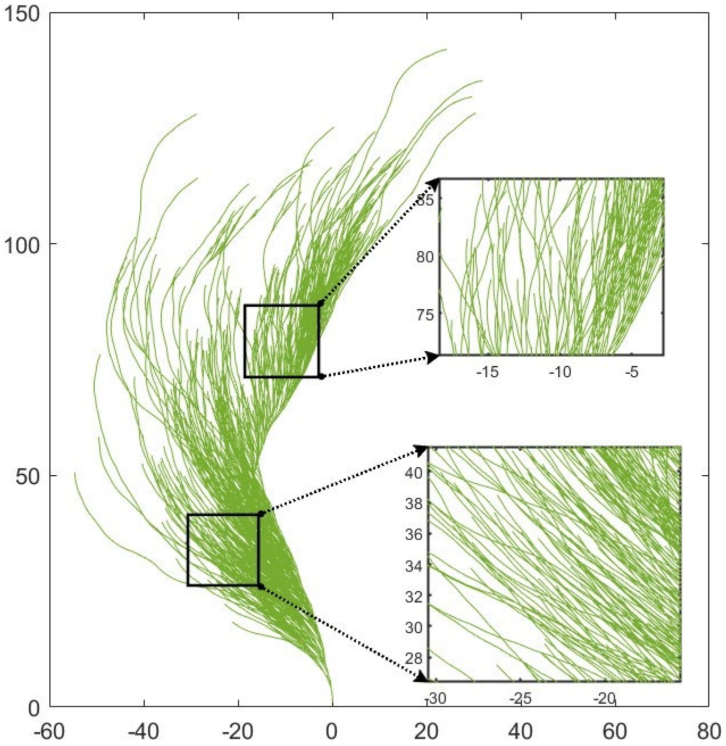


ZoomPlot

MATLAB Code for Interactive Magnification of Customized Regions

Version 1.5.1, 5-FEB-2024

Email: iqiukp@outlook.com

SYNTAX

```
zp = BaseZoom()

zp = BaseZoom(ax)

zp = BaseZoom(ax, subAxesPosition, zoomAreaPosition)

zp = BaseZoom(ax, zoomAreaPosition)
```

DESCRIPTION

The BaseZoom class provides interactive, customizable magnification of local regions in MATLAB figures and images by allowing users to select a subsection for zoomed display in a child axes.

```
zp = BaseZoom()
```

This command instantiates a BaseZoom object which, by default, is linked to the current active plotting axes, as returned by the gca function. There is no need to specify a figure or image class instance beforehand; the BaseZoom object will automatically apply to the last plotting action performed. Users can utilize mouse actions within an interactive environment to select and zoom into the desired area for inspection.

```
zp = BaseZoom(ax)
```

This command initializes a BaseZoom instance on the ax instance of the figure or image class, allowing users to choose the specific figure or image they want to work with in an interactive mode. The position of the sub-axes and the zoom area can be dynamically determined through mouse operations. This method does not require users to provide specific location parameters upfront but offers an intuitive way to make selections via the graphical user interface, enabling users to select and zoom into specific areas as needed during the application's execution.

```
zp = BaseZoom(ax, subAxesPosition, zoomAreaPosition)
```

This command is designed for the ax instance of the figure class to initialize a BaseZoom instance. Both subAxesPosition and zoomAreaPosition are four-element vectors (formatted as [left bottom width height]), where subAxesPosition defines the position and size of the sub-axes within the figure, and zoomAreaPosition defines the position and size of the area within the sub-axes to be magnified for closer viewing. It is important to note that both parameters are absolute coordinates, meaning their values correspond directly to the current plotting data range.

```
zp = BaseZoom(ax, zoomAreaPosition)
```

This command is suitable for the ax instance of the image class to initialize a BaseZoom instance. zoomAreaPosition is a four-element vector (formatted as [x y width height]), defining the pixel coordinate location and size of the image area to be magnified within ax. The pixel coordinates refer to the specific positions within the image data, which are related to the actual displayed size of the image.

FILES

- BaseZoom.m: This is the core code file that contains the main functionality for the zoom feature.
- parameters.json: This file serves as the configuration file for plotting parameters, allowing customization of the visual appearance of the ZoomPlot.
- manual.pdf: This is a help document that provides detailed

instructions for usage. It likely includes information on how to integrate and use the BaseZoom class within MATLAB, explanations of the various parameters and settings found in the parameters.json file.

