Issue #7338

Add scrollbars when needed to preserve specified figure size

Feature Descriptions

Currently, if a figure size is explicitly specified, but is larger than the display area (either due to the figure being larger than the display screen, or the containing window being smaller), the bottom and right boundaries of the figure gets cropped out. In order to solve this problem, issue #7338 suggests the addition of vertical and horizontal scroll bars to allow users to pan around the figure when the bottom and right edges would otherwise overflow the window.

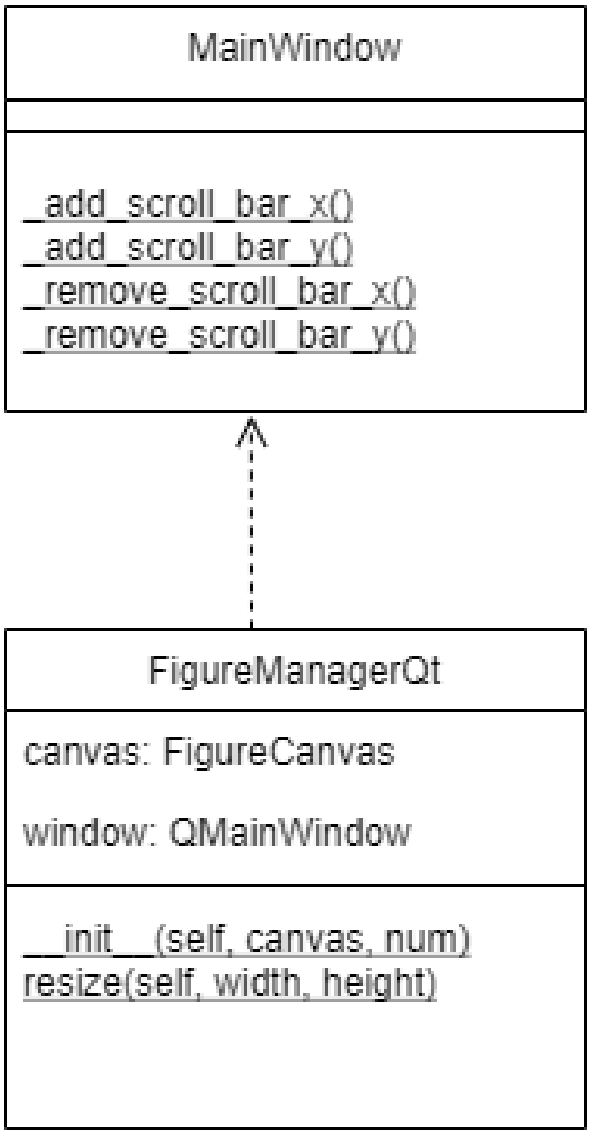
Implementation Plans

Code Organization

The following modifications will occur in backend_qt5.py:

- In the initialization of FigureManagerQT, we will check if the window is large enough to contain the canvas after the initial window resize. If not, we will add scroll bars to the widow using _add_scroll_bar_x() and _add_scroll_bar_y(), depending on weather a horizontal and/or vertical scroll bar is needed.
- In the function resize, we will check if after resizing, the window can fully contain the canvas.
    - If it can, the methods _remove_scroll_bar_x() and/or _remove_scroll_bar_y() will be called to remove the appropriate pre-existing scroll bar(s).
    - If not, we will call _add_scroll_bar_x() and/or _add_scroll_bar_y().
- We will add the following (4) custom methods to MainWindow:
    - _add_scroll_bar_x() and _add_scroll_bar_y()
        - These methods will be responsible for adding the horizontal and vertical scroll bars (respectively) from a MainWindow.
    - _remove_scroll_bar_x() and _remove_scroll_bar_x()
        - These methods will remove the horizontal and vertical scroll bars (respectively) from a MainWindow.
- Additionally, the methods will give an error if a scroll bar already exists, or there is none to remove, when adding or removing scroll bars respectively.

UML:


MainWindow
_add_scroll_bar_x()
_add_scroll_bar_y()
_remove_scroll_bar_x()
_remove_scroll_bar_y()
FigureManagerQt
canvas: FigureCanvas
window: QMainWindow
__init__(self, canvas, num)
resize(self, width, height)