#General Functions

"""

plot\_basic\_elements(model,ax,pumps=True,valves=True,reservoirs=True,

tanks=True,links=True,nodes=True,savefig=False, save\_name=None, legend=True)

Description

-----------

Plotting function that only plots elements without any data attached to them.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

pumps: Takes boolean. Determines whether to draw pumps or not.

valves: Takes boolean. Determines whether to draw valves or not.

reservoirs: Takes boolean. Determines whether to draw reservoirs or not.

tanks: Takes boolean. Determines whether to draw tanks or not.

links: Takes boolean. Determines whether to draw links or not.

nodes: Takes boolean. Determines whether to draw nodes or not.

savefig: Takes boolean. Determines whether the figure is saved or not.

save\_name: Takes string. If not defined, default save name will be the name of the network. IF save\_name is defined, it will prepend a suffix onto the network name to act as the name of the .png file. For example save\_name=’Pressure’ on the network ‘CTown.inp’ will result in a file called ‘PressureCTown.png’

legend: Takes boolean. Determines whether to draw legend or not.

plot\_discrete\_nodes(model,ax,bin\_edge\_num=5,parameter=None, value=None, get\_tanks=False,get\_reservoirs=False,bins='automatic', bin\_size\_list = None, bin\_shape\_list = None,bin\_label\_list = None, bin\_border\_list = None, bin\_border\_width\_list = None, savefig=True, tanks=True, reservoirs=True, pumps=True, valves=True,legend=True,legend\_title = None,

legend\_loc\_1='upper right', legend\_loc\_2='lower right',save\_name=None, cmap='tab10', color\_list=None)

Description

-----------

Plotting function for discrete node data.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

bin\_edge\_num: Takes integer. Number of bin edges to be created.

parameter: Takes string. Name of parameter, refer to list in get\_parameter() description.

value: Takes string or integer. This parameter is relevant when working with parameters that use timesteps. Takes ‘max’ ‘min’ ‘mean’ or an integer that represents the timestep index.

get\_tanks: Takes boolean. Determines if parameter is collected for tanks.

get\_reservoirs: Takes boolean. Determines if parameter is collected for reservoirs.

bins: Takes string or 1D array. If set to ‘automatic’ bins will be created in accordance to the parameter bin\_edge\_num. If array is provided, it’s length should be equal to the parameter bin\_edge\_num.

bin\_size\_list: Takes list. List of each bin’s node size. Length of list should be equal to number of bins.

bin\_shape\_list: Takes list. List of each bin’s marker shape. Length of list should be equal to number of bins.

bin\_label\_list: Takes list. Label that each bin will use if legend is drawn. By default is set to be equal to the bin range. Length of list should be equal to number of bins.

bin\_border\_list: Takes list. List of each bin’s node border color. Length of list should be equal to number of bins.

bin\_border\_width\_list: Takes list. List of each bin’s node border width. Length of list should be equal to number of bins.

savefig: Takes boolean. Determines whether the figure is saved or not.

tanks: Takes boolean. Determines whether to draw tanks or not.

reservoirs: Takes boolean. Determines whether to draw reservoirs or not.

pumps: Takes boolean. Determines whether to draw pumps or not.

valves: Takes boolean. Determines whether to draw valves or not.

legend: Takes boolean. Determines whether to draw legend or not.

color\_list: Takes list. List of each bin’s node color. Length of list should be equal to number of bins. IMPORTANT: cmap must be set to None.

legend\_title: Takes string. Title of discrete/unique data legend.

legend\_loc: Takes string. Location of base elements legend.

legend\_loc2: Takes string. Location of discrete/unique data legend.

save\_name: Takes string. If not defined, default save name will be the name of the network. IF save\_name is defined, it will prepend a suffix onto the network name to act as the name of the .png file. For example save\_name=’Pressure’ on the network ‘CTown.inp’ will result in a file called ‘PressureCTown.png’

cmap: Takes string. Colormap to be used for node colors.

color\_list: Takes list. List of each bin’s node color. Length of list should be equal to number of bins. IMPORTANT: cmap must be set to None.

plot\_discrete\_links(model, ax,bin\_edge\_num=5, parameter=None, value=None, bins='automatic', bin\_width\_list=None, bin\_label\_list=None,color\_list=None,tanks=True, reservoirs=True, pumps=True, valves=True,cmap='gist\_heat',legend=True, legend\_title=None, legend\_loc\_1='upper right',

legend\_loc\_2='lower right',savefig=True,save\_name=None)

