Instructions to run code

The imported libraries used are NumPy, and Matplotlib.
*Matplotlib's interactive mode is enabled thus making MacOSX unusable. Operating systems other than Windows, and MacOSX have not been tested.

Parameter	***Meaning***	***Type***	***Symbol***	***Units***
T	Simulation runtime	float		s
simu_mult	Simulation speed factor	int		
dx	Step size between points	float	dx	m
dt	Simulation time step	float	dt	s
L	Medium length	float	l	m
c_func	Wave velocity as a function of x	function	$c(x)$	m/s
curr_dx	Current point to save data for	int		"m"
PULSE	If true, a pulse will be sent. Otherwise, a standing wave will be generated.	boolean		
pulse_f	Frequency of pulse	float		Hz
pulse_t	Time length of pulse	float		s
halves	Number of standing wave sine halves.	int		
resm	Resolution multiplier for the fourier transform frequency.	int		

Lines 56-58 initialize the dynamic plot figures. The parameters for them could change there.

Line 151 (end of run() function) could be modified to obtain different measurements using get_instafreq(), get_spatfreq(), get_tempfreq(), and

adding to the different lists. Line 160 could be modified to plot a different graph.

- *Function Usage*

*m could be spliced from M or obtained by a different means.

get_instafreq(m, dom="k"):

Hilbert transform to find k(x) (dom="k") for m, a discrete y(x) *or* f(t) (dom="f") for m, a discrete y(t). Returns k(x) *or* f(t) and the amplitude A(x) as a tuple.

get_spatfreq(m):

FFT. Returns the fundamental (maximum) angular frequency and amplitude as a tuple.

get_tempfreq(k, average=True, s=0, e=-1):

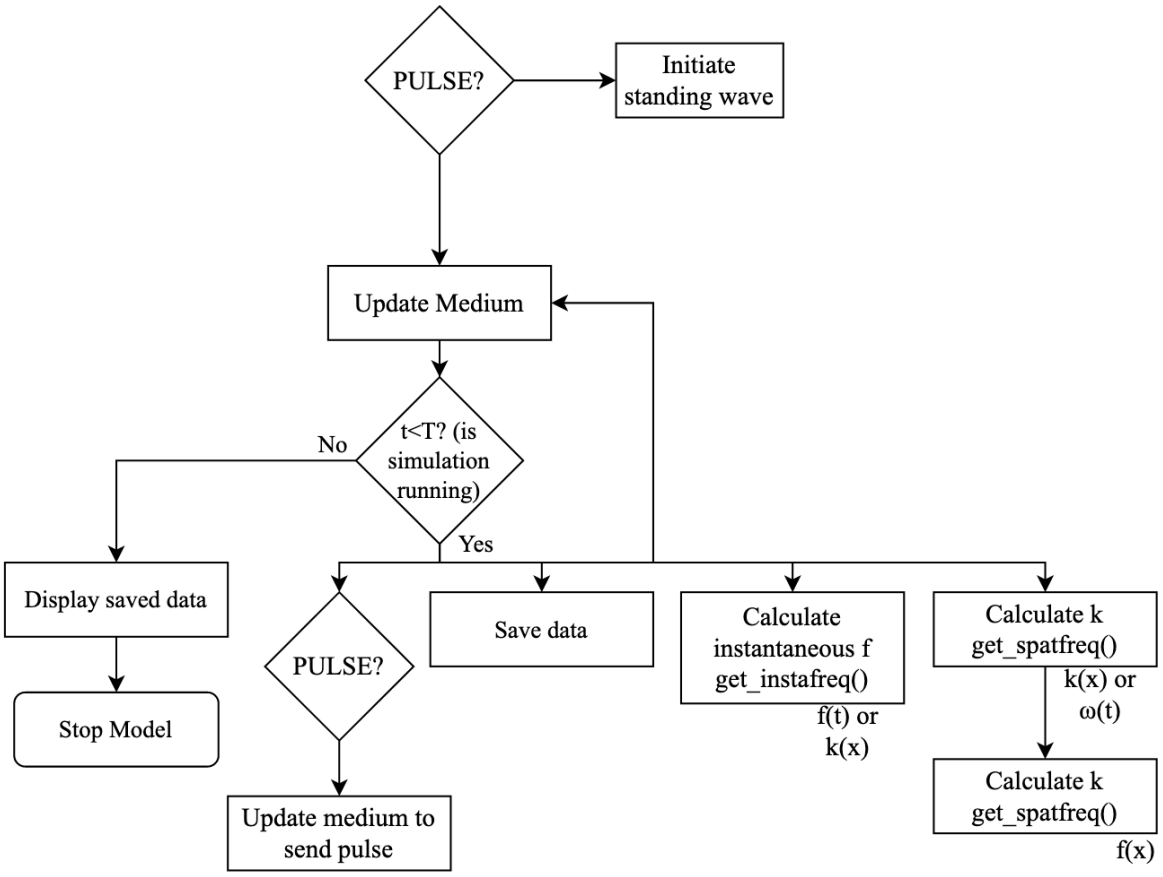
Takes in the angular spatial frequency and returns the matching temporal frequency using $f=kc/2\pi$. For averaging c(x) s,e start and end indexes of the medium section to average are required. Otherwise f(x) is computed (notice dimensions of k).

standing_wave(m, l, h):

Sets a medium in an initial standing wave state. m is the medium. l is the medium length. h is the number of sine halves.

pulse(f, time):

Sets the position y(x=0) to a sine wave with frequency f for time seconds. When the time is over, y(x=0)=0



Updating the wave:

$$y_{i,j+1} = 2y_{i,j} - y_{i,j-1} + \frac{c^2}{c'^2}\left[y_{i+1,j} + y_{i-1,j} - 2y_{i,j}\right]$$