Report 31st August 2018

I sorted out the importing problems, and now the process is automated. Also found out that the number of nodes is not fixed, *even with the same parameters values (stochastic simulation)*, which means the number of columns in the files to be imported is variable (I automated this step as well, so it’s all good on the importing side).

The simulations here is with the default values specified in the notes by Luke.

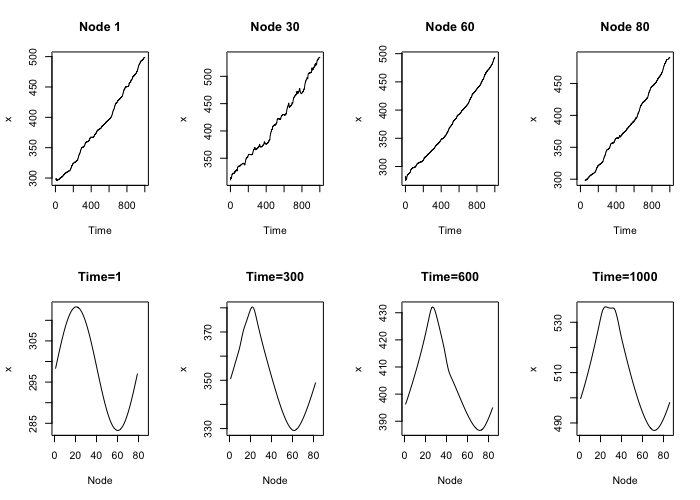
# Cell coordinates

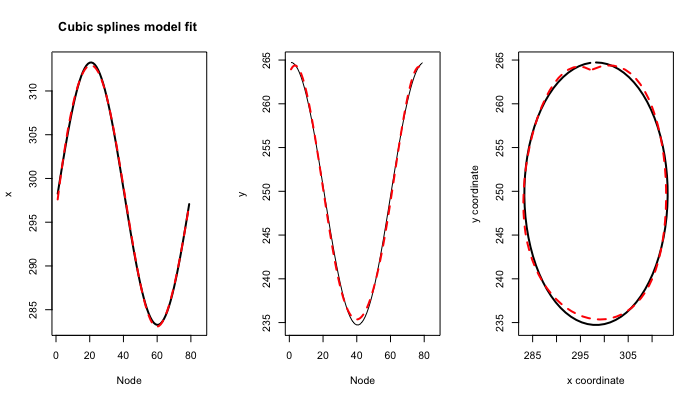
* x and y coordinates at each finite element node.
* 1000 x 86 matrices (1 for each coordinate). Note the dimension of the nodes, 86, is not fixed. I also got 83, 85 in other simulations.

## xy.pngX coordinate

Plot below (top row) shows x as a function of time for nodes 1, 30, 60 and 80. This representation does not make sense if the node positions are arbitrary. This is roughly a straight line.

The bottom row shows the evolution of x measured at all the nodes for time points 1, 300, 600 and 1000. Curve has a sinusoidal shape.



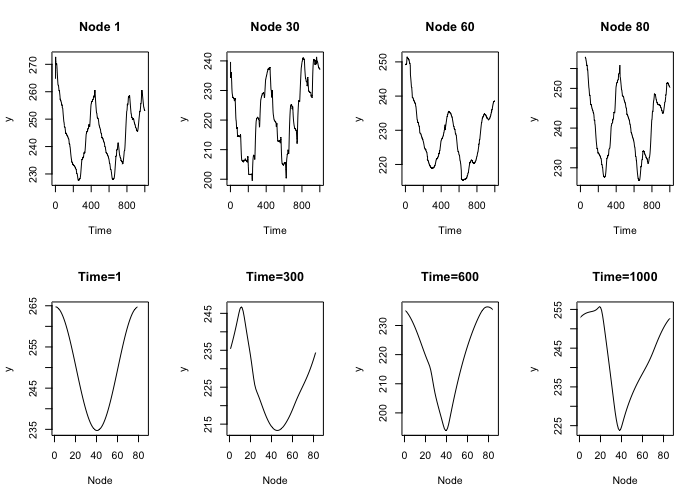


## Y coordinate

Same as with the x coordinate, the plot below (top row) shows y as a function of time for nodes 1, 30, 60 and 80.