Description

-----------

Plotting function for discrete link data.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

bin\_edge\_num: Takes integer. Number of bin edges to be created.

parameter: Takes string. Name of parameter, refer to list in get\_parameter() description.

value: Takes string or integer. This parameter is relevant when working with parameters that use timesteps. Takes ‘max’ ‘min’ ‘mean’ or an integer that represents the timestep index.

bins: Takes string or 1D array. If set to ‘automatic’ bins will be created in accordance to the parameter bin\_edge\_num. If array is provided, it’s length should be equal to the parameter bin\_edge\_num.

bin\_label\_list: Takes list. Label that each bin will use if legend is drawn. By default is set to be equal to the bin range. Length of list should be equal to number of bins.

savefig: Takes boolean. Determines whether the figure is saved or not.

bin\_width\_list: Takes list. List of each bin’s link width. Length of list should be equal to number of bins.

bin\_label\_list: Takes list. Label that each bin will use if legend is drawn. By default is set to be equal to the bin range. Length of list should be equal to number of bins.

color\_list: Takes list. List of each bin’s node color. Length of list should be equal to number of bins. IMPORTANT: cmap must be set to None.

tanks: Takes boolean. Determines whether to draw tanks or not.

reservoirs: Takes boolean. Determines whether to draw reservoirs or not.

pumps: Takes boolean. Determines whether to draw pumps or not.

valves: Takes boolean. Determines whether to draw valves or not.

cmap: Takes string. Colormap to be used for link colors.

legend: Takes boolean. Determines whether to draw legend or not.

color\_list: Takes list. List of each bin’s node color. Length of list should be equal to number of bins. IMPORTANT: cmap must be set to None.

legend\_title: Takes string. Title of discrete/unique data legend.

legend\_loc: Takes string. Location of base elements legend.

legend\_loc2: Takes string. Location of discrete/unique data legend.

savefig: Takes boolean. Determines whether the figure is saved or not.

save\_name: Takes string. If not defined, default save name will be the name of the network. IF save\_name is defined, it will prepend a suffix onto the network name to act as the name of the .png file. For example save\_name=’Pressure’ on the network ‘CTown.inp’ will result in a file called ‘PressureCTown.png’

plot\_continuous\_nodes(model,ax,parameter=None, value=None, tanks=True, reservoirs=True, pumps=True, valves=True,cmap='gist\_heat', color\_bar\_title=None,node\_size=100, node\_shape='.',savefig=True, save\_name=None)

Description

-----------

Plotting function for continuous node data.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

parameter: Takes string. Name of parameter, refer to list in get\_parameter() description.

value: Takes string or integer. This parameter is relevant when working with parameters that use timesteps. Takes ‘max’ ‘min’ ‘mean’ or an integer that represents the timestep index.

tanks: Takes boolean. Determines whether to draw tanks or not.

reservoirs: Takes boolean. Determines whether to draw reservoirs or not.

pumps: Takes boolean. Determines whether to draw pumps or not.

valves: Takes boolean. Determines whether to draw valves or not.

cmap: Takes string. Colormap to be used for node colors.

color\_bar\_title: Takes string. Title of colormap.

node\_size: Takes integer. Size of nodes on the plot.

node\_shape: Takes string. Shape of the nodes on the plot.

savefig: Takes boolean. Determines whether the figure is saved or not.

save\_name: Takes string. If not defined, default save name will be the name of the network. IF save\_name is defined, it will prepend a suffix onto the network name to act as the name of the .png file. For example save\_name=’Pressure’ on the network ‘CTown.inp’ will result in a file called ‘PressureCTown.png’

plot\_continuous\_links(model,ax,parameter=None,value=None,min\_width=1,

max\_width=5,tanks=True, reservoirs=True, pumps=True, valves=True,cmap='gist\_heat',color\_bar\_title=None,savefig=True, save\_name=None)

Description

-----------

Plotting function for continuous link data.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

parameter: Takes string. Name of parameter, refer to list in get\_parameter() description.

value: Takes string or integer. This parameter is relevant when working with parameters that use timesteps. Takes ‘max’ ‘min’ ‘mean’ or an integer that represents the timestep index.

min\_width: Takes integer. Minimum link width.

max\_width: Takes integer. Maximum link width.

tanks: Takes boolean. Determines whether to draw tanks or not.

reservoirs: Takes boolean. Determines whether to draw reservoirs or not.