REQUIREMENTS

- Version 1.5 is only compatible with MATLAB release R2018b and later versions.
- The Image Processing Toolbox must be installed.

PREPARATIONS

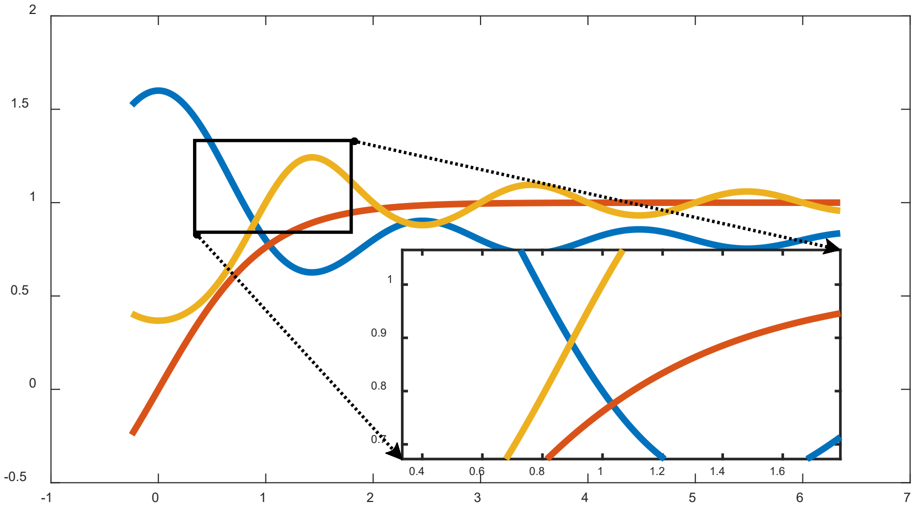
Add `BaseZoom.m` and `parameters.json` to MATLAB's search path or to the working directory where the plotting code resides.

EXAMPLES

Interactive Local Magnification for Figure Class

Create a zoomable area within the current axes interactively:

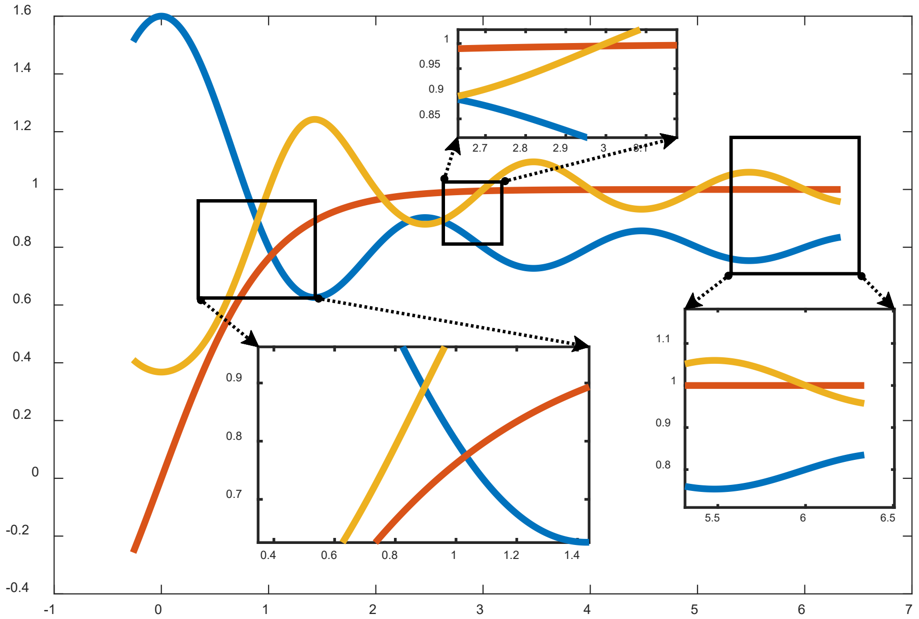
```
x = linspace(-0.1*pi,2*pi, 1000)+rand(1)*0.2;
y = cell(1, 3);
y{1, 1} = 0.8*sinc(x)+0.8;
y{1, 2} = tanh(x);
y{1, 3} = exp(-sinc(x));
figure
hold on
for i = 1:3
    plot(x, y{1, i}, 'LineWidth', 3)
end
box on
zp = BaseZoom();
zp.run;
```



