアログに表示される名前) は polarY0 です。

極座標グラフのトレースのリストに別のトレースがあることに気づくかもしれません。

polarAutoScaleTrace は、AutoScale コマンドがグラフのサイズとスケールを調整することを確実にするために使われる非表示の極座標トレースです。

これにより、極座標軸、グリッド、ラベルがグラフの範囲内に収まるようになります。

Modify Trace Appearance ダイアログを使って、グラフトレースに対して行える変更をすべて行うことができます。

それが、Polar Trace セクションの設定項目が少ない理由です。

Polar Trace セクションの設定項目は、オプションの「Fill to Origin (原点まで塗りつぶす)」とエラーバーに関連するものです。

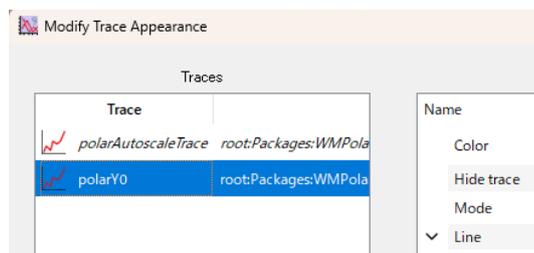
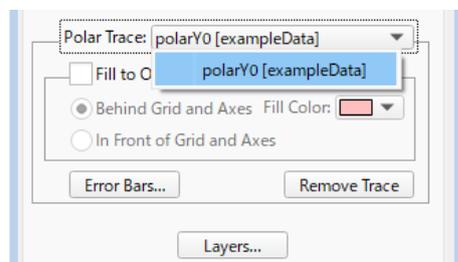
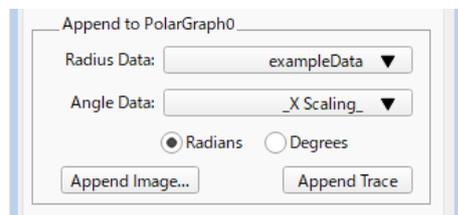
Fill to Origin とエラーバーは、半径と角度のデータから自動的に更新されます。

Remove Trace をクリックすると、選択した極座標トレースがグラフから削除されます。

これは、Remove from Graph ダイアログを使うよりも望ましい方法です。

polarAutoscaleTrace は削除しないでください。

Layers ボタンは、極座標グリッド、軸、ラベルなどが、どのレイヤー上に描画されるかに関する詳細をコントロールします。



「Range」タブ

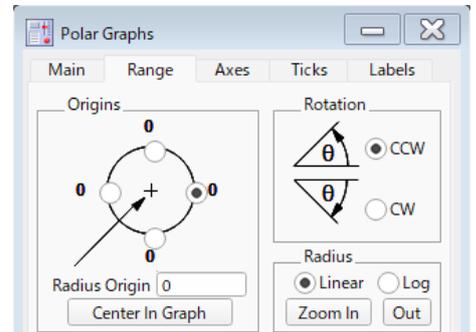
Range タブでは、極座標グラフの座標系を設定します。

- 角度と半径の原点の物理的な位置
- 半径の原点の値
- 時計回りまたは反時計回りの角度の方向
- 線形と対数の半径の軸
- 軸とグリッドの範囲

極座標の基本的な方向は、Range タブの Origins、Rotation、Radius セクションで設定されます。

Center In Graph ボタンは、半径の原点をグラフの中心に配置します (Autoscale もオフにします)。

Rotation ラジオボタンは、角度の増加の方向をコントロールします。



Radius ラジオボタンは、極座標-直角座標変換に \log_{10} 演算が含まれるかどうかを決定します。変換の方程式は：

(線形半径変換)

$$\text{PolarY0} = (\text{radius} - \text{radiusOrigin}) * \sin(\text{angle})$$

$$\text{PolarX0} = (\text{radius} - \text{radiusOrigin}) * \cos(\text{angle})$$

(対数半径変換)

$$\text{PolarY0} = \max(0, \log(\text{radius}/\text{radiusOrigin})) * \sin(\text{angle})$$

$$\text{PolarX0} = \max(0, \log(\text{radius}/\text{radiusOrigin})) * \cos(\text{angle})$$

半径の原点が 0 というのは、対数軸にとっては本当に悪い考えであることには注意してください。

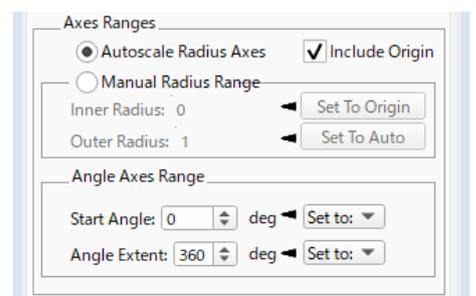
Zoom In/Out ボタンは、グラフに表示される半径の範囲を拡大または縮小します。

また、グラフの Autoscale モードをオフにします。

Autoscale モードに戻すには Ctrl+A を使います。

Axis Ranges は、描画された極座標軸がグラフ内に収まるかどうかをコントロールします。

グラフが Autoscale モードの場合、polarAutoscaleTrace はグラフ領域内に軸を維持しようとしますが、Center In Graph、Zoom In/Out ボタンはこの機能を無効にします。



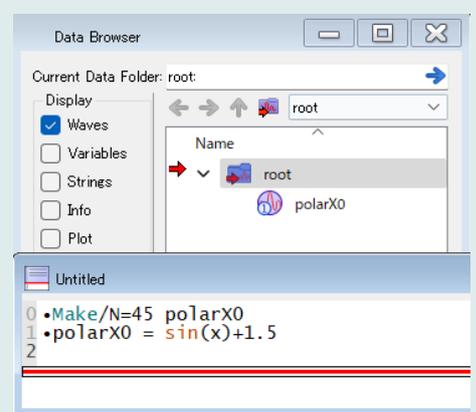
Autoscale Radius Axes (これは、グラフを Autoscale モードにはしません) は、半径軸の範囲を設定し、最小値と最大値の半径データ値を自動的にカバーします。

軸の範囲には、「適切な」値が選択されていますが、アルゴリズムは Igor のネイティブのグラフ軸で使われているアルゴリズムよりかなり原始的なものです。

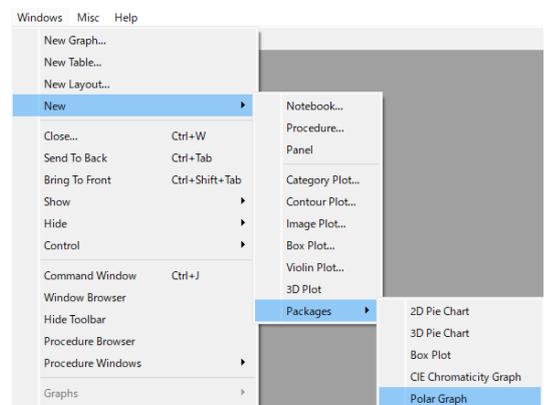
新しい Experiment を作成したところからの手順で確認します。
 (サンプルの Experiment に含まれるデータとは異なります)