pumps: Takes boolean. Determines whether to draw pumps or not.

valves: Takes boolean. Determines whether to draw valves or not.

cmap: Takes string. Colormap to be used for link colors.

color\_bar\_title: Takes string. Title of colormap.

savefig: Takes boolean. Determines whether the figure is saved or not.

save\_name: Takes string. If not defined, default save name will be the name of the network. IF save\_name is defined, it will prepend a suffix onto the network name to act as the name of the .png file. For example save\_name=’Pressure’ on the network ‘CTown.inp’ will result in a file called ‘PressureCTown.png’

animate\_plot(model,ax,function,fps=3,first\_timestep=0,last\_timestep=None,gif\_save\_name='gif',\*\*kwargs)

Description

-----------

Creates a .gif file of functions at all specified timesteps.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

function: Takes wdn-viz general function. The function that the user wants to be animated.

fps: Takes integer. Frames per second that the animation runs at.

first\_timestep: Takes integer. Timestep that the animation will start at.

last\_timestep: Takes integer. Timestep that the animation will end at.

gif\_save\_name: Takes string. Save name of the .gif file.

\*\*kwargs: Takes parameters. These are all the keyword parameters that the function the user wishes to plot would normally take.

plot\_unique\_data(model, ax, parameter=None, parameter\_type=None,data\_type=None,excel\_columns=None,customDataValues=None, bins='automatic',bin\_size\_list = None, bin\_shape\_list = None, bin\_edge\_num=None, bin\_width\_list=None, bin\_label\_list=None,bin\_border\_list = None, bin\_border\_width\_list = None,color\_list=None,min\_width=1,max\_width=5,tanks=True, reservoirs=True, pumps=True, valves=True,cmap='gist\_heat',legend=True, legend\_title=None,node\_size=100, node\_shape='.',legend\_loc\_1='upper right', legend\_loc\_2='lower right',savefig=True,save\_name=None,color\_bar\_title=None)

Description

-----------

Plotting function that handles excel data, unique parameters such as diameter, and custom data created inside of python.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

parameter\_type: Takes string. Type of element the parameter is. Can either be ‘node’ or ‘link’

data\_type: Takes string. Type of data that the excel data is. Can be ‘unique’ ‘continuous’ or ‘discrete’. Unique data is data which is grouped into non-numerical, distinct categories, such as pressure groups or demand patterns. Continuous and discrete data both involve numerical data, where the two differ in how they are represented. Continuous data is represented by a color bar, while discrete data is represented by separate bins that the data is put into.

excel\_columns: Takes array of length 2. An array of excel columns that should be used. The first value is the column # of the excel file that contains the list of elements, while the second is the column # of the excel file that contains the list of data values.

customDataValues: Takes array of length 2. An array that contains information from custom data created in python. The first value is the list of elements, while the second is the data values.

\*\*Inherits all parameters from discrete node/link general functions and continous node/link general function.

#Base Functions

"""

initialize\_model(inp\_file)

Description

-----------

Runs wntr/EPANET simulations and stores results as well as network parameters in a dictionary for further use by all other functions.

Parameters

----------

inp\_file: Takes string. Location in directory where input file (ending in .inp) is located. Can be ‘inputfile.inp’ if in root directory. If in subfolder, take care to use \\, for example 'Networks\\input file.inp’

convert\_excel(model,file,data\_type,element\_index,value\_index)

Description

-----------

Converts an excel file into the correct dictionary structure needed to be used with drawing functions.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

file: Takes string. Location in directory where excel file (ending in .xlsx) is located. Same rules on where the file is located as inp\_file

data\_type: Takes string. Type of data that the excel data is. Can be ‘unique’ ‘continuous’ or ‘discrete’. Unique data is data which is grouped into non-numerical, distinct categories, such as pressure groups or demand patterns. Continuous and discrete data both involve numerical data, where the two differ in how they are represented. Continuous data is represented by a color bar, while discrete data is represented by separate bins that the data is put into.

element\_index: Takes integer. This is the column index number that the names of nodes/links are located in the excel file.

value\_index: Takes integer. This is the column index number that the values associated with each node/link are located in the excel file.

save\_fig(model, save\_name=None)

Description

-----------

Saves a .png of the figure to the root folder. One can edit where the image is saved to by editing model[‘image\_path’]

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

save\_name: Takes string. If not defined, default save name will be the name of the network. IF save\_name is defined, it will prepend a suffix onto the network name to act as the name of the .png file. For example save\_name=’Pressure’ on the network ‘CTown.inp’ will result in a file called ‘PressureCTown.png’

get\_parameter(model,parameter\_type,parameter,value=None,tanks=False, reservoirs=False)

Description

-----------

Retrieves specific element parameter from either nodes or links. There are 4 distinct groups of parameters. Nodes/links either do not change with time or have unique values at each timestep.