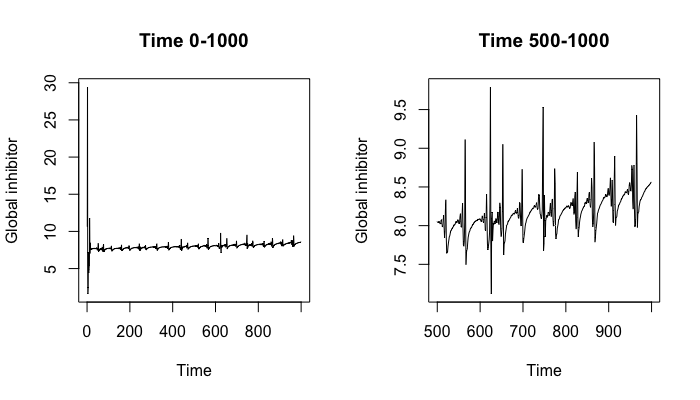
The bottom row shows the evolution of y measured at all the nodes for time points 1, 300, 600 and 1000.

In both scenarios, the pattern looks similar across time points/nodes, but with different degrees of smoothness.



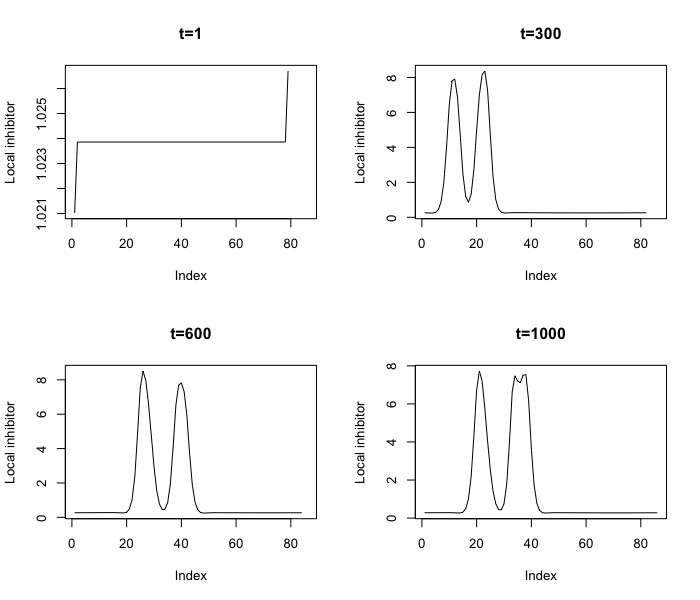
# Global inhibitor

* 1000 x 1 vector: 1 value at each time point



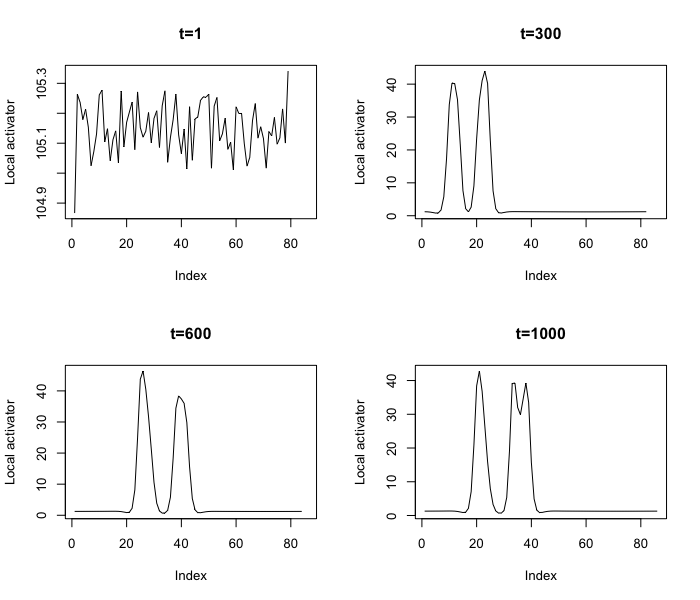
# Local inhibitor

* 1000 x 86 matrices



# Local activator

* 1000 x 86 matrices



# Local stimulus strength

* 1000 x 86 matrices

