

Application of the Feature-Detection Rule to the Negative Selection Algorithm

Mario Poggiolini and Andries Engelbrecht

Department of Computer Science, University of Pretoria, Pretoria, South Africa

Abstract

The Negative Selection Algorithm developed by Forrest et al. was inspired by the way in which T-cell lymphocytes mature within the thymus before being released into the blood system. The mature T-cell lymphocytes exhibit an interesting characteristic, in that they are only activated by non-self cells that invade the human body. The Negative Selection Algorithm utilises an affinity matching function to ascertain whether the affinity between a newly generated (NSA) T-cell lymphocyte and a self-cell is less than a particular threshold; that is, whether the T-cell lymphocyte is activated by the self-cell. T-cell lymphocytes not activated by self-cells become mature T-cell lymphocytes. A new affinity matching function termed the feature-detection rule is introduced in this paper. The feature-detection rule utilises the interrelationship between both adjacent and non-adjacent features of a particular problem domain to determine whether an antigen is activated by an artificial lymphocyte. The performance of the feature-detection rule is contrasted with traditional affinity matching functions, currently employed within Negative Selection Algorithms, most notably the r-chunks rule (which subsumes the r-contiguous bits rule) and the hamming distance rule. This paper shows that the feature-detection rule greatly improves the detection rates and false alarm rates exhibited by the NSA (utilising the r-chunks and hamming distance rule) in addition to refuting the way in which permutation masks are currently being applied in artificial immune systems.

Keywords: Artificial immune systems, Affinity matching functions, Negative selection, Feature selection

1. Introduction

Artificial immune systems (AISs) emerged in 1986 as a new Computational Intelligence paradigm [10]. An AIS can be defined as a system of interconnected components which emulate one or more characteristics resident within the humoral/natural immune system (NIS) to accomplish a particular task. The NIS has evolved over millions of years and comprises various facets which act in tandem to protect the body. The NIS has a genetic (germ-line) and an adaptive (somatic) component. The innate immune system is genetically based: that is, it does not require a previous encounter with an antigen to be able to recognise it. The adaptive immune system, however, is capable of fine-tuning its response to an encountered antigen and does develop a memory [4, 18, 21].

The adaptive immune system has been the focal point of AIS research since its inception [7, 8]. Adaptive immunity encompasses three major components, namely learning, memory, and adaptability [21]. A key constituent of the adaptive immune system is the lymphocyte, which can be further segmented into B-cell lymphocytes and T-cell

lymphocytes. Although both B-cell and T-cell lymphocytes originate in the bone marrow, their roles within the NIS are fundamentally different [4, 18]. Firstly, T-cell lymphocytes undergo a vigorous maturation process within the thymus before being released into the blood system, whereas B-cell lymphocytes are simply released into the blood system after formation. Secondly, the antibodies released by B-cell lymphocytes can bind freely with antigens, whereas T-cell lymphocytes can bind only to antigens embedded within major histocompatibility complex (MHC) molecules offered by antigen presenting cells like macrophages [21]. T-cells are primarily responsible for cell-mediated immunity whereas B-cells are primarily responsible for humoral immunity (immunity of the human immune system). Immature T-cells migrate to the thymus to learn the concept of “self” by undergoing two selection processes namely: positive and negative selection [5]. The negative selection process destroys all T-cells with a strong avidity to a self-MHC peptide complex. The positive selection process destroys all T-cells with a relatively weak avidity to a self-MHC peptide complex. For a complete overview of the B-cell maturation process, refer to [4, 18, 21].

The Negative Selection Algorithm (NSA) developed by Forrest et al. [11] is inspired by the T-cell maturation process and has been widely adopted by the AIS community. One of the compelling reasons underpinning its adoption is that it is conceptually simple and allows a variety of different affinity matching functions to be employed (affinity matching functions are the means through which the NSA determines the affinity between a detector and a non-self/self-cell). The most popular affinity-matching functions currently employed by AIS researchers (within the context of the NSA) have limited foresight in that they merely consider relationships between adjacent attributes of a self/non-self vector and a detector vector to determine whether the detector is activated by a particular antigen.

The work presented in this paper:

- Discusses a new affinity-matching function, which utilises the interrelationships between both adjacent and non-adjacent attributes of a self/non-self vector and a detector to determine whether the detector is activated by the self/non-self vector.
- Contrasts the new affinity-matching function to traditional affinity-matching functions, employed by the NSA, and demonstrates that the new affinity-matching function exhibits superior detection rates whilst minimising false-alarm rates (detection rates and false alarm rates are typical measures used to quantify the performance of the NSA).
- Discusses how relevant attributes/features of a self/non-self vector (used by the feature-detection rule) can easily be determined by making a simple modification to the NSA algorithm.
- Examines the schism between views on permutation masks (permutation masks are an implementation of MHC molecules) currently existing within AIS literature (some researchers state that permutation masks are a vital mechanism to reduce holes induced by detectors, whereas other researchers disagree).

The remainder of this paper is organised as follows:

- Section 2 discusses the shape-space theory, affinity matching functions traditionally employed by AISs (most notably the NSA class of algorithms), undetectable strings/holes induced by affinity matching functions followed by approaches undertaken to minimise the impact of holes.
- Section 3 discusses metrics that are used to quantify the performance of AISs (particularly the NSA class of algorithms).
- Section 4 discusses Forrest et al.’s NSA, the effect of affinity matching functions on the performance of the NSA and the discriminative power of detectors under the NSA.
- Section 5 introduces the feature-detection rule, discusses the discriminative power and matching probability of detectors under the feature-detection rule and highlights how the NSA can be modified such that the discovery of relevant features is included in the learning process.
- Section 6 discusses the experiments conducted to investigate the performance of the feature-detection rule vs. the traditional affinity matching functions discussed in section 2.
- Section 7 summarises the concepts pertaining to the feature-detection rule examined in this paper in addition to stipulating future studies that can be undertaken to expand upon the work presented in this paper.

2. SHAPE SPACE THEORY AND AFFINITY MATCHING FUNCTIONS

Regardless of the nature of a particular AIS algorithm, most AIS algorithms employ the shape-space concept proposed by Perelson and Oster (1979), which allows a quantitative description of the interactions of receptor molecules residing on lymphocytes and antigens. The shape-space theory states that a population of N individuals (detectors) can be represented as a finite volume, V , containing n points (where n is the dimensionality of a detector). Each detector has a volume, V_r , surrounding it such that any complementary antigen that lies within V_r is recognised by the individual. The term V_r is called the detection area, and its size depends on a parameter, r , known as the affinity threshold. In this shape space, a detector molecule is depicted as a vector, \mathbf{x} , with coordinates (x_1, x_2, \dots, x_n) , whereas an antigen/non-self artefact is depicted as a vector, \mathbf{y} , with coordinates (y_1, y_2, \dots, y_n) . Building on the shape-space theory, AIS researchers have defined a variety of approaches in which the affinity between an antigen and a detector can be quantified, via a variety of affinity matching functions discussed next.