Constant Node Parameters:

‘base\_demand’

‘elevation’

‘emitter\_coefficient’

‘initial\_quality’

Node Parameters w/ Timestep:

‘head’

‘demand’

‘leak\_demand’

‘leak\_area’

‘leak\_discharge\_coeff’

Constant Link Parameters:

‘length’

‘minor\_loss’

‘bulk\_coeff’

‘wall\_coeff’

Link Parameters w/ Timestep:

‘flowrate’

‘velocity’

‘headloss’

‘friction\_factor’

‘reaction\_rate’

‘quality’

There is also ‘diameter’ and ‘roughness’ however these are covered in another function.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

parameter\_type: Takes string. Type of element the parameter is. Can either be ‘node’ or ‘link’

parameter: Takes string. Name of parameter, refer to list in get\_parameter() description.

value: Takes string or integer. This parameter is relevant when working with parameters that use timesteps. Takes ‘max’ ‘min’ ‘mean’ or an integer that represents the timestep index.

tanks: Takes boolean. Determines if parameter is collected for tanks.

reservoirs: Takes boolean. Determines if parameter is collected for reservoirs.

get\_demand\_patterns(model)

Description

-----------

Specialized function that plots demand pattern groups.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

bin\_parameter(model,parameter\_results,element\_list,bin\_edge\_num,

bin\_list='automatic')

Description

-----------

Takes results from get\_parameter() and puts them into bin groups for discrete plotting. One quirk with this function is that if provided custom bin edges with the bin\_list parameter, it will add new bins < [value] and/or > [value] if some parameter values do not fall in the custom range given by the user. Additionally, if no parameter values fall inbetween a bin, the package will automatically delete it. This is important when using customization options in general functions, as the user should provide values for each bin in the case of doing bin customization.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

parameter\_results: Takes pandas Series. Contains parameter results for each element from get\_parameter()

element\_list: Takes list. Contains the names of each element that has a parameter value associated with it.

bin\_edge\_num: Takes integer. Number of bin edges to be created.

bin\_list: Takes string or 1D array. If set to ‘automatic’ bins will be created in accordance to the parameter bin\_edge\_num. If array is provided, it’s length should be equal to the parameter bin\_edge\_num.

draw\_nodes(model,node\_list,parameter\_results=[],node\_size=300,

node\_color='k',cmap='tab10',node\_shape='.',edge\_colors='k',

line\_widths=0,label=None)

Description

-----------

Drawing function used for continuous plots or custom node drawing.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

node\_list: Takes 1D array. List of nodes that are to be drawn.

parameter\_results: Takes pandas Series. Contains parameter results for each element from get\_parameter()

node\_size: Takes integer. Size of nodes on the plot.

node\_color: Takes string. Color of the nodes on the plot.

cmap: Takes string. Colormap to be used for node colors. Only relevant when parameter\_results is set.

node\_shape: Takes string. Shape of the nodes on the plot.

edge\_colors: Takes string. Color of node borders on the plot.

line\_widths: Takes integer. Width of the node borders on the plot.

label: Takes string. Label for nodes, used when drawing legend.

draw\_links(model,link\_list,parameter\_results=[],edge\_color='k',

cmap='tab10',widths=[])

Description

-----------

Drawing function used for continuous plots or custom link drawing.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

link\_list: Takes 1D array. List of links that are to be drawn.

parameter\_results: Takes pandas Series. Contains parameter results for each element from get\_parameter()

edge\_color: Takes string. Color of the strings on the plot.

cmap: Takes string. Colormap to be used for node colors. Only relevant when parameter\_results is set.

widths: Takes 1D array. Array of the width of each link on the plot.

draw\_base\_elements(model,ax,nodes=True,links=True,reservoirs=True,tanks=True,pumps=True,valves=True,legend=True)

Description

-----------

Drawing functions for all general plotting functions. Draws nodes, links, tanks, reservoirs, and valves without any data attached to them. Can be customized with parameters to not include certain elements.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

nodes: Takes boolean. Determines whether to draw nodes or not.

links: Takes boolean. Determines whether to draw links or not.