1. コマンドラインで次を実行します。

```
Make/N=45 polarX0
polarX0 = sin(x)+1.5
```

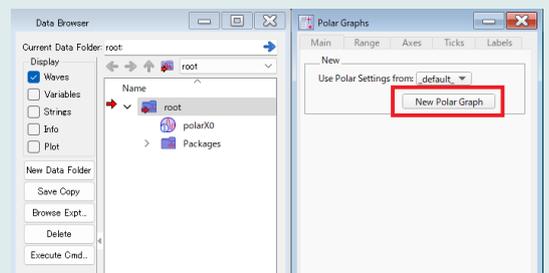


2. メニュー Windows → New → Packages → Polar Graph を選択します。

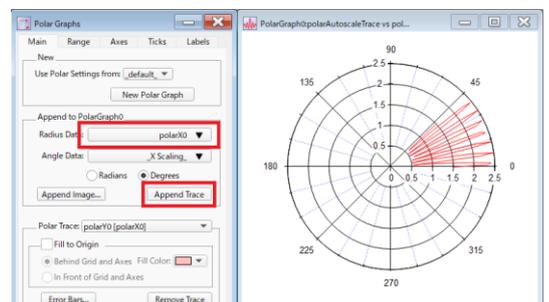


3. Polar Graphs パネルが表示されます。

New Polar Graph ボタンをクリックします。

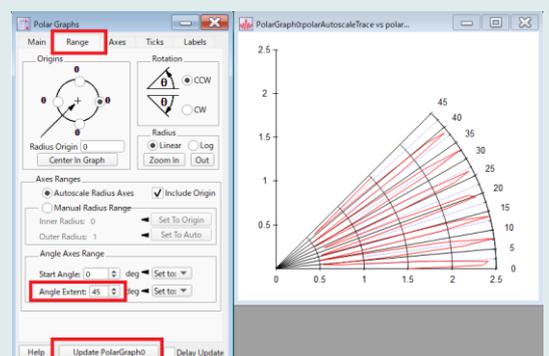


4. Radius Data ポップアップメニューで polarX0 を選択し、Append Trace ボタンをクリックします。



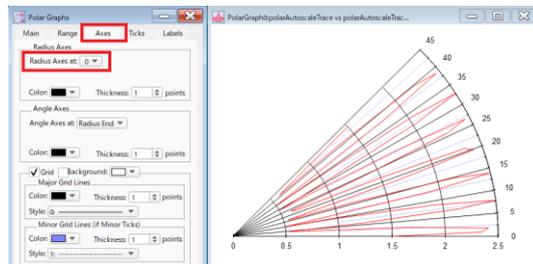
5. Range タブをクリックし、Angle Axes Range セクションの Angle Extent を 45 に設定します。

パネル下部の Update PolarGraph0 ボタンをクリックします。



6. 縦の軸が余分なので、消しておきます。

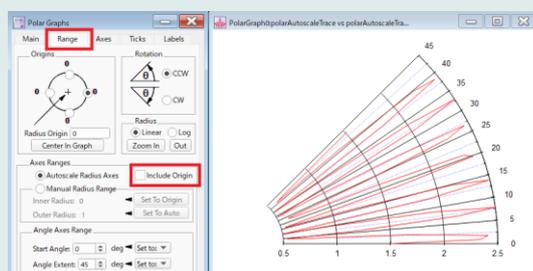
Axes タブをクリックし、Radius Axes セクションの Radius Axes at ポップアップメニューで 0 を選択します。



7. **Range タブをクリックして、Include Origin チェックボックスをオフにします。**

軸の範囲も手動で設定できます。

角度軸の自動スケール設定はありません。



注記：上の例で示されているように、軸の範囲は極座標グリッドの範囲も設定します（グリッドはデフォルトで Axes タブで有効になっています）。

「Axes」タブ

Axes タブでは、半径軸、角度軸、グリッドの位置と外観をコントロールします。

- 半径軸が描画される場所、その色と線の太さ
- 角度軸が描画される場所、その色と線の太さ
- グリッドの有効化
- 主グリッド線、副グリッド線の表示
- 隣接グリッド線の最小間隔
- 円弧や円をどれだけ正確に描画するか

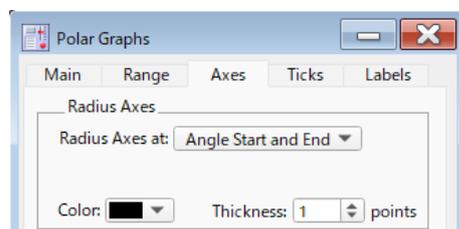
これらの半径軸コントロールを使って、半径軸の位置と外観をコントロールします。

半径軸は、グラフの原点から、またはグラフの一方から描画することができます。

その場所は、Radius Axes at ポップアップメニューで指定します。

メニューの多くは角度または角度の一覧を「度数」で表したものです。

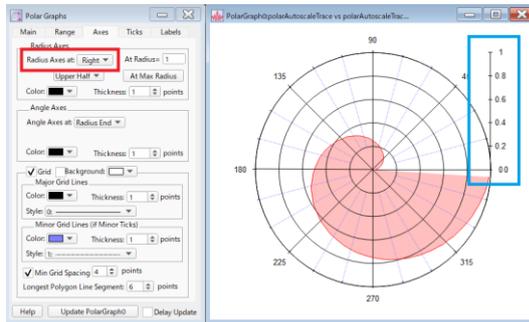
以下、設定できる項目について説明します。



Radius Axes at: の項目	意味
Angle Start	Range タブの Angle Axes Range の Start Angle の位置に表示
Angle End	Range タブの Start Angle + Angle Extend の位置に表示
Left	グラフの左側、指定の半径で垂直に表示
Right	グラフの右側、指定の半径で垂直に表示

Top	グラフの上側、指定の半径で水平に表示
Bottom	グラフの下側、指定の半径で水平に表示
All Major Angles	すべての主な（ラベルが付けられた）角度に表示
All Listed Angles	ユーザーが指定するカンマで区切られた角度（度数）に表示

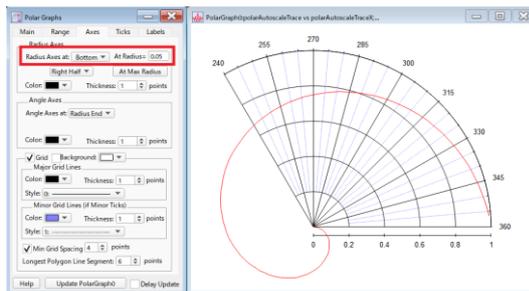
Left、Right、Top、Bottom の選択により、これらの軸の詳細を指定するその他のコントロールが表示されます。



Both Halves のオプションは、Top Half（上半分）と Bottom Half（下半分）です。

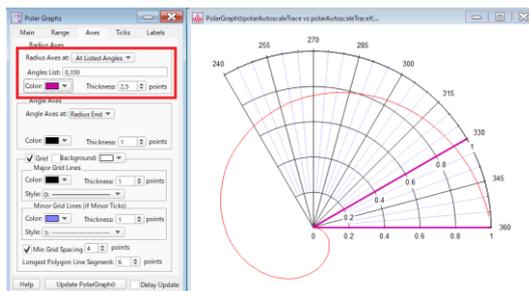
Bottom または Top 半径軸に対しては、この選択肢は Left Half（左半分）と Right Half（右半分）になります。

半径軸は元の円の半径の位置にあるため、At Radius を 0.05 に設定すると、右図のようになります。



At Listed Angles を選択すると、半径軸または軸が描画される角度（度数）を入力するコントロールが表示されます。

これらの角度軸コントロールで、角度軸の位置と外観をコントロールします。



角度軸は常に原点を中心とする円弧です。

位置は、ポップアップの Angle Axes によって決まります。

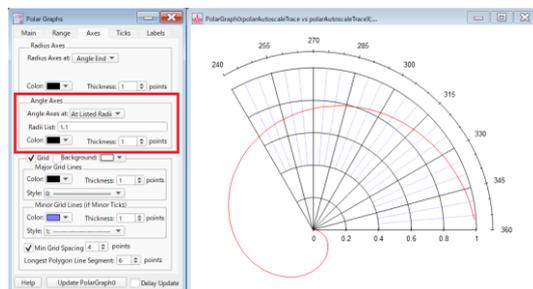
Angle Axes at: の項目 意味

Radius Start	Range タブの Inner Radius、または Autoscale の最小 *
Radius End	Range タブの Outer Radius、または Autoscale の最大
Radius Start and End	2つの角度軸が描画 *
All Major Radii	すべての主な（ラベルが付けられた）半径に表示
At Listed Radii	カンマで分けられた、指定された半径 *