Implement Multiple Local Magnifications for Figure Class

Set up multiple zoomable areas within the same figure:

```
x = linspace(-0.1*pi,2*pi, 1000)+rand(1)*0.2;
y = cell(1, 3);
y{1, 1} = 0.8*sinc(x)+0.8;
y{1, 2} = tanh(x);
y{1, 3} = exp(-sinc(x));
figure
hold on
for i = 1:3
    plot(x, y{1, i}, 'LineWidth', 3)
end
box on
zp = BaseZoom();
zp.run;
zp.run;
zp.run;
```

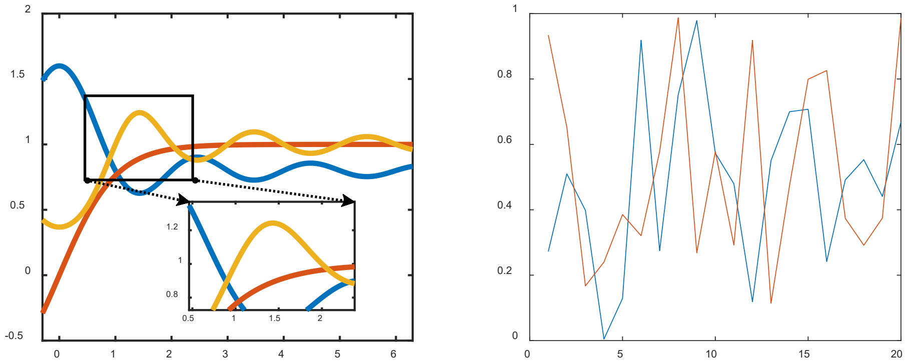


Specify Axes for Local Magnification for Figure Class

Define a zoomable area within a specific set of axes:

```
x = linspace(-0.1*pi,2*pi, 1000)+rand(1)*0.2;
y = cell(1, 3);
y{1, 1} = 0.8*sinc(x)+0.8;
y{1, 2} = tanh(x);
y{1, 3} = exp(-sinc(x));
figure

% axes 1
subplot(1, 2, 1)
hold on
for i = 1:3
    plot(x, y{1, i}, 'LineWidth', 3)
end
box on;
set(gca, 'LineWidth', 1.2, 'TickDir', 'in', 'YScale', 'linear');
ax_1 = gca;

% axes 2
subplot(1, 2, 2)
plot(rand(20,2))
ax_2 = gca;
```

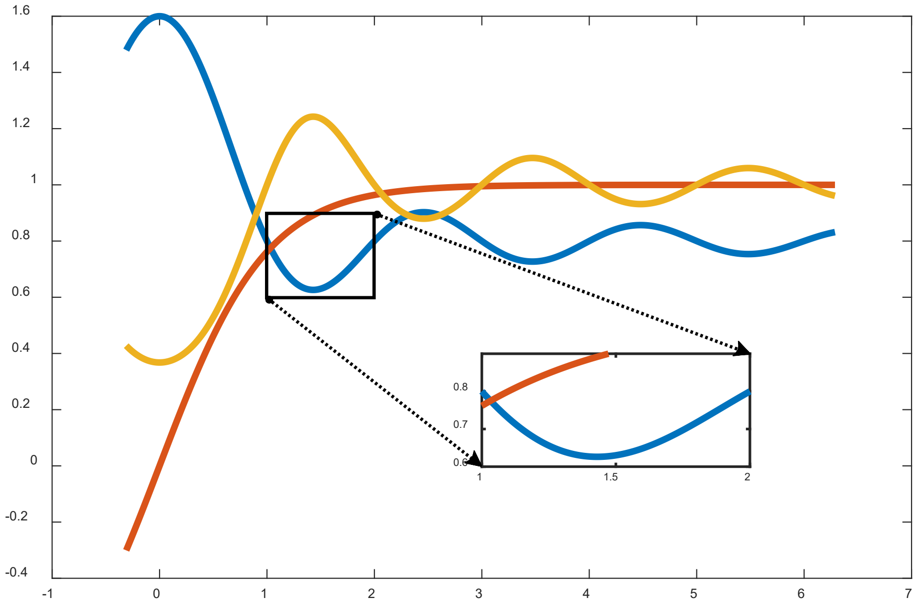
```
zp = BaseZoom(ax_1);
zp.run;
```



Manually Set SubAxes and Zoom Area for Figure Class

Create a zoomable area within a specific axes object and define both the subplot display and the zoom region manually:

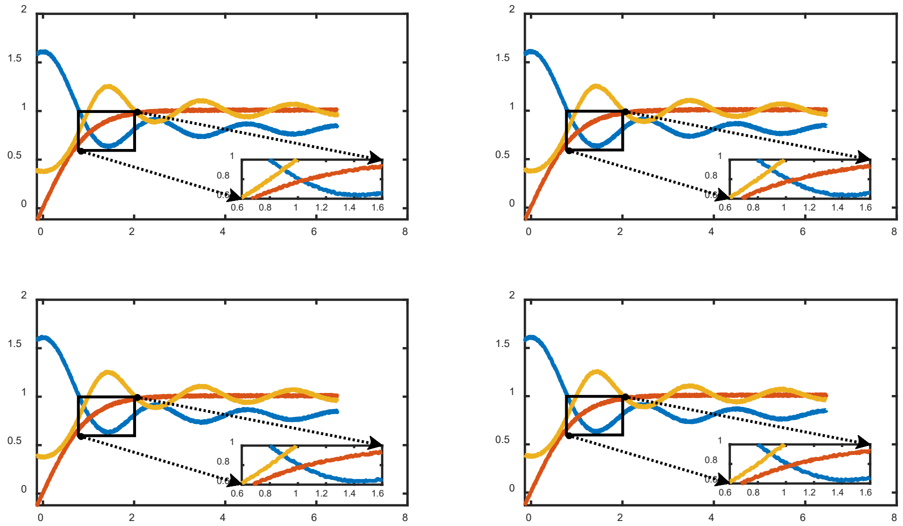
```
x = linspace(-0.1*pi,2*pi, 1000)+rand(1)*0.2;
y = cell(1, 3);
y{1, 1} = 0.8*sinc(x)+0.8;
y{1, 2} = tanh(x);
y{1, 3} = exp(-sinc(x));
figure
hold on
for i = 1:3
    plot(x, y{1, i}, 'LineWidth', 3)
end
box on;

% zoom plot
subAxesPosition = [3, 0, 2.5, 0.4];
zoomAreaPosition = [1, 0.6, 1, 0.3];
zp = BaseZoom(subAxesPosition, zoomAreaPosition);
zp.run;
```



Manually ZoomPlot in Sub Plots for Figure Class

Set up a zoomable area within specified sub plots by manually defining the display parameters and zoom region:

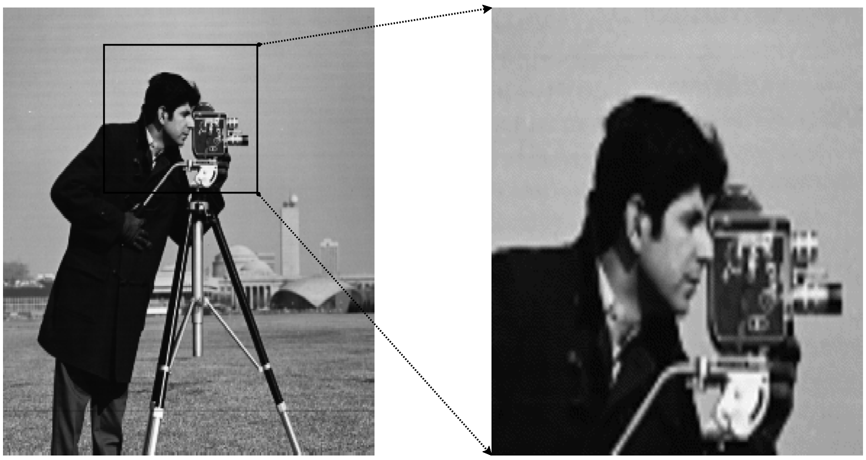
```
x = linspace(-0.1*pi,2*pi, 1000)+rand(1)*0.2;
y = cell(1, 3);
y{1, 1} = 0.8*sinc(x)+0.8;
y{1, 2} = tanh(x);
y{1, 3} = exp(-sinc(x));
ax = cell(4, 1);
figure
for j = 1:4
    subplot(2, 2, j)
    hold on
    for i = 1:3
        plot(x, y{1, i}+0.02*rand(1, length(x)), 'LineWidth', 2)
    end
    box on;
    set(gca, 'LineWidth', 1.2, 'TickDir', 'in', 'YScale', 'linear');
    ax{j, 1} = gca;
    subAxesPosition = [3.5, 0.1, 2.5, 0.4];
    zoomAreaPosition = [0.6, 0.6, 1, 0.4];
    zp = BaseZoom(ax{j, 1}, subAxesPosition, zoomAreaPosition);
    zp.run;
end
```