reservoirs: Takes boolean. Determines whether to draw reservoirs or not.

tanks: Takes boolean. Determines whether to draw tanks or not.

pumps: Takes boolean. Determines whether to draw pumps or not.

valves: Takes boolean. Determines whether to draw valves or not.

legend: Takes boolean. Determines whether to draw legend or not.

draw\_discrete\_nodes(model,ax,nodes,bin\_list,bin\_size\_list=None, bin\_label\_list=None,bin\_shape\_list=None,cmap='tab10',bin\_border\_list = None, bin\_border\_width\_list = None,color\_list = None)

Description

-----------

Drawing function for discrete and unique node plotting.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

nodes: Takes dictionary. Dictionary of node-value pairs that are organized by bins. Created from bin\_parameter()

bin\_list: Takes list. List of bin names.

bin\_size\_list: Takes list. List of each bin’s node size. Length of list should be equal to number of bins.

bin\_label\_list: Takes list. Label that each bin will use if legend is drawn. By default is set to be equal to the bin range. Length of list should be equal to number of bins.

bin\_shape\_list: Takes list. List of each bin’s marker shape. Length of list should be equal to number of bins.

cmap: Takes string. Colormap to be used for node colors.

bin\_border\_list: Takes list. List of each bin’s node border color. Length of list should be equal to number of bins.

bin\_border\_width\_list: Takes list. List of each bin’s node border width. Length of list should be equal to number of bins.

color\_list: Takes list. List of each bin’s node color. Length of list should be equal to number of bins. IMPORTANT: cmap must be set to None.

draw\_discrete\_links(model,ax,links, bin\_list, bin\_width\_list=None, bin\_label\_list=None,cmap='tab10', color\_list = None):

Description

-----------

Drawing function for discrete and unique link plotting.

Parameters

----------

model: Takes dictionary. Contains necessary information about network.

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

links: Takes dictionary. Dictionary of link-value pairs that are organized by bins. Created from bin\_parameter()

bin\_list: Takes list. List of bin names.

bin\_width\_list: Takes list. List of each bin’s link width. Length of list should be equal to number of bins.

bin\_label\_list: Takes list. Label that each bin will use if legend is drawn. By default is set to be equal to the bin range. Length of list should be equal to number of bins.

cmap: Takes string. Colormap to be used for link colors.

color\_list: Takes list. List of each bin’s node color. Length of list should be equal to number of bins. IMPORTANT: cmap must be set to None.

draw\_legend(ax,bin\_list=[],title=None,pumps=True,

loc='upper right',loc2='lower right')

Description

-----------

Drawing function for legend for both base elements and discrete/unique data.

Parameters

----------

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

bin\_list: Takes list. List of bin names.

title: Takes string. Title of discrete/unique data legend.

pumps: Takes boolean. If pumps are not being drawn, this should be set to False. This ensures that the pump matplotlib Patch isn’t added to the base elements legend.

loc: Takes string. Location of base elements legend.

loc2: Takes string. Location of discrete/unique data legend.

draw\_color\_bar(ax,g,cmap,color\_bar\_title=None):

Description

-----------

Drawing function for color bars for continuous plots.

Parameters

----------

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

g: Takes networkx object. This is the object where the elements plotted by network x are stored from nxp.draw\_networkx\_edges or nxp.draw\_networkx\_nodes.

cmap: Takes string. Colormap to be used for element colors.

color\_bar\_title: Takes string. Title of colormap.

draw\_label(model,ax,labels,x\_coords,y\_coords,nodes=None)

Description

-----------

Drawing function for labels on the plot.

Parameters

----------

ax: Takes matplotlib ax. Matplotlib axes that the elements are being drawn on.

labels: Takes string list. List of labels that the user wants to be drawn.

x\_coords: Takes list. List of x coordinates for labels. The x coordinate location in the list corresponds to the same index in the labels parameter.

y\_coords: Takes list. List of y coordinates for labels. The y coordinate location in the list corresponds to the same index in the labels parameter.

nodes: Takes list. List of nodes that the coordinates are relative to. IMPORTANT: If nodes is not set to something other than none, the coordinates should range from 0-1 as they represent absolute coordinates, where (0,0) is the bottom left corner of the figure, while (1,1) is the top right corner of the figure. However, if nodes is set to something other than None, then the coordinates the user provides are relative to the nodes. The coordinate system greatly differs between networks so it may take some trial and error on the user to position the label correctly when using a list of nodes.

"""