(* RadiusStart が原点の場合、その角度軸は描画されない)

At Listed Angles を選択すると、半径軸または軸が描画される角度、At Listed Radii を選択すると角度軸または角度軸が描画される半径を入力するコントロールが表示されます。

半径より少し大きな値を指定すると、右のようになります。

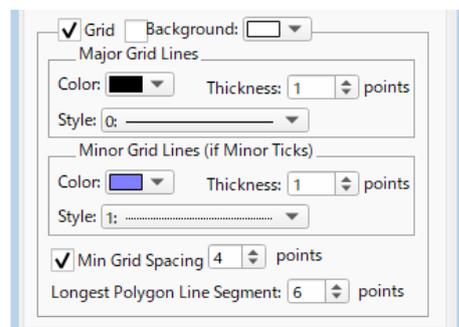


極座標グリッド

極座標グリッド（範囲は Range タブで、主目盛と補助目盛は Ticks タブで設定）の外観は、右図のコントロールによって設定されます。

オプションのグリッド背景色 (Background:) は、指定した半径と角度軸の範囲をカバーします。

Min Grid Spacing（最小グリッド間隔：ポイント単位）は、グリッド線が非常に接近している場合（この例では 4 ポイントより接近している場合）、グリッド線が省略されます。

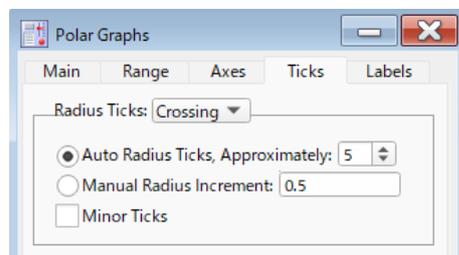
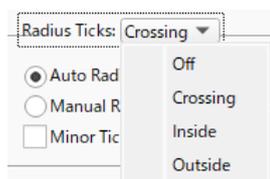


「Ticks」タブ

Ticks タブでは、軸目盛の位置と外観をコントロールし、その結果として軸ラベルの値をコントロールします。

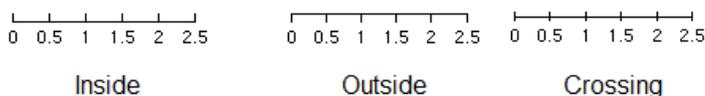
- 目盛のオン/オフ
- 描画された目盛がどのように軸と交差するか
- メモリの太さと長さ
- ラベル付きの主目盛と補助目盛の間の増分

目盛のオン/オフや位置は、Radius Ticks ポップアップメニューで設定します。



Inside と Outside の意味は、通常のラベルの位置との相対的な関係です。

Inside はラベルの反対側の軸側に目盛を表示し、Outside はラベルと同じ側に目盛を描画します。



半径目盛の範囲は、Range タブの Axes Range でコントロールされます。

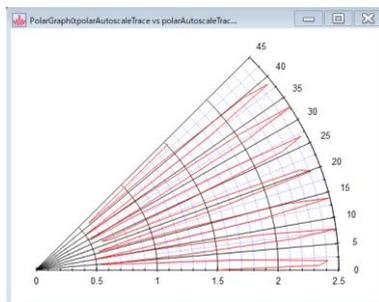
主目盛（特にどの値にラベルを付けるか）の配置は、Auto Radius Ticks または Manual Radius Increment によってコントロールします。

Auto Radius Ticks は、軸の全範囲にわたって、指定した数の主半径目盛をほぼ等間隔に配置します。

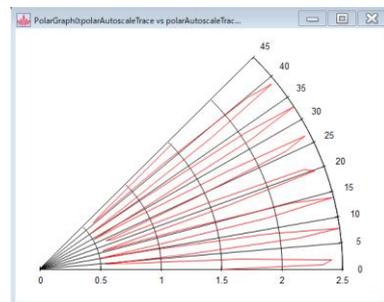
Manual Radius Increment は、主半径目盛を $\text{Start Radius} + n * \text{Manual Radius Increment}$ に配置します。

（Start Radius は、Range タブで選択した手動の Inner Radius または Autoscale Start Radius です）

Minor Ticks（補助目盛）チェックボックスは、半径軸の補助目盛と補助グリッド線の両方をコントロールします。



Minor Ticks オン



Minor Ticks オフ

Minor Ticks がチェックされている場合、グリッドと補助目盛の正確な数を設定します。

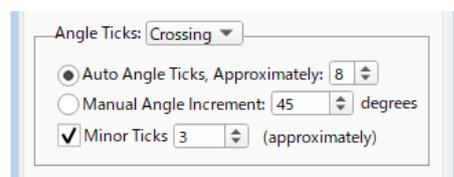
4 ticks を選択すると、主目盛が5つの刻みに分割されます。

角度目盛は半径目盛とほぼ同じように機能しますが、Auto Angle Ticks の補助目盛の数をある程度コントロールできるという例外があります。

手動の Minor Ticks は、半径目盛と同じように機能します。

主目盛と補助目盛のサイズと長さは、Tick Sizes コントロールで設定します。

これらは、半径と角度の目盛の両方に適用されます。



「Labels」タブ

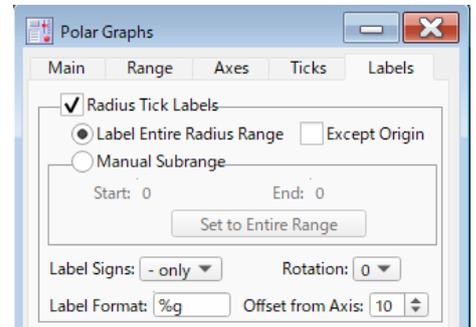
Labels タブでは、軸目盛ラベルの表示をコントロールします。

- どの主目盛をラベル付けするか（すべてまたはサブレンジ）
- ラベルを軸からどれだけオフセットするか
- 単位と符号を含むラベル形式の追加または非表示
- ラベルの回転
- ラベルのフォント、サイズ、色、透明度

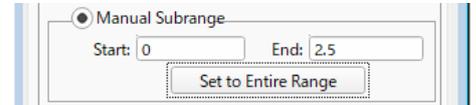
Radius Tick Labels (半径目盛ラベル) チェックボックスで半径目盛ラベルのオン/オフを切り替えます。

チェックを入れると、主目盛にラベルが付けられますが、ラベル付けをサブレンジに制限することもできます。

Label Entire Radius Range を選択した場合に表示される範囲は、Range タブの Axis Range で設定します。



これは、Manual Subrange の Set to Entire Range ボタンをクリックしたときの範囲でもあります。



半径のラベルは通常、すべて正の値ですが、「Label Signs」ポップアップメニューでマイナス記号を非表示にしたり、プラス記号を強制的に追加したりできます。



デフォルトは「- only」で、マイナス記号は表示されますが、プラス記号は表示されません。

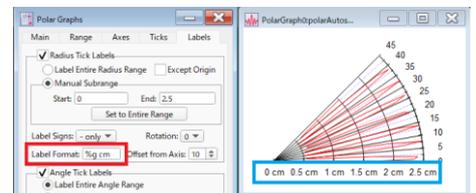
半径目盛ラベルは、数値と Label Format を使った sprintf コマンドで作成されます。

```
String labelString
sprintf labelString labelFormat, labelValue // デフォルトの labelFormat は "%g"
```

