1. Function written.
2. An basic strategy would involve monitoring the distance to the walls above and below the flyer, the difference between these two will be the input to the controller. When the flyer gets closer to the upper wall than the lower wall, the flyer should thrust downwards, and when it’s closer to the upper wall, it should thrust upwards (a P-controller). This strategy can be improved by reducing the commanded thrust if it would cause a significant sudden change to the input (a P-D controller). However, this strategy is prone to leaving a steady state error, which can be eliminated by adding a thrust component which is proportional to the integral of this error (a P-I-D controller).  
   A downside to this approach, however, is that there is still information given which is being left unused. Namely, there are a 5 sample points given for both the top and bottom walls. An ideal strategy would use the above P-I-D technique to balance the flyer between all these points. While 5 separate P-I-D loops could be set up, this sort of control task with a high dimensionality on the sensor inputs is perfect for a neural network. So, a neural network was devised (detailed in part 3) whose architecture is capable of emulating 5 concurrent P-I-D loops. One problem with this approach, however, is that since the gradient for the environment is unknowable, a simple back-propagation technique won’t work for training the network; so, a genetic algorithm can be employed instead.
3. A basic P-I-D algorithm would work as follows: