

### **Week Starting 18th July.**

My main aim for this week has been to become proficient in the use of MATLAB, and continue my reading into various aspects of the Artificial Immune System, specifically the General Suppression Control Framework, and gain greater understanding of the significance of current research in the different areas.

By Monday afternoon, I'd finished reading and trying examples in the "Getting Started" tutorial that comes with MATLAB, covering the core topics, including matrices, arrays, programming, graphics, and creating GUI's. I then took a look at the Modular Robot MATLAB code in greater detail, and found it to be relative short, and easy to read. As a practice in the use of MATLAB, I modified the code to display the separate figures in one window, and experimented with figure drawing properties, and the figure editor.

I started on Tuesday by reading "An Immuno Control Framework for Decentralized Mechatronic Control". This is the paper that accompanies the MATLAB code for the Mechatronic arm, and defines the General Suppression Framework used to model the suppression hypothesis of the immune discrimination theory. The use of such a framework is to generate "emergent group behaviour" from a fully distributed and hence robust system. Where many introductory papers on AIS's tend to skip over the suppression hypothesis, this paper provides greater detail and demonstrates its power by actually implementing a system inspired from it. After reading the paper, I began to think that although powerful, the framework would not have a direct use in image processing, such as object classification, as this is not really what the framework is about. The framework relies on a "local environment" that cells can affect change within, and the power of the framework is the stabilising nature of the suppression and stimulation feedback from neighbouring cells and the environment. I have been trying to think if cells distributed over the surface of an image could stimulate each other in such a way as to identify shapes or patterns, where the feedback of the system is in the cells location on the image. Perhaps cells would gather and disperse based on very local, computable information, and converge into a formation that identified features of an image. This requires more thought.

In the afternoon, I went on to read "Immunologic Response Manipulation of AIS Agents". This paper formalises the algorithms used in "Immunologic Control Framework for Automated Material Handling", including manipulation of compatibility lists, but is not so useful at the moment.

I also read "General Suppression Control Framework Application in Self-balancing robots", which provides greater depth on details and power of the GSCF. This time, the application uses several "suppressor cells", compared to the single suppressor cell in the Mechatronic arm, and demonstrates how suppressor cells can stimulate each other. Much of the paper though is repeated from its predecessor, though the system implemented is very interesting.

On Tuesday evening, I wrote a very small MATLAB program to test the negative selection algorithm as defined in "AIS: A novel paradigm to pattern recognition". I also borrowed their character recognition example. I took a bitmap containing the letters A-Z, and used MATLAB to disassemble the monochrome image into an array of matrices representing each character, which I would expose my randomly initialised population to. My aim was to determine if a given sample was A-Z (self) or other (non-self). The results were not very successful, but this I think was due to a rather inadequate "self" pattern set. Not only were they very limited in number, but contained entirely unique symbols, so it would be hard for the algorithm to display any kind of learning at all. I implemented several affinity measures based on the binary encoding of the characters, including hamming distance, r-continuous bits, and a custom measure that favours greater quantities of greater length bit sequences. The detection of self and non-self was very sensitive to the algorithm parameters. The font I was using also only used 4x6 fonts, which didn't provide very much hamming distance between letters. The r-continuous affinity measure seems to be quite popular, though for 2D data such as images, it seems horizontally biased. I wondered if a combination of horizontal and vertical r-continuous calculations would be of benefit, and if analysis of pixel distributions in one dimension about both axis would allow patterns with displaced symbols to be compared.

I spent Wednesday learning some more about MATLAB, including how to use a toolbox. I wrote a small program to capture a frame from an image source using the image acquisition toolbox, and then experimented with simulink, which is incredibly similar in operation to the Direct X SDK's filtergraph

application GUI (and COM in windows in general), with the notion of pins and processing blocks. I joined a few image processing blocks, and read a little about its integration with MATLAB.

I went on to read “The Clonal Selection Algorithm with Engineering Applications”, and “Improved Pattern recognition with Artificial Clonal Selection” in preparation for the next day when I planned to program a clonal selection algorithm. Between the two papers, I was introduced to CLONCLAS, which I initially thought was the main algorithm used for pattern classification, but it doesn't really provide any classification ability. It needs adapting.

On Thursday I used the CLONALG algorithm to demonstrate the learning capability of clonal selection. As discussed in the previous two papers I'd read, the algorithm, initialised from a random population, converged to equal the training pattern. Though there are many parameters that can be altered, they do not seem very sensitive, and the random population converged to be identical to the training item within ~60 generations, given an initial population of 50 receptors. I started to design a classification algorithm based on CLONEALG, though by the end of Thursday I'd found a paper describing one, called the Clonal Selection Classification Algorithm (CSCA). This was quite a lengthy document which I finished reading on Friday. The document mentioned a machine learning and visualisation toolkit written in Java, called WEKA, so I downloaded this, and played with it a bit to see the strengths of some other algorithms in classification and clustering problems.

I spent the rest of the day modifying my clonal selection algorithm for classification, and creating a higher resolution (10x12) collection of the handwritten letters A-Z for testing the algorithm, each letter repeated 5 times. I've not had time to test the algorithm as it has gotten quite late, but the few test runs look very promising, with matured cells displaying quite a high general affinity for patterns in their class (i.e. their letter), hopefully demonstrating good generalisation.

Over the week, I've become very confident about using the various features of MATLAB. It allows prototypes to be created very quickly with access to the various toolkits and matrix/vector operations. I especially like how quickly and easily figures allow data to be visualised.

Next week I think it'll be best to start looking at potential problems to apply Artificial Immune System theory to. Clonal algorithms will be useful in the area of pattern recognition where generalisation of cells will hopefully allow classification of patterns that have been trained from huge datasets, providing a large performance increase over non-machine learning techniques.

I've thought about applying AIS's to interpreting sign language, focusing on pattern recognition of the gestures. Though I understand this would be very complicated, a reduced sign language involving only hand motion could first be established, and the use of coloured gloves would simplify collection of gesture data. The biggest hurdle in recognising full sign language would I think be the fact that a sign could be of any length, with mannerisms that could be repeated differing numbers of times. There are of course many other applications of pattern recognition in image processing that would be interesting to explore, such as texture classification and object or face recognition.

Also, although I've questioned the usefulness of the general suppression framework in image processing, it feels like it could be quite cleverly applied somehow, and it would be challenging to try.

Over the weekend I'll finish experimenting with my clonal selection algorithm, and try and think of interesting problem areas that may be solved by immune inspired algorithms.

This report has become quite long, so if you would rather I omitted certain details in following weeks, please let me know.

Thanks again for your time.  
Steve