Label Format 文字列には、任意の半径単位を含めることができます。

これらは自動的に挿入されることはありません。

手動で cm を入力して追加すると右図のようになります。



Igor Pro 6.23 以降では、出力された文字列の結果が有効な文字列式である場合、その式を評価して実際の目盛ラベルを計算します。

つまり、RadiusLabel(value) という文字列関数を定義し、Label Format を RadiusLabel(%g) に設定すると、この関数の結果が半径ラベルとして使われることとなります。

これは角度目盛ラベルにも適用されます。

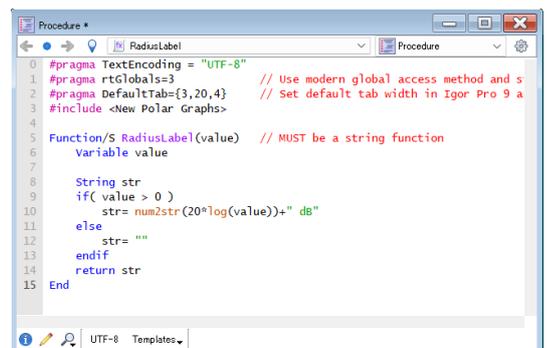
以下は、半径の $20 * \log_{10}$ を表示し、dB を付加する RadiusLabel 文字列関数の例です。

メニュー Windows → Procedure Windows → Procedure Window を選択します。

次のコードを貼り付けます。

```
// 文字列関数でなければならない
Function/S RadiusLabel(value)
    Variable value

    String str
    if( value > 0 )
        str= num2str(20*log(value))+ " dB"
    else
        str= ""
    endif
endif
```



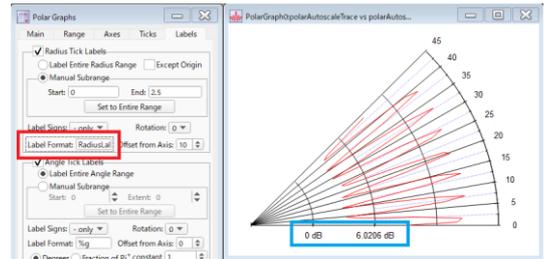
```
return str
```

End

貼り付けたら、下の Compile ボタンをクリックします。

Label Format のボックスを RadiusLabel(%g) に変更し、パネル下部の Update PolarGraph0 をクリックします。

軸ラベルが計算され、dB が付加されます。



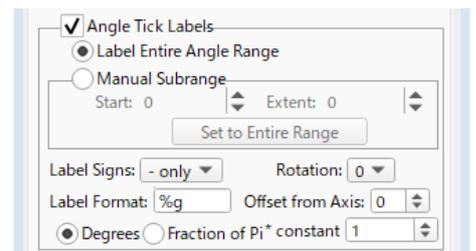
目盛ラベルは、中央アンカー (SetDrawEnv textxjust=0, textyjust=0) を使って DrawText コマンドで描画されます。

ラベルのアンカーは、軸から Offset from Axis で指定したポイント数だけオフセットされます。

Angle Tick Labels (角度目盛ラベル) は、半径目盛ラベルと同じように機能します。

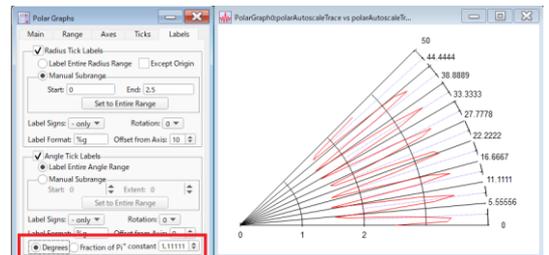
ただし、角度固有の追加設定がいくつかあります。

Degrees / Fraction of Pi ラジオボタンと *constant (*定数) は、角度目盛ラベルに表示される数値を決定します。

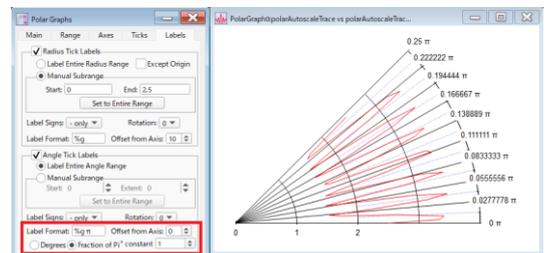


*constant パラメーターが 1 に設定されている場合、角度は度数法または円周率の分数でそのまま表示されます。

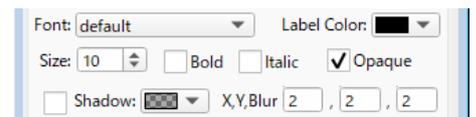
ラジオボタンで Degrees を選択し、*constant を $100/90=1.111111$ に設定すると、度ではなくグラード(メートル法の度数)で表示されます。



Fraction of Pi 設定は、Label Format の %g を、ギリシア文字のパイ (π) が続くように計算された数字に置き換えます。



半径目盛と角度目盛のフォント、フォントサイズ、スタイル、色、透明度の設定は共通です。

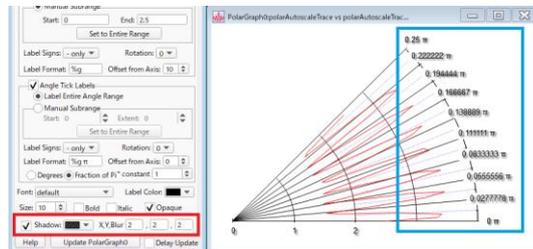


Opaque をチェックすると、ラベルの背後に四角形が描画されます。

四角形はグラフの背景色と同じ色で塗られています。

グラフの背景色を変更した場合は、Update <graphName> ボタンをクリックする必要がある場合があります。

Shadow のチェックをオンにすると、ラベルのオフセットぼかしコピーが適用され、背景から目立つようになります。
値はポイント単位です。明るい影よりも暗い影のほうが効果的です。