Interactive Local Magnification for Image Class

Initialize an interactive zoomable region within the current image displayed on the axes:

```
[X, cmap] = imread('cameraman.tif');
imshow(X, cmap);
zp = BaseZoom();
zp.run;
```

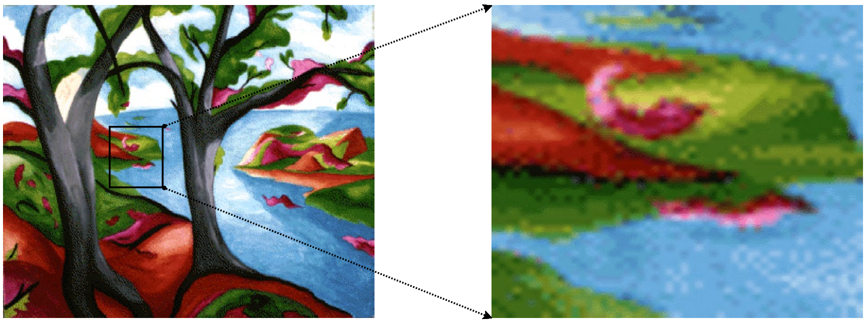


Manually Set Zoom Area for Image Class

Configure a specific zoomable region within an image displayed on the axes:

```
[X, cmap] = imread('trees.tif');
```

```
imshow(X, cmap);
zoomAreaPosition = [100, 100, 50, 50];
zp = BaseZoom(gca, zoomAreaPosition);
zp.run;
```



PARAMETERS

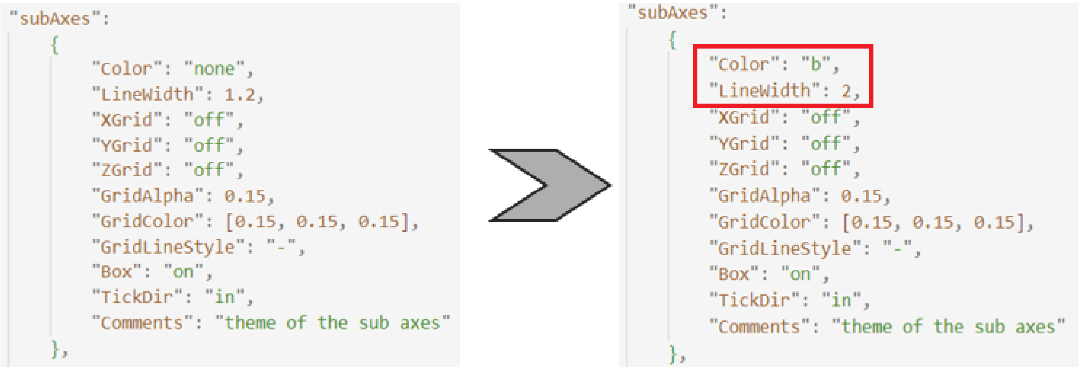
The JSON file contains configuration parameters for the ZoomPlot feature, allowing users to customize the theme of ZoomPlot by modifying the default settings.

Sub Axes Theme

The "subAxes" section sets the appearance of the sub-axes, including the background color, line width, grid display, grid transparency and color, border display, tick direction, and more.

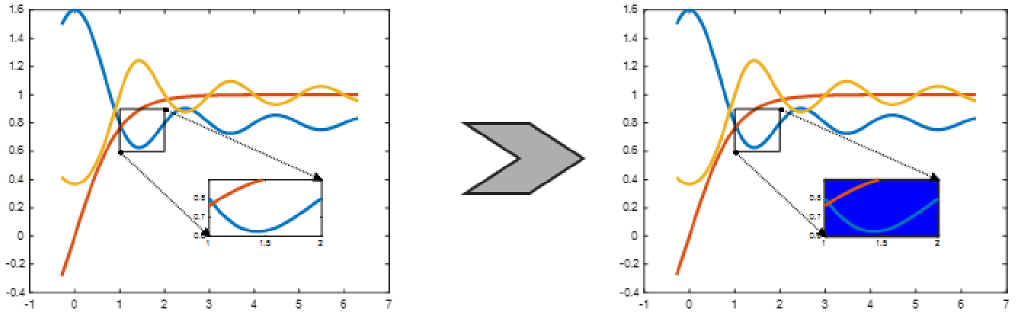
```
    "subAxes":
        {
            "Color": "none",
            "LineWidth": 1.2,
            "XGrid": "off",
            "YGrid": "off",
            "ZGrid": "off",
            "GridAlpha": 0.15,
            "GridColor": [0.15, 0.15, 0.15],
            "GridLineStyle": "-",
            "Box": "on",
            "TickDir": "in",
            "Comments": "theme of the sub axes"
        }
```