2.1. AFFINITY MATCHING FUNCTIONS

The affinity-matching functions presented in this section attempt to mimic the bonding process that occurs in the NIS. When an antigen and receptor bind covalently, they do so with varying degrees of strength depending on how well the detector's V region can recognise the antigen. Given a detector, \mathbf{x} , and an antigen, \mathbf{y} , a number of affinity matching functions can be defined to determine whether \mathbf{x} and \mathbf{y} match. Three popular affinity matching functions frequently employed with the NSA, namely the hamming distance rule, r -contiguous bits rule, and r -chunks rule are described below:

1. **Hamming Distance Rule:** The hamming distance (HD) between two binary vectors is the number of corresponding bits that differ. For example, if $\mathbf{x} = (1, 0, 0, 1)$ and $\mathbf{y} = (1, 1, 0, 1)$ then the hamming distance between \mathbf{x} and \mathbf{y} , $f_{HD}(\mathbf{x}, \mathbf{y})$, is 1. Two detectors match under the HD rule if their hamming distance is less than r .
2. **r -Contiguous bits Rule:** The r -contiguous bits (RCBITS) rule states that two binary vectors match if they have identical bits in at least r contiguous positions [23]. For example, if $\mathbf{x} = (1, 0, 1, 0, 0, 0, 0, 0)$ and $\mathbf{y} = (0, 1, 1, 0, 0, 1, 1, 1)$ then the number of r contiguous bits between \mathbf{x} and \mathbf{y} , $f_{RCBITS}(\mathbf{x}, \mathbf{y})$, is 3. The RCBITS rule is a very popular matching rule, since it is theoretically simple and lends itself equally to both mathematical and statistical analysis [28].
3. **r -Chunks bits Rule:** The r -Chunks (RCHK) matching rule, as conceptualised by Balthrop et al. [3], was inspired by the RCBITS rule and matching rules for classifier systems developed by Holland [17]. RCHK detectors are specified by a window of size r in which all r bits in the window must match the given string in question. The remaining bit positions are termed “don't cares” and are ignored. An RCHK detector is depicted as a vector, \mathbf{x} , of length r and a starting position w (that is, detection starts at position w and ends at position $w + r - 1$). For example, if $\mathbf{x} = (1, 0, 0, 1)$ and $\mathbf{y} = (1, 0, 0, 1, 0, 0, 0, 0)$ then $f_{RCHK}(\mathbf{y}, \mathbf{x}, 1) = 4$, where $f_{RCHK}(\mathbf{y}, \mathbf{x}, 1)$ is the application of the RCHK rule to vectors \mathbf{x} , \mathbf{y} , and $\mathbf{w} = 1$.

A number of experiments were performed by Balthrop et al. (2002) where they concluded that the RCHK rule performed better than the RCBITS rule for their data sets. It was shown by Esponda et al. (2004) that the RCHK rule subsumes the RCBITS rule. From each of the detection rules presented thus far (HD, RCBITS and RCHK) it is evident that a trade-off between the number of detectors and their affinity threshold exists. Larger affinity thresholds result in more specific matching, whereas smaller affinity thresholds result in more generic matching. In other words, the best values for these parameters are problem dependant and should be fine tuned for each new problem domain [15]. A further critical difference between the matching rules is the number of undetectable strings that they induced, which are termed “holes” by D'haeseleer et al. (1996). Holes are discussed in the next subsection.

2.2. HOLES INDUCED BY MATCHING RULES

Holes do not exist merely because of the limitations of the matching rules used, but also because of the similarity that exists between self and non-self cells. In reality, self and non-self are distributed at great distances from each other and holes exist for any approximate matching rule, even within the NIS [16]. Balthrop et al. (2002) identified two types of holes affecting the RCBITS rule namely length-limited holes and cross-over holes:

1. Cross-Over holes: A crossover hole, \mathbf{h} , occurs when all possible windows (the specified r contiguous positions of an RCHK detector within \mathbf{h}) are crossovers (defined below) of adjacent windows within a particular set of vectors (where the set is either a self-set or a non-self set depending on the AIS algorithm being employed). Given a set, S , of self strings and two vectors, $\mathbf{u}, \mathbf{v} \in S$, a crossover occurs between two adjacent windows, $\mathbf{w}_i = (v_i, v_{i+1}, \dots, v_{i+r-1})$, and, $\mathbf{w}_{i+1} = (u_{i+1}, u_{i+2}, \dots, u_{i+r})$, whenever bits $v_j = u_j \forall j : i + 1 \leq j \leq i + r - 1$ [3].
2. Length Limited Holes: Length-limited holes are holes that arise in full-length detectors, for example detectors which employ the RCBITS or HD rule. A length-limited hole is defined as a vector, \mathbf{h}' , which contains at least one window of r bits not present within the self-repertoire and for which a detector cannot be generated [3]. Take note that the RCHK rule does not induce length-limited holes [3].

It has been speculated by Hofmeyer and Forrest (2000) that MHC plays an important role within the NIS to protect a population of detectors from holes. In Hofmeyer and Forrest's view, MHC is a mechanism through which a single protein can be represented in a different way. Hofmeyer and Forrest went further to conclude that, because different representations induce different holes, leveraging multiple representations will reduce the overall number of holes. Hofmeyer and Forrest implemented the MHC mechanism by defining a permutation mask, $\mathbf{m} = (1, \dots, m_n)$, where each $m_i \in \{1, \dots, n\}$ specifies a new position for bit number i . The function $f_{PERMUTE}(\mathbf{w}, \mathbf{m})$ applies a permutation mask, \mathbf{m} , to a vector, \mathbf{w} . The permutation function is applied by generating a single random permutation mask, \mathbf{m} , for an entire global population of detectors. Each antigen, \mathbf{y} , is first processed by $f_{PERMUTE}(\mathbf{y}, \mathbf{m})$, before being introduced to a population of detectors [15]. A detailed study of the effect of Hofmeyer and Forrest's permutation mask used in conjunction with the NSA was performed by Stibor et al. (2006). Stibor et al. (2006) found that randomly generated permutation masks changed the shape and distribution of the entire data set, thus distorting its semantic meaning and resulting in detectors being randomly distributed within the search space, as opposed to being concentrated around self-regions. Furthermore, Stibor et al. (2006) also doubted whether permutation masks were appropriate at reducing the number of holes within negative selection algorithms by abstracting diversity. Contrary to the view of Stibor et al. (2006), Esponda et al. (2004) showed that:

- the NSA, under the RCBITS rule augmented by permutation masks contains the class of languages recognised by the NSA under the HD rule. That is, the RCBITS rule augmented by permutation masks is able to protect more sets than the HD rule.
- Permutation masks reduced the number of holes induced by both the RCBITS and HD rule.

The fundamental difference between the views of Esponda et al. (2004) and Stibor et al. (2006) is caused by the approach in which they investigated the efficacy of the permutation masks applied to the RCBITS rule. In the former case, Esponda et al. (2004) approached the investigation mathematically, whereas in the latter case Stibor et al. (2006) approached the investigation empirically. It is the view of this paper that both Stibor et al. and Esponda et al. (2004) are correct, in that permutation masks do eradicate holes induced by affinity-matching functions if the permutation induced by the permutation mask is meaningful. That is, the permutation mask should select both adjacent and non-adjacent bits where a relationship exists between the attributes of a self/non-self vector and a detector (in the order induced by the permutation). For example, if a relationship exists between attributes (1, 3, 5, 2, 4) of a self/non-self vector and a detector in a five-dimensional problem space, then it is logical to create a permutation mask that induces such a permutation on the entire self repertoire of strings before generating detectors (by utilizing the NSA). It was argued by Esponda et al. (2004) that such a permutation is indeed very difficult, if not computationally expensive, to infer. While this is true, an approach to generate meaningful permutation masks is illustrated in this paper.