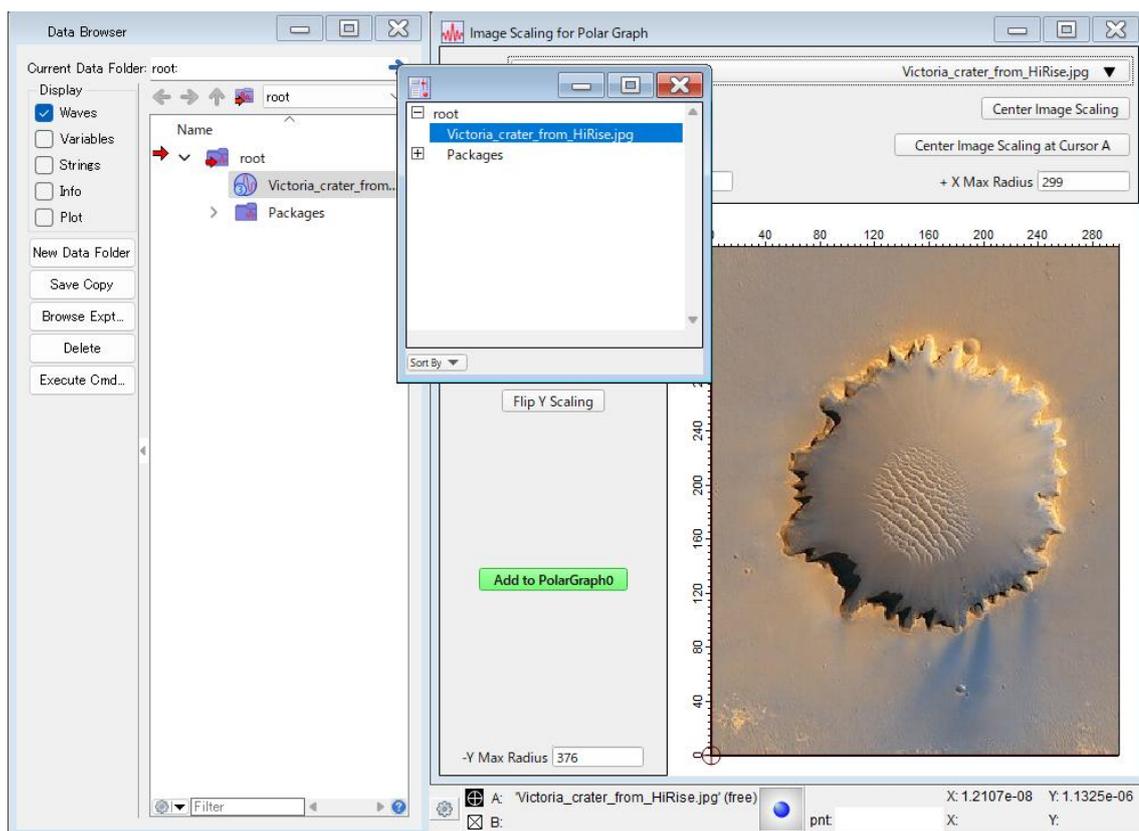


Append Image ダイアログ

極座標グラフ（トレースの有無にかかわらず）に画像を追加するには、Main タブの Append Image ボタンをクリックします。

これにより、Image Scaling for Polar Graph ダイアログが表示されます。

画像は、メニュー Data → Load Waves → Load Image で Experiment にロードしておきます。



使用画像 : [Mars Reconnaissance Orbiter](https://en.wikipedia.org/wiki/Mars_Reconnaissance_Orbiter) (MRO) [HiRISE](https://en.wikipedia.org/wiki/HiRISE) image of Victoria on October 3, 2006.

Wikipedia: https://en.wikipedia.org/wiki/Victoria_%28crater%29

Center Image Scaling をクリックすると、中央を示す位置が表示されます。

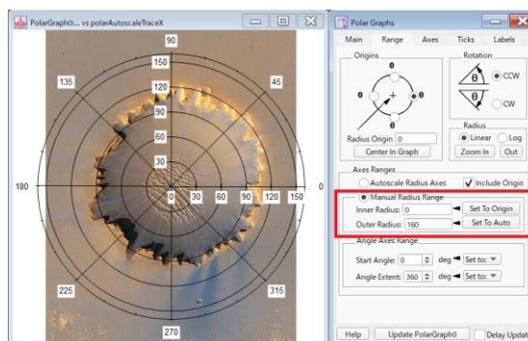
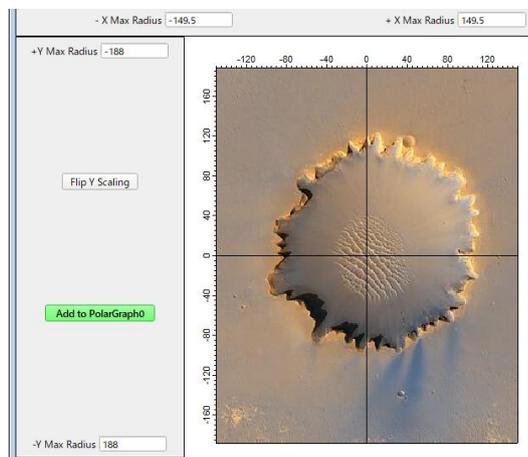
X/Y Max/Min Radius のボックスでスケールを調整します。

このダイアログの目的は、追加する画像を選択し、原点が便利な位置になるように、その X と Y のスケールを設定することです。

極座標グラフでは、グラフの下から上に向かって値が増加する Y 軸を使っているため、Flip Y Scaling (Y 軸反転) ボタンが役に立つ場合があります。

スケールを満足のいくように調整したら、Add to PolarGraph<n> ボタンをクリックします。

自動スケール半径と角度を計算するためのトレースがない場合、それらを表示するには、極座標グリッドを手動で定義する必要があります。



色とラベルを少し調整し、クレーターの輪郭を示す極座標のトレース (Radius_Crater) を追加し、透明の赤で原点まで塗りつぶすと次のようになります。

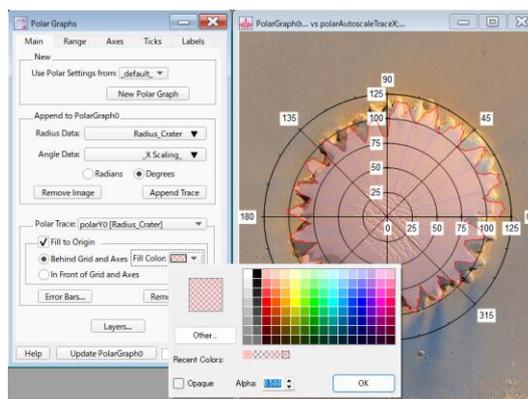
ここでは、輪郭のトレース用のデータがないため、仮に次のコマンドで生成したデータを輪郭データとして、画面をキャプチャしました。

Make/N=360 Radius_Crater

Radius_Crater = (x<90) ?

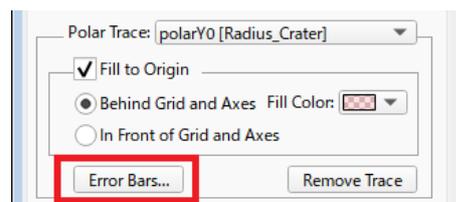
10 * sin(0.4 * x) + 110 + 0.1*gnoise(20) :

10 * sin(0.4 * x) + 90 + 0.1*gnoise(20)



Polar Error Bars ダイアログ

Igor Pro 6.33 で、極座標グラフコントロールパネルのメインタブにある Error Bars ボタンをクリックすると、Polar Error Bars ダイアログが開くようになりました。

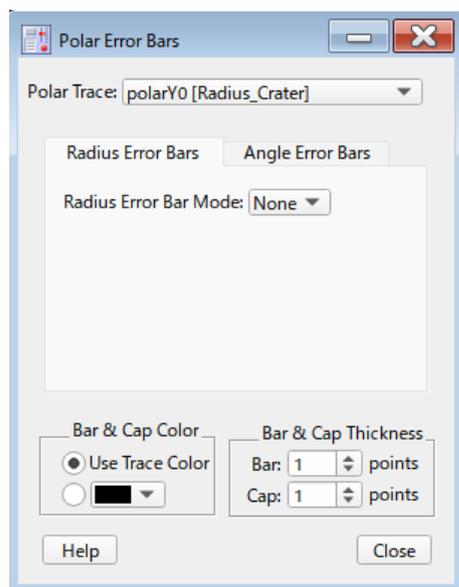


エラーバーは、極座標グラフの各トレースの特性であるため、別のトレースには異なるエラーバーが存在します。

Igor Pro 7.02 で、半径と角度のエラーバーが実装されました。極座標トレースは、半径と角度のエラーバーを任意に組み合わせることができます。

デフォルトでは、両方のエラーバーはオフ（None に設定）になっています。

（箱型の極座標エラーバーは実装されていません）



「Radius Error Bars」タブ

極半径エラーバーは通常のトレースエラーバーと類似しています。

Radius Error Bars:	意味
None	エラーバー無し
% of radius	[+/- 半径 * 指定された %] のエラーバー
sqrt of radius	[+/- sqrt(半径)] のエラーバー
+ constant	指定された定数 (+) のエラーバー
+/- constant	指定された定数 (+/-) のエラーバー
+/- wave	選択したウェーブからの値 (+/-) のエラーバー