For instance, changing the background of the sub-axes to blue and the line width to 2:



```
"subAxes":
    {
        "Color": "none",
        "LineWidth": 1.2,
        "XGrid": "off",
        "YGrid": "off",
        "ZGrid": "off",
        "GridAlpha": 0.15,
        "GridColor": [0.15, 0.15, 0.15],
        "GridLineStyle": "-",
        "Box": "on",
        "TickDir": "in",
        "Comments": "theme of the sub axes"
    },
```

```
"subAxes":
    {
        "Color": "b",
        "LineWidth": 2,
        "XGrid": "off",
        "YGrid": "off",
        "ZGrid": "off",
        "GridAlpha": 0.15,
        "GridColor": [0.15, 0.15, 0.15],
        "GridLineStyle": "-",
        "Box": "on",
        "TickDir": "in",
        "Comments": "theme of the sub axes"
    },
```

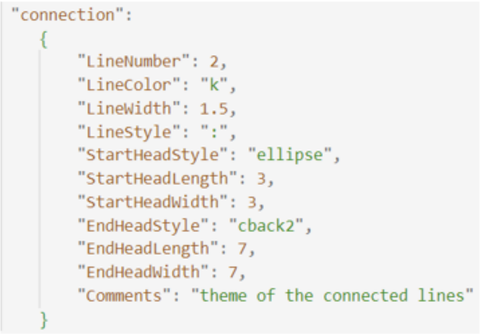


Zoomed Area Theme

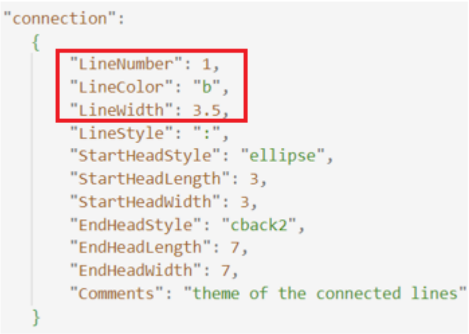
The "zoomedArea" defines the theme for the zoomed area's border color, fill color and style, and border line width.

```
    "zoomedArea":
        {
            "Color": "k",
            "FaceColor": "none",
            "FaceAlpha": 0,
            "LineStyle": "-",
            "LineWidth": 1.5,
            "Comments": "theme of the zoomed area"
        }
```

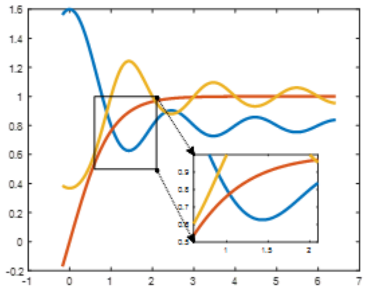
For example, changing the fill color of the zoom area to [0, 0.4471, 0.7412], the facealpha to 0.8, and the line width to 3.5:

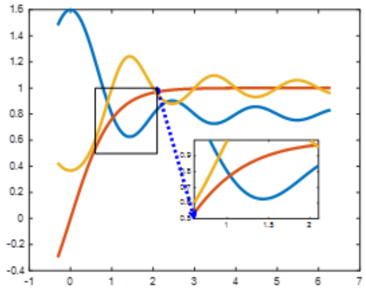


```
"connection":
    {
        "LineNumber": 2,
        "LineColor": "k",
        "LineWidth": 1.5,
        "LineStyle": ":",
        "StartHeadStyle": "ellipse",
        "StartHeadLength": 3,
        "StartHeadWidth": 3,
        "EndHeadStyle": "cback2",
        "EndHeadLength": 7,
        "EndHeadWidth": 7,
        "Comments": "theme of the connected lines"
    }
```



```
"connection":
    {
        "LineNumber": 1,
        "LineColor": "b",
        "LineWidth": 3.5,
        "LineStyle": ":",
        "StartHeadStyle": "ellipse",
        "StartHeadLength": 3,
        "StartHeadWidth": 3,
        "EndHeadStyle": "cback2",
        "EndHeadLength": 7,
        "EndHeadWidth": 7,
        "Comments": "theme of the connected lines"
    }
```



