E-coli Model

Rupert Young

April 20, 2013

1 Introduction

This document describes the perceptual controls systems used to implement a real-world model, on a Lego Mindstorms robot, of the behaviour of the bacterial organism E-coli. The basic manner by which E-coli moves around is a form of chemotaxis, that is, motion up (or down) the gradient of a chemical attractant. While E-coli senses an increase in the presence of a chemical it swims in a forward direction, otherwise it rotates for a random period and then swims in a new direction. The result is that E-coli spends more time swimming in the right direction climbing up the gradient and ending up in a local maximum of the chemical attractant.

2 Model

Figure 1 shows the systems involved with this implmentation. See [1] for video of robot executing these systems.

2.1 Sensor

The model has one sensor. In this example it is a simple light sensor which gives a reading of 0 to 100 representing the light level falling on the sensor. However, it could be any type of sensor, such as sound.

2.2 Actuators

The robot has two motors, which turn wheels allowing the unit to move around. The motors can be controlled either by commands for it to move forwards (or backwards) at a particular speed, or to rotate the motor through a specified angle, also at a specifiec speed.

2.3 Control Systems

The control systems, described from the top of figure 1, are as follows:

Sensor Control The top level system controls the level of perceived light. While there is an error signal the output is 1, otherwise 0. That is, while the perceived light level is less than the reference then the lower level systems are activated

Sensor Change Control The input to this control system is 0 or 1 representing whether the perceived change in the sensor value is positive or negative. A reference value of 1 represents an active system, and a goal of forward motion. The outputs of this system are either 0 or 1, respectively representing lower level goals for either forward motion or rotation of the robot unit.

Forward Control This system activates the lower level motor controls according to whether its reference is 0 or 1.

Speed Controls The speed controllers control the speed to be applied to the motors, according to a reference speed of either 0 or 200. The MotorWrite functions below the speed controllers set the speed of the actual motors with the value output from the speed controllers.

Rotate Control This system controls whether the unit should rotate. A reference of 1 indicates a goal of rotation, derived from higher level systems which detected that the change in the sensor value was not increasing. If the output of this system is 1 then a transfer function is activated which generates a random number which becomes the references for the amount by which to rotate the unit.

Rotate Angle Rotation Controls These two systems receive a reference goal of a randomly generated value indicating the angle through which to rotate the motors. One system receives a positive reference, the other a negative reference which results in the motors rotating in opposite directions, and so the unit itself will rotate, in a clockwise direction. There are two transfer functions which connect the error signal to the input of the higher Rotate Control system. These indicate to the higher system when the motor has finished rotating, by outputting a value of 1 when the error reduces to 0. This will result in an output for the Rotate Control system of 0 switching off the rotation system. However, this would result in the rotation system only activating once, therefore one of the transfer functions is set to habituate, over a few iterations of the program, from the value of 1 down to 0, reactivating the rotation system if the goal for rotation is still present.

Motor Conflict Control The two areas in figure 1 framed by a dashed line represent controls system either for moving forward or for unit rotation. Both these systems send commands to the same pair of motors. As there can be overlap between the activation of these systems, while the forward motion system is winding down due to the integration function, there is potential for conflicting signals being received by the motors. Therefore, the Motor Conflict is included to resolve such conflict and sits one layer above the two movement systems. It acheives this by taking as input the speed of the speed controls and only outputs a signal when the speed has reduced to zero.

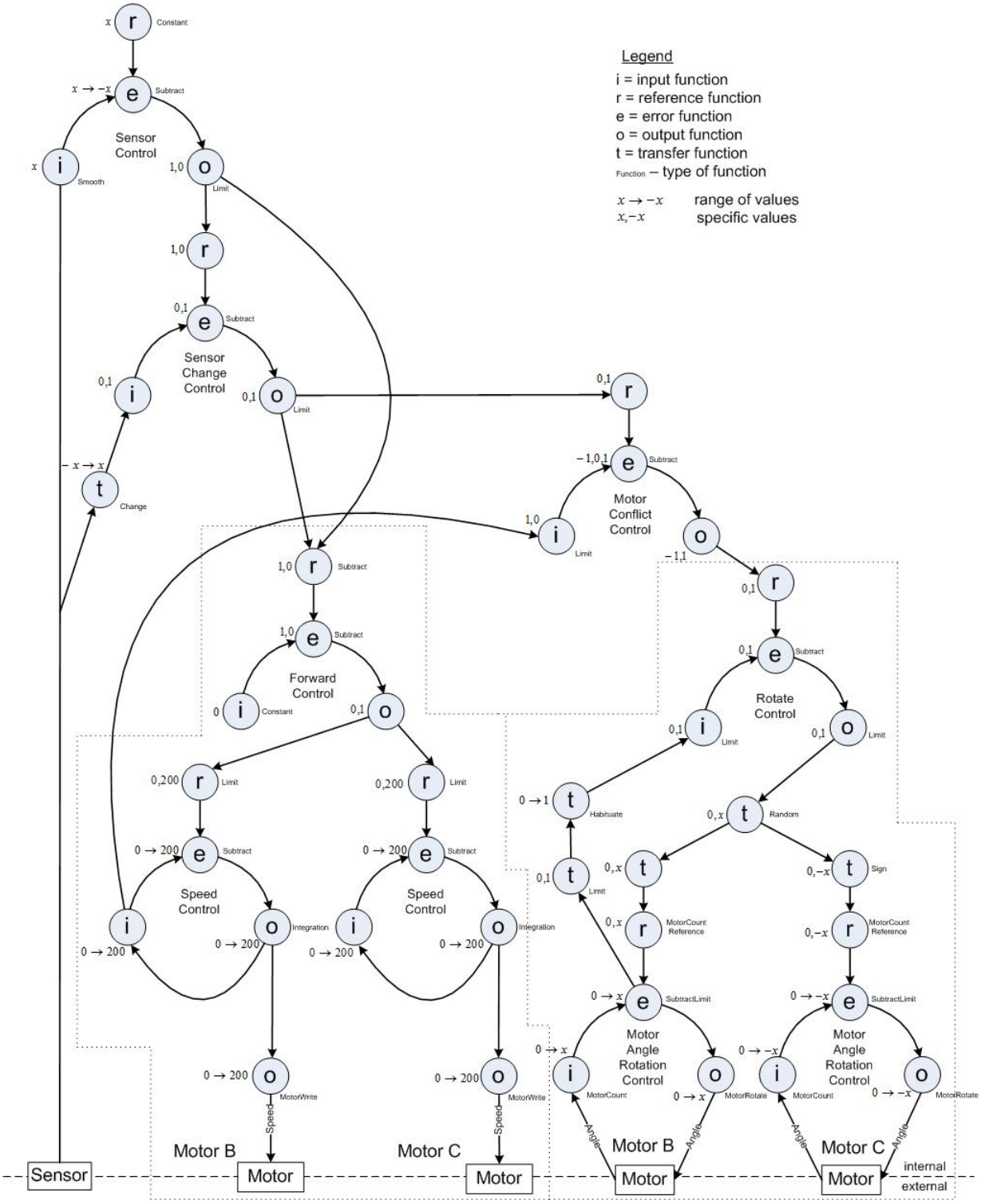

Figure 1: E-coli model control systems with separate systems for forward motion and rotation of the robot unit

References

[1] Rupert Young. Bill the bacterium. http://www.youtube.com/watch?v=ZAeHU6GGPdE, April 2013.