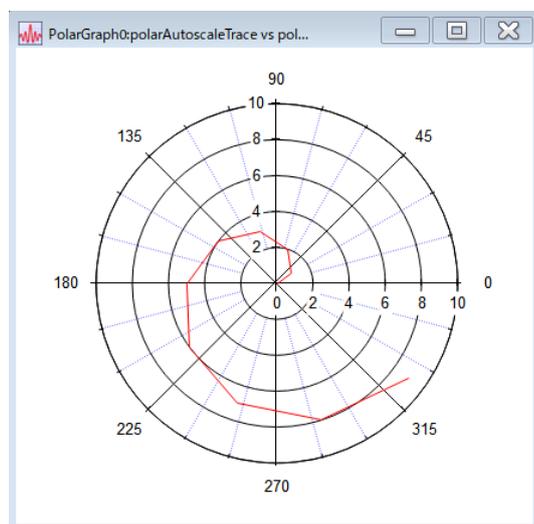
極半径エラーバーの例を示します。

新しい実験で次を実行します。

```
Make/N=10 exampleData = p
SetScale x 0, 2*PI, "rad", exampleData
```

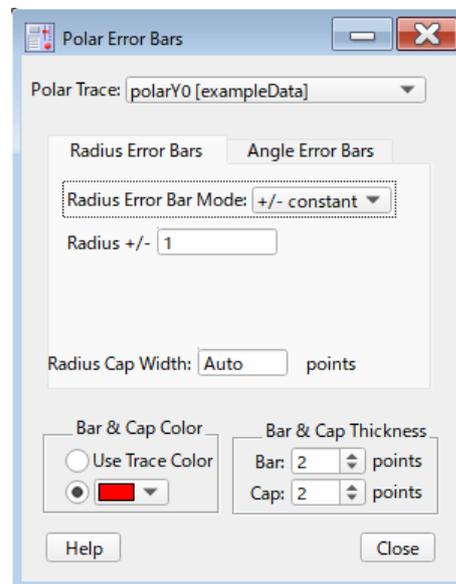
先に説明した手順で極座標にトレースを追加します。

Polar Graphs パネルで、Error Bars ボタンをクリックします。



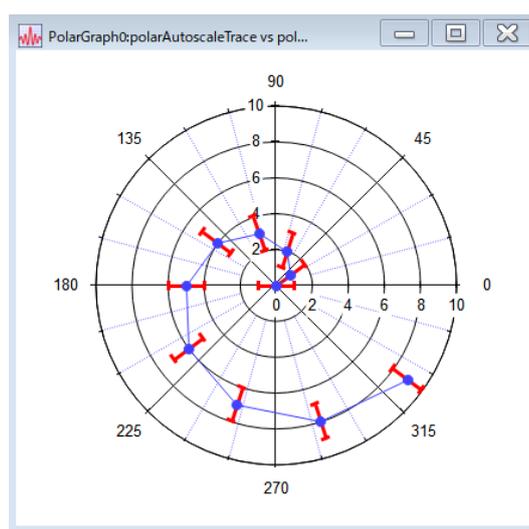
各項目を次のように設定します。

Radius Error Bar Mode: +/- constant
Radius +/-: 1
Bar & Cap Color: 赤
Bar & Cap Thickness: 両方を 2 ポイント



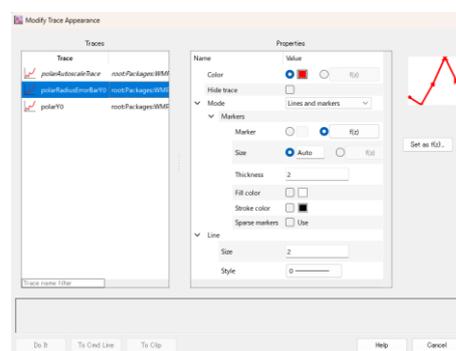
次に、トレースをダブルクリックして、Modify Trace Appearance ダイアログを表示して次のように設定します。

Color: 青
Mode: Lines and markers
Marker: 塗りつぶした円形



Modify Trace Appearance ダイアログを見ると、極座標エラーバーは Lines と $f(z)$ マーカーモードを使った polarRadiusErrorBarY0 vs polarRadiusErrorBarX0 トレースとして実装されていることがわかります。

エラーバー自体は、単に (NaN,NaN) のギャップを持つトレースラインです。



エラーバーのキャップは、カスタムマーカーのフック関数を使って描画されます（関連する関数は、New Polar Graphs.ipf 内の WMPolarErrorBarCapMarker という名前が付けられています）。

このトレースの外観は、自由にカスタマイズできますが、トレースモードとマーカー (f(z) というボタンで表示) は変更しないでください。

Polar Error Bars ダイアログでは、対応するトレースパラメーターを設定します。

Polar Error Bars 設定 トレースの設定

Cap Width

Marker Size (自動またはポイント単位)

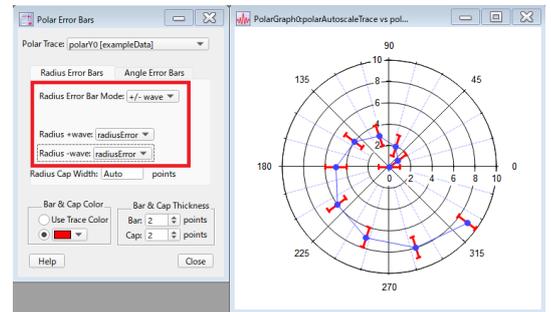
Cap Thickness	Marker Thick (ポイント単位)
Bar Thickness	Line Size (ポイント単位)
Bar & Cap Color	Trace Color

(これらの Modify Trace Appearance 設定を変更すると、Polar Error Bars ダイアログに反映されます)

+/- ウェーブモードを使っている場合は、半径ウェーブと同じ数の値を含むウェーブを使って、半径ごとのエラーを指定することができます。

例えば、Data → Make Waves で、10 ポイントの radiusError ウェーブを作成し、すべての値を 1 とします。

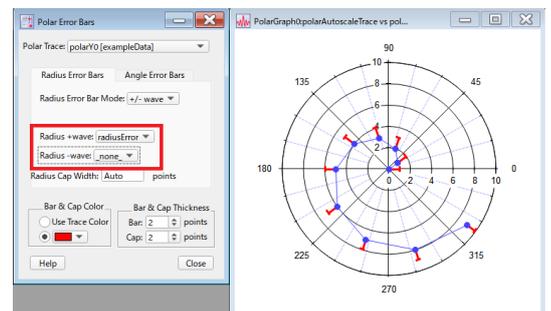
Radius Error Bar Mode を「+/- wave」に設定し、Radius +/- wave の両方をポップアップメニューから「radiusError」にすると、プロットは右図のようになります。



どちらかのウェーブに「_none_」を選択すると、上下のエラーバーの半分が省略されます。

半径が自動スケールされる場合、半径エラーウェーブは自動スケール計算に含まれます。

Delay Update チェックボックスがオフになっている場合、エラーウェーブに変更を加えると、それが使われている極座標グラフが自動的にスケールされます。



「Angle Error Bars」タブ

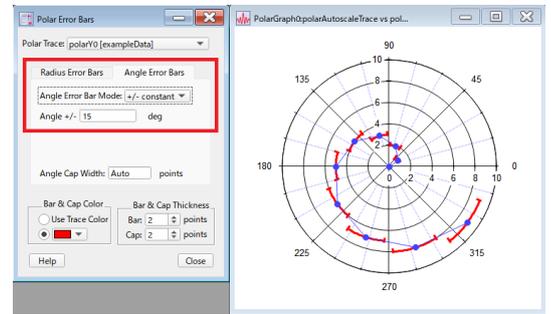
極角エラーバーは通常のトレースエラーバーとは少し異なり、角度の範囲は一定の半径の曲線で描画されます。

使うことができる極角エラーバーの選択肢は次の通りです。

Angle Error Bars:	意味
None	エラーバー無し
% of angle	[+/- 角度 * 指定された %] のエラーバー
sqrt of angle	[+/- sqrt(角度)] のエラーバー
+ constant	指定された定数[角度] (+) のエラーバー
+/- constant	指定された定数[角度] (+/-) のエラーバー
+/- wave	選択したウェーブからの値[角度] (+/-) のエラーバー