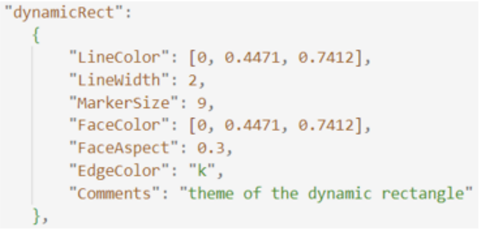


Dynamic Rectangle Theme

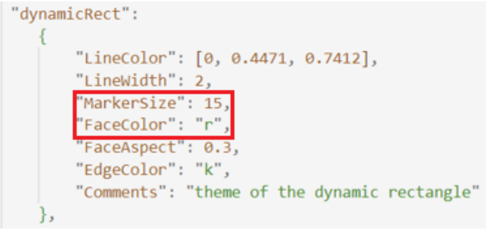
The "dynamicRect" is used for setting the theme of the dynamic rectangle, including the line color, line width, marker style, size, and edge color.

```
    "dynamicRect":
        {
            "LineColor": [0, 0.4471, 0.7412],
            "LineWidth": 2,
            "Marker": "s",
            "MarkerSize": 9,
            "FaceColor": [0, 0.4471, 0.7412],
            "FaceAspect": 0.3,
            "EdgeColor": "k",
            "Comments": "theme of the dynamic rectangle"
        }
```

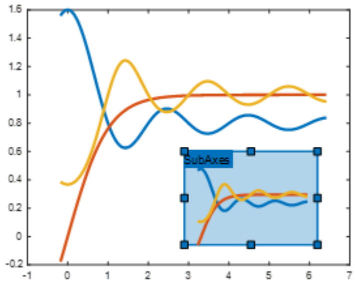
For example, changing the size of the dynamic rectangle's corner handles to 15 and the fill color to red:

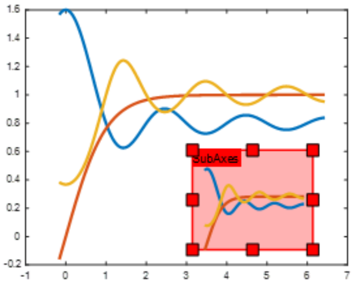


```
"dynamicRect":
    {
        "LineColor": [0, 0.4471, 0.7412],
        "LineWidth": 2,
        "MarkerSize": 9,
        "FaceColor": [0, 0.4471, 0.7412],
        "FaceAspect": 0.3,
        "EdgeColor": "k",
        "Comments": "theme of the dynamic rectangle"
    },
```



```
"dynamicRect":
    {
        "LineColor": [0, 0.4471, 0.7412],
        "LineWidth": 2,
        "MarkerSize": 15,
        "FaceColor": "r",
        "FaceAspect": 0.3,
        "EdgeColor": "k",
        "Comments": "theme of the dynamic rectangle"
    },
```



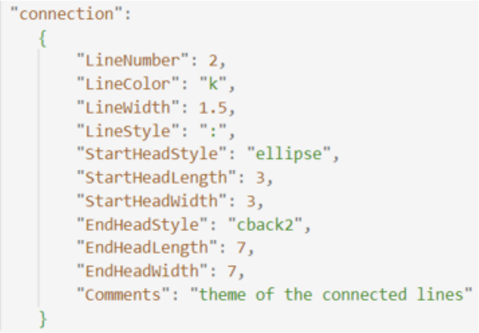


Connection Lines Theme

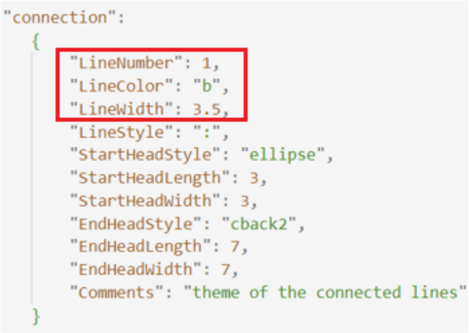
The "connection" includes the style settings for the connecting lines, such as line number, color, width, style, and the arrowhead's style, length, and width. For instance, the number of connection lines can be set to 0, 1, or 2. Setting it to 0 means there will be no connecting lines between the zoomed area and the sub-axes. If set to 1, there will be only one connecting line between them. When set to 2, there will be two connecting lines (which is the default configuration).

```
    "connection":
       {
           "LineNumber": 2,
           "LineColor": "k",
           "LineWidth": 1.5,
           "LineStyle": ":",
           "StartHeadStyle": "ellipse",
           "StartHeadLength": 3,
           "StartHeadWidth": 3,
           "EndHeadStyle": "cback2",
           "EndHeadLength": 7,
           "EndHeadWidth": 7,
           "Comments": "theme of the connected lines"
       }
```

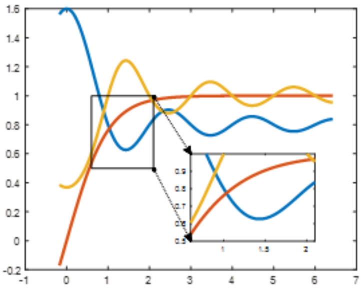
For example, change the number of connecting lines to 1, change the color to blue, and change the line width to 3.5:

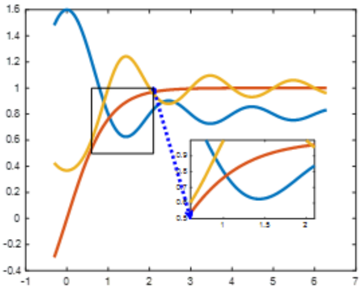


```
"connection":
    {
        "LineNumber": 2,
        "LineColor": "k",
        "LineWidth": 1.5,
        "LineStyle": ":",
        "StartHeadStyle": "ellipse",
        "StartHeadLength": 3,
        "StartHeadWidth": 3,
        "EndHeadStyle": "cback2",
        "EndHeadLength": 7,
        "EndHeadWidth": 7,
        "Comments": "theme of the connected lines"
    }
```



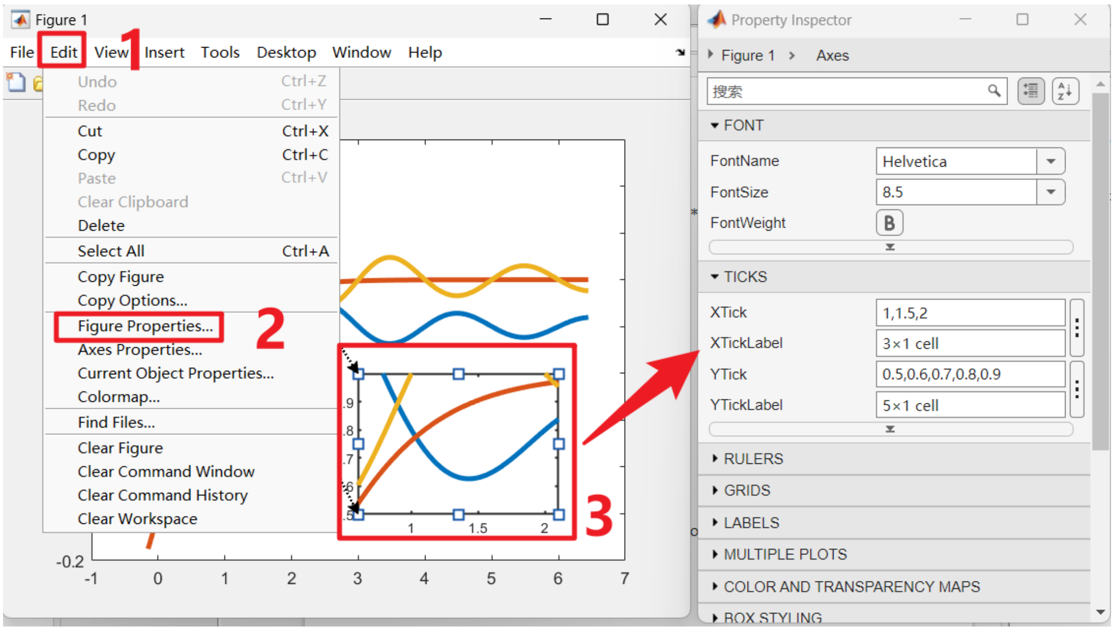
```
"connection":
    {
        "LineNumber": 1,
        "LineColor": "b",
        "LineWidth": 3.5,
        "LineStyle": ":",
        "StartHeadStyle": "ellipse",
        "StartHeadLength": 3,
        "StartHeadWidth": 3,
        "EndHeadStyle": "cback2",
        "EndHeadLength": 7,
        "EndHeadWidth": 7,
        "Comments": "theme of the connected lines"
    }
```





More Parameter Configuration

More parameter configurations can be modified through the figure properties window of MATLAB:



FontName Helvetica
FontSize 8.5
FontWeight B
XTick 1,1.5,2
XTickLabel 3×1 cell
YTick 0.5,0.6,0.7,0.8,0.9
YTickLabel 5×1 cell