The next section highlights popular metrics used to quantify the performance of AIS algorithms (most notably the NSA class of AIS algorithms).

3. AIS ALGORITHM PERFORMANCE METRICS

The most popular metrics employed by AIS researchers to report on the performance of NSAs and a number of other AIS algorithms are false positives, true positives, false negatives and true negatives:

- False positives (FPs) occur when self-patterns are incorrectly classified as non-self.

- True positives (TPs) occur when self-patterns are correctly classified as self.
- False negatives (FNs) occur when non-self patterns are incorrectly classified as self.
- True negatives (TNs) occur when non-self patterns are correctly classified as non-self.

These measures can be combined in a more meaningful way to create two additional metrics, termed the detection rate (DR) and false-alarm rate (FR), defined by [26] as:

$$DR = \frac{TP}{TP + FN} \quad (1)$$

$$FR = \frac{FP}{FP + TN} \quad (2)$$

The next section describes Forrest et al.'s NSA algorithm and how affinity-matching functions employed by detectors affect the performance of the NSA followed by a discussion on the discriminative power of a detector.

4. THE NEGATIVE SELECTION ALGORITHM

The main premise of the NSA developed by Forrest et al. (1994) is to generate a set of candidate detectors, C , such that $\forall x_i \in C$ and $\forall z_p \in S$, $f_{MATCH}(x_i, z_p) < r$, where x_i is a detector, z_p is a pattern and $f_{MATCH}(x_i, z_p)$ is an affinity-matching function.

Pseudocode for the NSA algorithm is given in Figure 1. Forrest et al.'s (1994) original NSA uses a single global affinity threshold, r , in conjunction with the RCBITS rule for each individual detector within the population of detectors, C . The affinity threshold is determined through a process of trial and error, whereby the threshold yielding the best system performance is chosen as the target affinity threshold. A general framework to aid in choosing an optimum value for r in conjunction with the RCBITS rule was provided by Ayara et al. (2002). While the original NSA utilised the RCBITS rule, AIS researchers have discovered that the choice of the affinity matching function has an impact on the performance of the NSA algorithm [3, 13]. The next subsection, discusses two approaches undertaken by AIS researchers to analyse the performance of the NSA, the former a more mathematical approach and the latter a graphical approach.

```

1  Let counter,  $n_c$ , be the number of self-tolerant detectors to train;
2  Let  $C$  be an empty set of self-tolerant detectors;
3  Let  $r$  be the affinity threshold;
4  Create a training set,  $D_{TRAIN}$ , comprised of self patterns;
5  while  $|C| \neq n_c$  do
6    Randomly generate a detector,  $x_i$ ;
7    matched := false;
8    for each self pattern,  $z_p \in D_{TRAIN}$  do
9      if  $f_{MATCH}(x_i, z_p) > r$  then
10       matched:= true;
11       break;
12     end
13   end
14   if matched = false then
15     Add  $x_i$  to  $C$ ;
16   end
17 end

```

Figure 1. Pseudocode for the Negative Selection Algorithm.

4.1. EFFECT OF AFFINITY MATCHING FUNCTIONS ON THE NEGATIVE SELECTION ALGORITHM

One of the notable advantages of the NSA over many of the other AIS algorithms is that, besides being theoretically simple, the NSA allows any matching function to be employed (although this statement is true for a large majority of AIS algorithms it is not true for all AIS algorithms [6]). Different matching functions, however, induce different detection regions for each detector, $\mathbf{x}_i \in C$, and thus have a direct influence on the performance of the NSA. This section discusses two of the most prominent analyses performed by several AIS researchers on how matching functions influence the performance of the NSA. The original analysis performed by Forrest et al. (1994), which is rooted in probability theory, is discussed first followed by an alternative analysis performed González et al. (2003).

Noting that the NSA is probabilistic, Forrest et al. (1994) derived five equations that can be used to determine how many self tolerant detectors, n_c , need to be generated by the NSA in order to protect a set of self strings, S , with a certain failure probability, P_f . These equations are presented below. The equations derived by Forrest et al. are based on the probability, P_M , that two random strings match in at least r positions, which was defined by Percus et al. (1993, 1996) as:

$$P_M \approx n_{alph}^{-r} \left[\frac{(n-r)(n_{alph}-1)}{(n_{alph}+1)} \right] \quad (3)$$

where n_{alph} is the number of symbols contained within the alphabet of the strings (for example for a binary string n_{alph} is 2) and n is the length of a string. It should be noted that this approximation is only good if $n_{alph}^{-r} \ll 1$. If this constraint is not satisfied, then the exact equation must be used [27]. Equation (3) exhibits two characteristics: there is a linear increase in P_M as n increases, and there is an exponential decrease in P_M as r increases. Similar to equation (3), Wierzchoń (2000) derived an equation to calculate the probability that two random binary strings have a hamming distance of r . This equation can be used if the HD rule is employed with the NSA:

$$P_M = 2^{-n} \sum_{i=r}^n \binom{n}{i} \quad (4)$$

Take note that equations (5) to (9) assume that P_M is calculated using equation (3). The probability of a random string/detector not matching any self-strings within S is given by Forrest et al. (1994) as:

$$P_S = (1 - P_M)^{|S|} \quad (5)$$

The probability that n_c self-tolerant detectors fail to detect an antigen, P_f , is given by Forrest et al. (1994) as:

$$P_f = (1 - P_M)^{n_c} \quad (6)$$

where n_c is the number of self-tolerant detectors to train. If P_M is small and n_c is large, then,

$$P_f \approx e^{(-P_M n_c)} \quad (7)$$

The number of self-tolerant detectors, n_c , needed to attain a certain failure probability, P_f , and matching probability, P_M , is given by Forrest et al. (1994) as:

$$n_c = \frac{-\ln(P_f)}{P_M} \quad (8)$$

The number of initial detectors, n_r , before censoring (i.e. applying the NSA to the detectors to verify whether any of them are not self-tolerant), needed to generate n_c detectors is given by Forrest et al. (1994) as:

$$n_r = \frac{-\ln(P_f)}{P_M(1 - P_M)^{|S|}} \quad (9)$$

One of the major disadvantages of the NSA is that if P_M , P_f , and n_c are fixed, then an exponential increase in n_r can be observed. Forrest et al. state that this can also be viewed in a positive light, in the sense that if such a set of n_c detectors were generated by a supercomputer, then it is highly unlikely that a change to self would go undetected [11]. Interestingly, based on a study of the suitability of the NSA for network intrusion detection, Kim

and Bentley (2001) cited this factor as the primary reason for the NSA having failed to perform effectively. Due to the NSA suffering from a severe-scaling problem, Kim and Bentley concluded that the NSA should rather be used as a filter for invalid detectors as opposed to the generation of detectors. There are several variations of the NSA in existence which overcome the severe-scaling problem, like the Linear Time-Detector Generating Algorithm [6]. In contrast to Forrest et al.'s (1994) approach, González et al.'s (2003) methodology attempts to visualise the shape space defined by Perelson and Oster (1979). In terms of Perelson and Oster's shape-space theory, detectors should be distributed throughout the entire shape space such that the detection region of each detector is able to detect a number of structurally similar antigen [24]. The shape space, however, is rarely two-dimensional and, consequently, the process of distributing detectors throughout the shape space is a non trivial process. The approach developed by González et al. (2003) can be described as follows: Any point (x, y) taken from a problem space corresponding to the domain $[0.0, 1.0]^2$ can be mapped to a binary string, $b_0, b_1 \dots b_7, b_8, b_9 \dots b_{15}$, of length 16 where the first eight bits encode the integer value $\lfloor 255 \cdot x + 0.5 \rfloor$ and the last eight bits encode the integer value $\lfloor 255 \cdot y + 0.5 \rfloor$. By using the mapping to study the results of binary matching rules with regards to different training sets, González et al. (2003) drew the following conclusions:

- The binary matching rules studied, namely, RCBITS, RCHK, and HD, cannot produce a good generalisation of the self-space, resulting in poor coverage of the non-self space. The reason that the binary matching rules do not produce a good generalisation of the self-space is that they are not able to accurately capture the affinity relation employed in the real space within the self/non-self (binary space).
- The matching rule used by NSA needs to be chosen in such a way that the affinity relationship between points in the problem space is preserved when the relationship is transposed to the self/non-self space.

Over and above the analyses performed by Forrest et al. (1994) and González et al. (2003) discussed above, another important concept known as the “discriminative power of a detector” was defined by Wierzchoń (2000) and is discussed in the next subsection.

4.2. THE DISCRIMINATIVE POWER OF A DETECTOR UNDER THE RCBITS RULE

Wierzchoń (2000) defines the discriminative power of a detector as the number of unique strings detected by a detector using the RCBITS rule. In his definition Wierzchoń refers to a template of order r . A template of order r is a string of length n consisting of $n - r$ blank symbols (each blank symbol is represented with a ‘*’). A template is specified by using the symbol, $t_{i,w}$, where w denotes a string composed of r bits and i denotes the starting position of the w bits. The discriminative power of a detector can be found by counting the number of unique strings recognised by each template, $t_{i,w}$, induced by a detector, x .

As an example, consider the detector 001101 and let $r = 3$. The detector induces the following templates: $t_{1,001} = 001***$, $t_{2,011} = *011**$, $t_{3,110} = **110*$, and $t_{4,101} = ***101$. The first template, $t_{1,001} = 001***$, recognises 2^{n-r} unique strings. The second template, $t_{2,011} = *011**$, matches strings $s_1 = 0011**$ and $s_2 = 1011**$. However, s_1 is also recognised by $t_{1,001}$. Hence the total number of strings recognised by $t_{2,011}$ is halved. Following this reasoning, Wierzchoń (2000) found that the discriminative power of a detector, under the RCBITS rule, is equal to $2^{n-r-1} \times (2 + n - r)$.

The feature-detection rule is discussed in section 5. The mathematical properties associated with the feature-detection rule are then discussed. This is followed by a discussion of the interrelationship between the feature-detection rule and other affinity-matching functions discussed in this paper. Section 5 concludes with a discussion on positional-bias introduced by the feature-detection rule and how it is addressed in this paper.

5. THE FEATURE DETECTION RULE

The feature-detection rule differs vastly from the RCHK, HD, and RCBITS rules in that it uses the interrelationships between antigen fragments and a detector to ascertain whether the antigen is detected by the candidate detector. To illustrate the characteristics of the feature-detection rule, consider the following fictitious study. Suppose that a study was carried out on a sample of individuals who had developed cancer and a sample of individuals who had not developed cancer in their lifetimes. The purpose of the study was to develop an algorithm that could deduce

whether a person had indeed developed cancer by looking at an instance of the attributes/characteristics captured by the experimental study. Assume that the study interviewed each individual within a sample and captured the following data:

- Attribute 1: Does the individual smoke?
- Attribute 2: Does the individual work more than 60 hours per week?
- Attribute 3: Does the individual have a companion?
- Attribute 4: Does the individual drink alcohol more than six times per week?
- Attribute 5: Does the individual take frequent holidays?
- Attribute 6: Does the individual sleep at least eight hours per day?
- Attribute 7: Does the individual eat five portions of fruit and vegetables each day?
- Attribute 8: Does the individual exercise less than three times per week?

Consider Figure 2, which depicts such a group of attributes and their relative values (note that each attribute is a binary attribute). Figure 2 represents a particular individual having cancer if

- the individual smokes,
- the individual drinks alcohol more often than six times per week,
- the individual sleeps less than eight hours per day, and
- the individual does not exercise three times per week.

Some of the attributes are relevant to the problem that the algorithm is trying to solve, whereas others are irrelevant. It is also possible that a group of attributes and the value that each attribute carries are relevant to the outcome of the problem. The feature-detection rule refers to the characteristics described in the above example as features. From the above, the values of features 1, 4, 6 and 8 are relevant to the outcome of the problem, whereas the values of features 2, 3, 5 and 7 are irrelevant to the outcome of the problem. If an AIS algorithm were used to generate detectors to determine whether an individual has cancer, based on the values of the 8 attributes, it would in fact be more effective to:

- pre-process each self-string, comprised of all 8 features, into a shorter self-string comprising of features 1, 4, 6, and 8, and
- generate detectors, of length 4, against the pre-processed self-strings by employing the RCBITS rule.

The above example illustrates the premise upon which the feature-detection rule is based. The feature-detection rule is applied by:

- pre-processing a string/vector into a shorter string/vector consisting of features relevant to the outcome of the problem (relevant features can be determined by using mathematical techniques such as principle component analysis etc.) currently under consideration (see the definition of the feature vector, \mathbf{p} , below); and
- applying the RCBITS rule to the newly generated string/vector (consisting of relevant features) and a detector, \mathbf{x} , to determine whether the affinity between \mathbf{x} and the newly generated string/vector is greater than r , the affinity threshold.

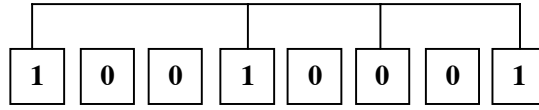


Figure 2. Overview of the feature-detection rule. The numbers at the bottom of the figure represent each of the 8 binary attributes. The binary attributes are interconnected if they are relevant to the outcome of the problem.