注記：最後の3つの選択肢の極角エラーは、角度データがラジアンであっても常に度単位で表示されます。

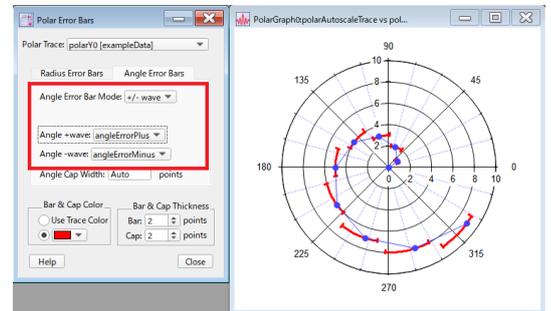
「+/- constant」で 15 度の極角エラーバーは次のようになります。



「+/- wave」モードを使う場合は、半径ウェーブと同じポイント数を含むウェーブを使って、角度ごとのエラーを指定することができます。

角度データがラジアン単位であっても、このウェーブではエラーを度単位で指定する必要があることを覚えておいてください。

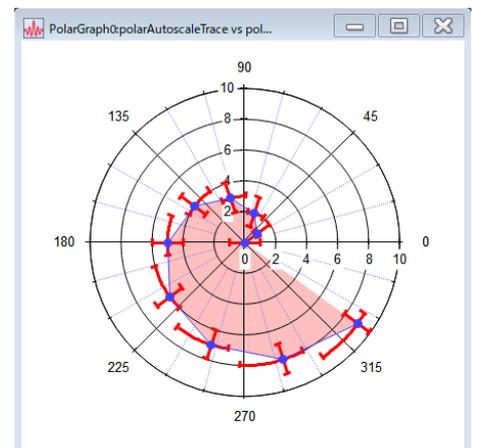
例えば、Data → Make Waves で、10 ポイントの angleErrorPlus と angleErrorMinus の 2 つのウェーブを作成し、angleErrorPlus の値を 20、angleErrorMinus の値を 10 とします。



Radius Error Bar Mode を「+/- wave」に設定し、Radius +/- wave の両方をポップアップメニューから選択すると、プロットは右図のようになります。

各トレースに、角度と半径の両方のエラーバーを表示することができます。

(右図は Fill to Origin で着色)



極座標系の場合、自動の凡例は 2 次的なものになる可能性があります。

エラーバーのトレースを凡例から削除するか、凡例を次のように変更することをお勧めします。

Graph メニューから Add Annotation (または Edit Annotation) を選択し、Add Annotation ダイアログを開きます。

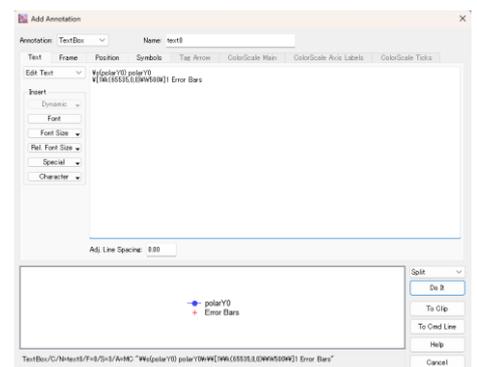
テキストボックスに次を入力します。

¥s(polarY0) polarY0

¥[1¥k(65535,0,0)¥W500¥]1 Error Bars

Do It をクリックすると凡例が追加されます。

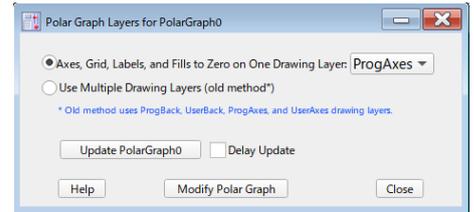
マウスでドラッグして適切な位置に移動します。



Polar Layers ダイアログ

「Main」タブの Layers ボタンをクリックすると、極座標グラフの要素を描画する方法を選択する Polar Graph Layers ダイアログが表示されます。

Polar Graph パッケージのバージョン 9.03 以降では、極座標グラフの要素は2つの異なる方法で描画できます。



このダイアログでは、極座標グラフの要素を描画するための2つの方法のうちの1つを選択します。

1. Axes, Grid, Labels, and Fills to Zero on One Drawing Layer
(軸、グリッド、ラベル、ゼロまでの塗りつぶし、を1つのレイヤーに描画)
2. Use Multiple Drawing Layers (old method*)
(複数の描画レイヤー [ProgBack, UserBack, ProgAxes, UserAxes] を使う)

Axes, Grid, Labels, and Fills to Zero on One Drawing Layer が、バージョン 9.03 からのデフォルト実装です。

Axes, Grid, Labels, and Fills to Zero on One Drawing Layer の詳細

描画レイヤー全体を再描画する代わりに、描画する「仮想レイヤー」を定義するために、名前の付いたグループオブジェクトを使います。

背面から前面へ順に、選択可能な描画レイヤーが、次の目的で使われる名前付きグループオブジェクトで埋められます。

仮想レイヤー (グループ名)	使用
gridBackground	オプションのグリッド背景
fillToZeroBack	グリッドと軸の後ろで原点まで塗りつぶす
polarGrid	極座標グリッド
polarAxes	極座標軸
polarAxesLabels	極座標軸ラベル
fillToZeroFront	グリッドと軸の前で原点まで塗りつぶす

これらの仮想レイヤーのすべてに描画オブジェクトが含まれていない場合でも、グループ名は再作成マクロ内に後ろから前の順で表示されます。

従来の方とは異なり、任意の描画レイヤーにある独自のオブジェクト（極座標グラフ要素を描画するために使われるレイヤーにあるオブジェクトを含む）はそのまま残されます。

極座標グラフ要素を描画する描画レイヤーを自由に選択できるため、画像プロットの前面に極座標軸を描画するのが簡単になります（ProgAxes は、画像プロットの前面にある最も下のレイヤーです）。

Multiple Drawing Layers (old method) の詳細

この古い方法は、極座標グラフが最大4つの異なる描画レイヤー上のコンテンツを完全にコントロールすることを前提としています。

通常とは逆の順序で、複数の描画レイヤーが次のように使われています。

グラフィケーター	使用
(ウィンドウの背景)	
ProgBack	グリッドと軸の後ろで原点まで塗りつぶす オプションでグリッドの背景
UserBack	極座標グリッドと極座標軸
(軸と画像)	
ProgAxes	極座標軸ラベル
UserAxes	グリッドと軸の前で原点まで塗りつぶす
(トレース)	
ProgFront	ユーザー用にリザーブ
UserFront	ユーザー用にリザーブ
(注釈)	

この古い方法を使うと、デフォルトの UserFront レイヤーまたは ProgFront レイヤーに描画したものは、Polar Graph パッケージによって変更することができません。

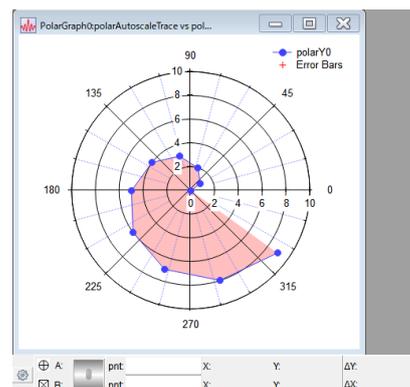
極座標グラフが更新されるたびに、ユーザー用にリザーブされていないレイヤーはすべて消去され、再描画されます。