The definition of the feature-detection rule, $f_{FEATURE}$, is formally stated below: Let $f_{SELECT}(\mathbf{w}_1, \mathbf{p})$ be a function such that, given a binary vector $\mathbf{w}_1 = (w_1, w_2, \dots, w_n)$ and a vector of integer positions $\mathbf{p} = (p_1, p_2, \dots, p_{n'})$, representing the positions of relevant features, where $n' \leq n$, the function constructs a vector $\mathbf{w}'_1 = (w_{p_1}, w_{p_2}, \dots, w_{p_{n'}})$ by using the positions stipulated in \mathbf{p} . In other words the feature-detection rule generates a vector \mathbf{w}'_1 by selecting elements of \mathbf{w}_1 as dictated by \mathbf{p} . For example if $\mathbf{w}_1 = (1, 0, 1, 0, 1)$ and $\mathbf{p} = (1, 3, 5)$ then $f_{SELECT}(\mathbf{w}_1, \mathbf{p}) = (1, 1, 1)$. Now consider an antigen, $\mathbf{y} = (y_1, y_2, \dots, y_n)$ (binary vector), a detector $\mathbf{x} = (x_1, x_2, \dots, x_{n'})$ (binary vector) and a vector \mathbf{p} (integer vector) of dimensionality n' , where the dimensionality of the problem space/antigen is n , the dimensionality of a detector is n' , $n' \leq n$ and \mathbf{p} comprises a subset of feature positions of \mathbf{y} , that is, $\mathbf{p} = (y_i, y_{i+1}, \dots, y_j)$, where each y_i appears only once and $i \geq 1$ and $j \leq n$. Antigen \mathbf{y} and detector \mathbf{x} match under the feature-detection rule, $f_{FEATURE}$, if $f_{RCBITS}(\mathbf{x}, f_{SELECT}(\mathbf{y}, \mathbf{p})) \geq r$. In other words antigen \mathbf{y} and detector \mathbf{x} match if there are r contiguous features. Consider the example where

$$\begin{aligned} \mathbf{x} &= (1, 1, 0, 1) \\ \mathbf{y} &= (1, 0, 0, 1, 0, 0, 0, 1) \\ \mathbf{p} &= (1, 4, 6, 8) \\ r &= 2 \end{aligned}$$

Then,

$$f_{FEATURE}(\mathbf{x}, \mathbf{y}, \mathbf{p}) = f_{RCBITS}((1, 1, 0, 1), f_{SELECT}((1, 0, 0, 1, 0, 0, 0, 1), (1, 4, 6, 8))) = 4$$

The following can be stated for the example above assuming that the feature-detection rule is not used:

- Depending on which features are related and whether they occur in close proximity to one another, the RCBITS rule would not be a good choice of an affinity-matching function between \mathbf{y} and \mathbf{x} . For example, if an exclusive relationship existed between feature 1 and feature 8, then only one $\mathbf{x} = (1, 0, 0, 1, 0, 0, 0, 1)$ with $r = 8$ would be able to detect \mathbf{y} . But such a detector is too specific to antigen \mathbf{y} . The same applies to the RCHK rule, since the RCHK rule subsumes the RCBITS rule [9].
- The HD rule has the ability to capture the relationship between different features, of a self/non-self vector and a detector, only if the difference between the features with no relationship is less than a particular threshold, r . To illustrate what is meant by this statement, suppose that there are two vectors \mathbf{y} and \mathbf{x} where there is a relationship between feature 1 and feature 8, no relationship between features 2..7, and that $r = 2$. The HD rule is based upon the number of bits that differ, thus feature 1 and feature 8 must be equal in both \mathbf{y} and \mathbf{x} and at most 1 feature between bits 2..7 may be different.

Determining a subset of relevant features, \mathbf{p} , corresponds to a well known feature extraction problem and is a vital component underpinning the performance of the feature-detection rule. The approach used to determine \mathbf{p} is discussed in section 5.2. Subsection 5.1, discusses the discriminative power and the matching probability, P_M , induced by the feature-detection rule.

5.1. DISCRIMINATIVE POWER AND MATCHING PROBABILITY OF DETECTORS UNDER THE FEATURE DETECTION RULE

The discriminative power of a detector employing the feature-detection rule can be calculated using a similar argument presented by Wierchoń (2000). From the example in section 5, where $\mathbf{x} = (1, 1, 0, 1)$ and $\mathbf{p} = (1, 4, 6, 8)$, $t_1 = 1**1*0**$ induces two templates, $t_1' = 1**1*0*1$ and $t_1'' = **1*0*0$. Similarly, $t_2 = ***1*0*1$ induces two templates, $t_2' = 0**1*0*1$ and $t_2'' = 1**1*0*1$. However, since $t_1' = t_2''$, the number of strings that t_2 can detect

are effectively halved. This argument can be generalised to any arbitrary detector. The probability that a detector, \mathbf{x} , matches an antigen, \mathbf{y} , can then be calculated by noting that a detector \mathbf{x} of length n' induces $n' - r + 1$ templates. The first template can recognise 2^{n-r} strings and each subsequent template, $(n' - r)$ in total, can only recognise $\frac{2^{n-r}}{2}$ strings. There are 2^n strings in total. Therefore,

$$P_M = \frac{(2^{n-r} + (n' - r).2^{n-r-1})}{2^n} \quad (10)$$

To compare the matching probability of the feature-detection rule with the other detection rules discussed in this paper, the matching probabilities for the RCBITS and RCHK rules are presented below. Take note that for the RCBITS rule, the exact equation is used (assuming that $n_{alph} = 2$) as opposed to the approximation provided by Percus et al. (1993, 1996) given in equation (3). For the RCBITS rule,

$$P_M = \frac{(2^{n-r} + (n - r).2^{n-r-1})}{2^n} \quad (11)$$

The matching probability, P_M , of the RCHK rule can easily be calculated by noting that the length of a detector under the RCHK rule is equal to the affinity threshold, r . Thus, each detector of length r can recognise 2^{n-r} strings. For the RCHK rule:

$$P_M = \frac{2^{n-r}}{2^n} \quad (12)$$

From equations (10) and (11), the matching probability, P_M , under the feature-detection rule is greater than P_M under the RCBITS rule if $n' > n$. Furthermore, from equations (10) and (12) the matching probability, P_M , under the feature-detection rule is greater than P_M under the RCHK rule if $n' > r$. Interestingly, P_M for the feature-detection rule is calculated in a similar manner to P_M for the RCBITS rule (see equations (10) and (11)). The feature-detection rule is therefore expected to suffer from the same scaling problems as the NSA (from a purely mathematical viewpoint). The matching probability, P_M , for the HD rule was given by equation (4). P_M under the feature-detection rule is greater than P_M under the HD rule if:

$$\frac{2^{n-r} + (n' - r).2^{n-r-1}}{2^n} > \frac{\sum_{i=r}^n \binom{n}{i}}{2^n} \quad (13)$$

From equations (10) and (13) and noting that $\sum_{i=r}^n \binom{n}{i} < 2^n$, the matching probability, P_M , under the feature-detection rule is greater than P_M under the HD rule if $n' > 2^{r+1} + r - 2$.

This effectively means that the probability of a randomly generated detector matching a self-string under the feature-detection rule is lower than that of a detector using either the RCHK, RCBITS (in all circumstances) and the HD rule (in most circumstances, i.e. where r is sufficiently small and n' is sufficiently large, which is most often not the case under the HD rule, because leveraging small r values will generate detectors that are too reactive producing poor DR and FR values).

The next subsection contrasts the feature-detection rule against random permutation masks in order to place the feature-detection rule in context.

5.2. THE FEATURE DETECTION RULE AND PERMUTATION MASKS

The most common affinity-matching functions used in the NSA, namely, the RCBITS and the RCHK rules, induce holes. Holes can be reduced by using a permutation mask against an antigen to reorder the bits of the antigen. Even though permutation masks are a mathematically feasible means of overcoming both crossover and length-limited holes, major flaws exist in the way in which they are implemented. Permutation masks are currently implemented by generation of an individual random permutation mask and application of the individual random permutation mask to a population of detectors generated under the NSA. A consequence of the application of permutation masks in this manner is that their benefits are occluded by what appears to be a shattering of the entire self-space (by randomly changing the shape of the entire self-space with a randomly generated permutation mask).