この場合、これらのレイヤーに加えた変更はすべて失われます。

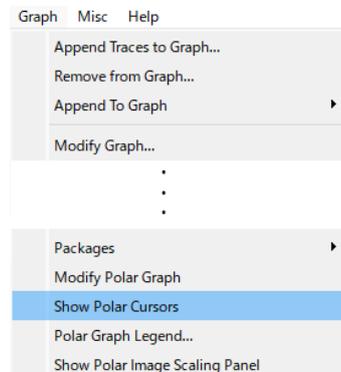
そのため、これを防ぐには新しい方法を使うことです。

極座標のカーソル

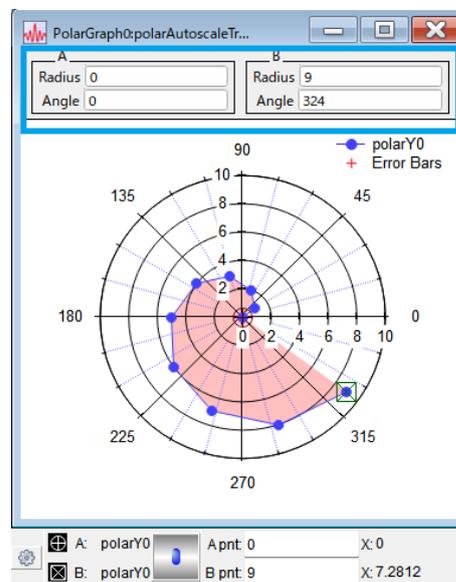
グラフの下にある通常のグラフカーソル (メニュー Graph → ShowInfo / 詳細はマニュアルの情報パネルとカーソルの項を参照) は、カーソルの X 座標と Y 座標のみを表示します。



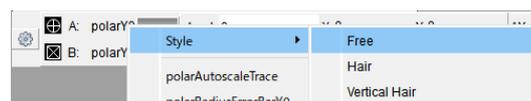
カーソル位置の角度と半径を表示するには、メニュー Graph → Show Polar Cursors を選択します。



すると、カーソル位置の半径と角度を表示する ValDisplay コントロールが追加されます。



カーソルを極座標トレースから解放するには、情報パネルのトレース名を右クリックし、コンテキストメニューから Free を選択します。



開放してもカーソル位置の半径と角度の値は表示されます。

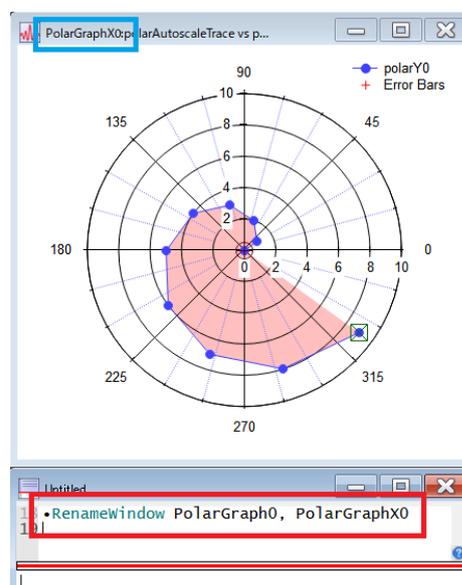
極座標グラフの名前の変更

作成済みの極座標グラフの名前を変更しても、関連付けられたグラフの設定と接続は失われません。

ただし、関連付けは古いグラフ名（「PolarGraph0」など）で保持されています。

```
RenameWindow PolarGraph0, PolarGraphX0
```

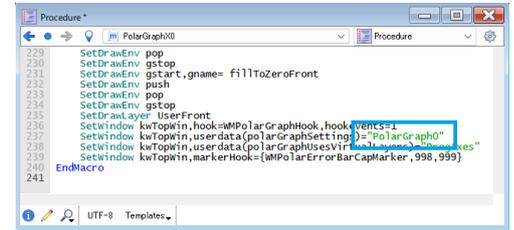
グラフのタイトルを変更することもできます。



極座標グラフの関連付けられた名前は、グラフの再作成マクロで確認できます。

メニュー Windows → Procedure Windows → Procedure Window を開いて、SetWindow コマンドの userdata で確認できます。

これは、グラフの極座標グラフ名（およびデータフォルダー名）を PolarGraph0 として識別します。



再作成マクロとして極座標グラフを保存

ウィンドウの再作成マクロが保存されている場合、閉じたグラフの極座標グラフの設定は削除されません。

Polar Graphs パッケージから極座標グラフを切り離す

Polar Graphs パッケージのグラフとの接続を切り離すには、次のコマンドを入力します：

```
WMPolarDisconnectGraph("")
```

これにより、最前面の極座標グラフの接続が切断されます。

最前面の極座標グラフではない極座標グラフのグラフ名を指定することもできます。

切断されたグラフのサイズを変更しても、Polar Graphs パッケージの描画レイヤーがクリアされ、再描画されることはありません。

そのため、切断後にグラフのサイズを変更すると、結果が望ましいものではない場合があります。

Polar Graphs パッケージからグラフが切断されているかどうかを簡単に判断するには Graph メニューを使います。

メニューに Modify Polar Graph が表示されない場合は、グラフが切断されています。

次のコマンドで、接続を復元できます：

```
WMPolarReconnectGraph("", "")
```

最初のパラメーターにグラフ名、2番目のパラメーターにデータフォルダー名を指定することができます。

データフォルダー名は、グラフ名を変更しない限り、通常はグラフ名と同じです。

より確実なプログラミングの例を示します。

```
// 最前面の極座標グラフを切断
```

```
String graphName = WMPolarTopPolarGraph()
```

```
String dfName= WMPolarDisconnectGraph(graphName)
```

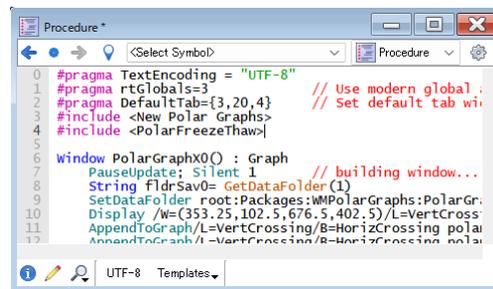
```
... // ここで何かを処理をする
```

```
// Polar Graph パッケージにグラフを再接続
```

```
WMPolarReconnectGraph(graphName, dfName)
```

極座標グラフを切断するより便利な方法として、極座標グラフを「凍結 (freeze)」または「解凍 (thaw)」するためのメニュー項目を追加する <PolarFreezeThaw> タグを使う方法があります。

保存した再作成マクロを開き、最初の部分に
#include <PolarFreezeThaw>
を追加します。
画面下部の Compile ボタンをクリックします。



```
0 #pragma TextEncoding = "UTF-8"  
1 #pragma rtGlobals=3  
2 #pragma DefaultTab={3,20,4} // Use modern global :  
3 #include <New Polar Graphs> // Set default tab wi  
4 #include <PolarFreezeThaw>  
5  
6 Window PolarGraphX0 : Graph // building window...  
7 PauseUpdate; Silent 1  
8 String fldrSav0 = GetDataFolder(1)  
9 SetDataFolder root:Packages:WM:Polar Graphs:Polar Gr  
10 Display /W=(353.25,102.5,676.5,402.5)/L=VertCross  
11 AppendToGraph/L=VertCrossing/B=HorizCrossing polai  
12 AppendToGraph/L=VertCrossing/B=HorizCrossing polai
```

グラフの再作成マクロを実行して、グラフを再描画します。

すると、Graph メニューに次の項目が追加されます：
Freeze top Polar Graph（最前面の極座標グラフを凍結）
Freeze all Polar Graphs（すべての極座標グラフを凍結）
Thaw all Polar Graphs（すべての極座標グラフを解冻）

「凍結」は、グラフを Polar Graphs パッケージから切り離し、
グラフのサイズを固定します。

極座標グラフを「解冻」すると、Polar Graphs パッケージに再接
続されます。