Shattering of the self-space attributed to permutation masks by Stibor et al. (2006) can be explained as follows. A permutation mask changes the form of a shape space. Using a single random permutation mask for an entire set of detectors generated under the NSA is equivalent to taking a wild guess by trying to infer a single and meaningful

alternate representation for the entire problem space. There may in fact be multiple representations of the problem space that are relevant to the problem at hand. For example, there may be multiple relationships involving entirely different subsets of features of self/non-self vectors and detectors within a particular problem domain. If the RCBITS rule and the RCHK rule are considered in the same context as the example presented in Figure 2, then two things would immediately become evident: (1) the RCBITS rule and RCHK rule are only effective when relationships between adjacent features of self/non-self vector and a detector exist, and (2) a random permutation of the problem space will in fact increase the efficacy of both the RCBITS and the RCHK rule where relationships between non-adjacent features of a self/non-self vector and a detector exist, because the result of the permutation in respect of the problem space can render two non-adjacent attributes adjacent. More random permutations are thus equivalent to finding more relationships between non-adjacent features. It was stated by Stibor et al. (2006) that finding a meaningful permutation mask to apply to an entire set of detectors is computationally expensive and, in fact, infeasible. Although their statement does hold true, the problem of finding a meaningful permutation mask can be approached from another angle.

Following the argument presented in the previous point, the RCBITS rule and the RCHK rule should actually be viewed as affinity-matching functions, which exploit the relationships that exist between adjacent features of a self/non-self vector and a detector. Relationships between non-adjacent features can be discovered through the application of a random permutation mask to an individual detector. Following the argument of Stibor et al. (2006), the aim should be to discover several permutation masks based on the conjunction of the problem space and the features of an individual detector, meaning that the problem is more computationally expensive than previously thought. Finding a vector of meaningful features, \mathbf{p} , to use in the feature-detection rule is equivalent to finding a meaningful permutation mask to apply to a particular problem domain. With the only difference being that the length of \mathbf{p} , n' , is less than or equal to the length of an artefact (an antigen, detector or self-string), n , resident within a particular problem domain. The approach used in this paper, which is surprisingly simple, is to couple the generation of a random permutation mask with the generation of each individual detector under the NSA. In other words, a detector is generated together with a random permutation mask and is checked against the entire self-set to ensure that the detector is not activated by a self-string before being added to the resultant repertoire of detectors. This means that the NSA is tasked with learning a permutation mask for each detector being generated.

The modified version of the NSA algorithm, described above, can be implemented by making the following changes to the pseudocode illustrated in Figure 1:

1. Replace line 6 with: Randomly generate a detector, x_i , and a feature vector, p_i ;
2. Replace line 9 with: **if** $f_{FEATURE}(x_i, z_p, p_i) > r$ **then**

From equations (10) and (12) it is evident that if $r = n'$, then a detector generated under the feature-detection rule is equivalent to a detector generated under the RCHK rule (the RCHK rule is used because it subsumes the RCBITS rule [9]), where each detector under the RCHK rule has a random permutation mask. But, if $n' > r$, then the discriminative power of a single detector under the feature-detection rule is equal to $n' - r$ RCHK detectors. Since the feature-detection rule is equivalent to the RCHK rule, with each detector having a random permutation mask in the worst case (when $r = n'$), it follows that the feature-detection rule produces less holes than the RCHK, RCBITS and the HD rule.

The next subsection discusses how the feature-detection rule introduces positional bias and how it is addressed in this paper.

5.3. POSITIONAL BIAS INTRODUCED BY THE FEATURE DETECTION RULE

The feature-detection rule is a more efficient form of the RCBITS rule, in that the feature-detection rule applies the RCBITS rule to a set of features \mathbf{p} of an artefact (which is an element of the problem domain), as opposed to all of the attributes comprising the artefact. The RCBITS rule however, introduces positional bias as discussed by Freitas et al. (2003). To illustrate this, consider the example in Figure 2. Based on Figure 2, the values of features 1, 4, 6 and 8 are relevant to the outcome of the problem, i.e. feature 1 must bear a value of 1, feature 4 must bear a value of 1, feature 6 must bear a value of 0 and feature 8 must bear a value of 1. Now if an affinity threshold of $r = 4$ is employed by the feature-detection rule and the features are ordered in the manner presented above, i.e. in the order of feature 1, feature 4, feature 6 and feature 8, then it is evident that there is positional bias, because if the features were selected in a different order, for example in reverse order, then the same detector that would have matched the features presented in their original sequence, i.e. 1101, will not match the features if they were presented in reverse order 1011.

Now consider how the feature-detection rule is applied within the context of this paper. The features together with the resultant detector (each detector has its own set of features) are generated by employing the NSA, i.e. the features are selected randomly against a randomly generated detector, meaning that regardless of the order in which the features are presented, the detector will only be activated, under the feature-detection rule, if its attributes match the re-ordered features. Thus if the feature-detection rule is applied within the context of the modified version of the NSA, discussed in section 5.2, it overcomes positional bias. To illustrate this example, consider the following scenario (with reference to Figure 2) with two randomly generated detectors: $\mathbf{x}_1 = (1, 1, 0, 1)$ with a position vector of $\mathbf{p}_1 = (1, 4, 6, 8)$ and $\mathbf{x}_2 = (1, 0, 1, 1)$ with a position vector of $\mathbf{p}_2 = (8, 6, 4, 1)$, an affinity threshold, $r = 4$ and an antigen $\mathbf{y} = (1, 0, 0, 1, 0, 0, 0, 1)$. Regardless of the fact that both \mathbf{x}_1 and \mathbf{x}_2 select the same set of features, albeit in a different order, both \mathbf{x}_1 and \mathbf{x}_2 are activated by antigen \mathbf{y} .

The next section discusses the experimental results obtained by conducting a number of experiments using the feature-detection rule, the RCHK rule with and without random permutation masks for each detector, the RCHK rule with a single random permutation mask applied uniformly to a population of detectors, and the HD rule.

6. EXPERIMENTAL RESULTS

The objectives of the experiments conducted are to:

- demonstrate the efficacy of the feature-detection rule (denoted as FDR) in contrast to the RCHK rule (with no permutation masks) and the HD rule;
- demonstrate the efficacy of the feature-detection rule in contrast to the RCHK rule (where each detector has a random permutation mask). For the purposes of this paper, scenarios in which a random permutation mask are used in conjunction with the RCHK rule are denoted as RCHK (MHC). Conversely, scenarios in which MHC masks are not used in conjunction with the RCHK rule are denoted as RCHK (No MHC);
- demonstrate how the application of an individual global MHC mask applied to a set of already generated detectors impedes the performance of the set of detectors. This point is important because it will validate the assertion made by this paper that MHC masks are being applied incorrectly within the context of the NSA. Scenarios in which a global MHC mask is applied to a pre-generated set of detectors is denoted as RCHK (Global MHC); and
- demonstrate that the performance (detection rate and false-alarm rate) of the feature-detection rule is equivalent to the RCHK rule (where each detector has a random permutation mask) at worst case.

The rest of this section is organised as follows: The experimental procedure used to conduct each experiment is described in section 6.1, followed by a discussion of the car evaluation experiment, iris experiment, Wisconsin breast cancer experiment, glass experiment and mushroom experiment in sections 6.2 - 6.6 respectively. A summary of the entire set of experiments is presented in section 6.7.

6.1. EXPERIMENTAL PROCEDURE

An experiment comprises five scenarios where a scenario pertains to a particular high-level objective:

1. Training a set of detectors with the modified version of the NSA, discussed in section 5.2, utilising the feature-detection rule.
2. Training a set of detectors with the NSA utilising the HD rule.
3. Training a set of detectors with the NSA utilising the RCHK rule with no permutation masks.
4. Training a set of detectors with the NSA utilising the RCHK rule with a single global permutation mask. Take note that the test sets within this particular scenario is executed by firstly generating a set of detectors and then applying a single randomly generated global permutation mask to the generated detector set.
5. Training a set of detectors with the NSA utilising the RCHK rule where each detector has its own randomly generated permutation mask.

A scenario is comprised of a number of test groups. Each test group utilises a different set of parameters (e.g. different values of r) to test the scenario. A test group in turn is comprised of several tests. Each test is executed with the parameters stipulated by its test group in addition to using a particular target population size, n_c . That is, different tests have different n_c values. The last test executed within a test group has the largest n_c value and is called the target test. The objective of the target test is to measure the performance of the NSA given a maximum n_c value in conjunction with the parameters associated with the target test. A new test set and training set was randomly created for each new execution of a test, in each test group, as follows:

- Each data set relating to a particular experiment was fragmented into a self and non-self set (explained for each experiment).
- A training set was created based on the self-data by randomly selecting 70% of the original self-set.
- The test set comprised the remaining 30% of the original self-set and a non-self set associated with the training data.

For each test executed within a particular test group, the TP, FP, FN, DR, and FR metrics were recorded. To ensure that the results are statistically significant, each test within a particular test group was executed 30 times, and the result of each metric records the mean value and the standard deviation of the metric.

When comparing the best target tests across different scenarios, within a particular experiment, to each other the Mann-Whitney U test (with continuity correction) was employed to test several hypotheses, tabulated in Table 1, across the FDR, HD, RCHK (No MHC), RCHK (Global MHC) and RCK rules. Each hypothesis is numbered independently, the alternate hypothesis corresponding to each hypothesis is suffixed with an “a”, and the average, μ , relating to a detection rule and metric is denoted as: $\mu_{DetectionRule}^{Metric}$. When testing each hypothesis a critical one-tailed value of $z_{0.05} = 1.96$ was used.

The data sets become increasingly more complex for each successive experiment performed in this section. The same data sets originally used by Graaff (2004) to test the efficacy of an AIS that evolved its detectors, are used by this paper. Graaff collected the data sets from the UCI Machine Learning Repository [1] and converted them into binary strings by using a binning technique. Binning is applied by dividing a floating-point attribute, c , into b bins/groups and then using equation (14) to discern which bin, $x_{c,j}$, a pattern belongs to:

$$G(x_{c,j}) = \frac{x_{c,j} - \min_{j=1,\dots,J}\{x_{c,j}\}}{\max_{j=1,\dots,J}\{x_{c,j}\} - \min_{j=1,\dots,J}\{x_{c,j}\}} \times b \quad (14)$$

where $x_{c,j}$ is the floating point value of attribute, c , in pattern, j , and J is the total number of patterns in the data set. The value returned by $G(x_{c,j})$ is rounded to the closest integer and converted to a binary string using standard binary encoding (The number of bits, n , needed to encode b is given by $\frac{\log(b)}{\log(2)}$).

The parameters used to test the NSA within a particular scenario were chosen by applying the framework suggested by Ayara et al. (2002) to choose an optimum affinity threshold, r , for the NSA under the RCBITS rule. Due to the mathematical similarity between the RCHK rule (regardless of whether a permutation mask is applied to a detector or set of detectors) and the feature-detection rule, the same set of affinity thresholds were used across all of the test groups in scenarios 1, 3, 4 and 5 for each particular experiment. Conversely, due to the differences in the mechanisms employed by the RCHK rule/feature-detection rule and the HD rule to determine whether two binary strings are activated by one another, the same affinity threshold, r , cannot be reused to compare the performance of the HD rule to either the RCHK rule or the feature-detection rule. Instead, the performance of the HD rule was optimised (by choosing an optimal r value) and scenarios were compared to one another within a particular experiment based on each scenario's best performing test group. The best performing test group has the greatest average $DR - FR$ value for its target test (the last test executed within the test group). Take note that the worst performing test group has the lowest average $DR - FR$ value for its target test. Five different experiments were performed, each discussed in a separate sub-section.

Table 1. Target Tests - Hypotheses

Hypothesis
$H_0 : \mu_{FDR}^{DR} \neq \mu_{HD}^{DR}$
$H_{0a} : \mu_{FDR}^{DR} \geq \mu_{HD}^{DR}$
$H_1 : \mu_{FDR}^{FR} \neq \mu_{HD}^{FR}$
$H_{1a} : \mu_{FDR}^{FR} < \mu_{HD}^{FR}$
$H_2 : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \neq \mu_{HD}^{DR} - \mu_{HD}^{FR}$
$H_{2a} : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \geq \mu_{HD}^{DR} - \mu_{HD}^{FR}$
$H_3 : \mu_{FDR}^{DR} \neq \mu_{RCHK(NoMHC)}^{DR}$
$H_{3a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(NoMHC)}^{DR}$
$H_4 : \mu_{FDR}^{FR} \neq \mu_{RCHK(NoMHC)}^{FR}$
$H_{4a} : \mu_{FDR}^{FR} < \mu_{RCHK(NoMHC)}^{FR}$
$H_5 : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \neq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$
$H_{5a} : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \geq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$
$H_6 : \mu_{FDR}^{DR} \neq \mu_{RCHK(GlobalMHC)}^{DR}$
$H_{6a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(GlobalMHC)}^{DR}$
$H_7 : \mu_{FDR}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{FR}$
$H_{7a} : \mu_{FDR}^{FR} < \mu_{RCHK(GlobalMHC)}^{FR}$
$H_8 : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$
$H_{8a} : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \geq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$
$H_9 : \mu_{FDR}^{DR} \neq \mu_{RCHK(MHC)}^{DR}$
$H_{9a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(MHC)}^{DR}$
$H_{10} : \mu_{FDR}^{FR} \neq \mu_{RCHK(MHC)}^{FR}$
$H_{10a} : \mu_{FDR}^{FR} < \mu_{RCHK(MHC)}^{FR}$
$H_{11} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \neq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$
$H_{11a} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \geq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$

6.2. CAR EVALUATION EXPERIMENT

The car evaluation data set imposes a valuation of cars based upon three characteristics, namely price, technical characteristics, and comfort. The price factor includes the price of the car and the cost of maintaining the car. The technical factor addresses the safety of the car. The comfort factor is concerned with the car's carrying capacity and the size of the car's boot in terms of luggage capacity. The data set consists of 1728 patterns, distributed between four classes: good, acceptable, unacceptable and very good. Each pattern in the data set is comprised of 6 nominal attributes, where the first three attributes each have 4 distinct values (represented by numbers 1..4 respectively) and the last three attributes each have 3 distinct values (represented by numbers 1..3 respectively). Graaff (2004) converted each pattern into a binary string of length 12, i.e. each attribute was encoded using 2 bits. The data sets were processed further to create a single self-set and non-self set as follows:

- Acceptable.self: Contains 384 patterns relating to the acceptable class.
- Acceptable.non-self: Contains all the patterns related to the unacceptable, good and very good classes. The set contains 1344 patterns in total.

The car evaluation data set, re-formatted as a set of binary strings, is the least complex data set, since each pattern consists of 12 binary-valued attributes. The results (mean, standard deviation and Mann-Whitney U test) of the best performing test group for each scenario are summarised in Table A.B.2, Table A.B.7 and Figure A.A.3 respectively. The results show that:

- The feature detection rule exhibits higher average *DR* values than the other detection rules (HD Rule, RCHK (No MHC) Rule, RCHK (Global MHC) Rule and RCHK Rule (MHC)).
- There is no statistically significant difference between the average *FR* value exhibited by the feature detection rule and the HD rule.
- The feature detection rule is the best performing rule because it exhibits a higher average *DR – FR* value than the other detection rules. Consequently the RCHK (Global MHC) is the worst performing rule because it exhibits the lowest average *DR – FR* value.

6.3. IRIS EXPERIMENT

The iris data set consists of three distinct classes, namely veriscolor, virginica and setosa. The setosa class is linearly separable from the veriscolor and virginica classes, whereas the veriscolor and virginica classes are not linearly separable from one another. Each pattern within the data set, comprising of four continuous attributes, was converted into a binary string of length 20 [14], i.e. each attribute was encoded using 5 bins. The data sets were processed further to create a single self-set and non-self set as follows:

- Virginica.self: Contains 50 patterns relating to the virginica class.
- Virginica.nonSelf: Contains all the patterns related to the veriscolor and setosa classes. The set contains 100 patterns in total.

The iris data set, re-formatted as a set of binary strings, is more complex than the car evaluation data set, because each pattern comprises 20 binary valued attributes. The results of the best performing test group for each scenario are summarised in Table A.B.3, Table A.B.8 and Figure A.A.4 respectively. The results show that:

- There is no statistically significant difference between the average *DR* value exhibited by the feature-detection rule and HD rule.
- The average *DR* value exhibited by the feature-detection rule is greater than the average *DR* value exhibited by the RCHK (No MHC), RCHK (Global MHC) and RCHK (MHC) rules.
- The average *FR* value exhibited by feature-detection rule is less than the average *FR* of the other detection rules.

- The feature detection rule is the best performing rule because it has the highest average $DR - FR$ values than the other detection rules.
- The HD rule is the worst performing rule.

6.4. WISCONSIN BREAST CANCER EXPERIMENT

The Wisconsin breast-cancer data set comprises 699 patterns distributed between two classes, namely, benign and malignant. Each pattern consists of nine nominal attributes, where the values of the attributes are in the range [1, 10] (take note that the attribute values are integers). The tenth attribute indicates the target class of the pattern and was subsequently removed from the data set. The data set has 16 missing values for the “bare nuclei” attribute. Each pattern was converted into a binary string of length 36 [14], i.e. each attribute was encoded using 4 bits. Patterns containing missing attributes were represented as binary strings, solely comprising of 1s.

The data sets were processed further to create a single self-set and non-self set as follows:

- Benign.self: Contains 458 patterns relating to the benign class.
- Benign.non-self: Contains all the patterns related to malignant class. The set contains 241 patterns in total.

The results of the best performing test group for each scenario are summarised in Table A.B.4, Table A.B.9 and Figure A.A.5 respectively. The results show that:

- The feature detection rule exhibits higher average DR values than the other detection rules.
- The feature detection rule exhibits lower average DR values than the other detection rules.
- The feature detection rule is the best performing rule because it has the highest average $DR - FR$ values than the other detection rules.
- The RCHK (Global MHC) rule is the worst performing rule.

6.5. GLASS EXPERIMENT

The glass data set comprises 214 patterns, distributed between 7 classes, with each class being a specific glass type. Each pattern, originally consisted of nine continuous-valued attributes, was converted into a binary string of length 45 [14], i.e. each attribute was divided into 32 bins. No patterns were recorded in the vehicle_windows_non_float class subset and, subsequently, this subset was excluded from the other data sets. The data sets were processed further to create a single self-set and non-self set as follows:

- Building_window_float.self: Contains 70 patterns relating to the building_window_float class.
- Building_window_float.non-self: Contains all the patterns related to the building_window_non_float, containers, headlamps, tableware and vehicle_window_float classes. The set contains 144 patterns in total.

The results of the best performing test group for each scenario are summarised in Table A.B.5, Table A.B.10 and Figure A.A.6 respectively. The results show that:

- There is no statistically significant difference between the average DR value exhibited by the feature-detection rule and other detection rules.
- There is no statistically significant difference between the average FR value exhibited by the feature-detection rule and the HD rule.
- The feature-detection rule exhibits a lower average FR value than the RCHK (No MHC), RCHK (Global MHC), and RCHK (MHC) rules.
- There is no statistically significant difference between the average $DR - FR$ value exhibited by the feature-detection rule and the HD rule, RCHK (No MHC), RCHK (Global MHC) rules.
- The average $DR - FR$ value exhibited by the feature-detection rule is greater than the RCHK (MHC) rule.
- The HD rule is the worst performing rule.

6.6. MUSHROOM EXPERIMENT

The mushroom data set comprises hypothetical samples, corresponding to 23 species of mushroom described by 22 nominal-valued attributes. Each pattern is classified as being edible, non-edible or unknown. The number of distinct values/categories that each attribute has is portrayed by the following vector of length 22:

[6, 4, 10, 2, 9, 4, 3, 2, 12, 2, 7, 4, 4, 9, 9, 2, 4, 3, 8, 9, 6, 7],

where each element in the vector corresponds to a particular attribute. The number of bits used to encode each attribute is given by the following vector:

[3, 2, 3, 1, 3, 2, 2, 1, 4, 1, 3, 2, 2, 3, 3, 1, 2, 2, 3, 3, 3, 3],

resulting in a binary string of length 57 [14]. The data sets were further processed to create a single self-set and non-self set as follows:

- Edible.self: Contains 4 208 patterns relating to the edible class.
- Edible.non-self: Contains all the patterns related to poisonous class. The set contains 3 916 patterns in total.

The mushroom data, re-formatted as a set of binary strings, is by far the most complex data set, in that it comprises 57 binary-valued attributes and contains a large number of both self (4208) and non-self data (3916). The results of the best performing test group for each scenario are summarised in Table A.B.6, Table A.B.11 and Figure A.A.7 respectively. The results show that:

- There is no statistically significant difference between the average *DR* value exhibited by the feature-detection rule and the HD rule.
- There is no statistically significant difference between the average *DR* value exhibited by the feature-detection rule and the RCHK (MHC) rule.
- The feature-detection rule has a higher average *DR* than the RCHK (No MHC) and RCHK (Global MHC) rules.
- The average *FR* value exhibited by the feature-detection rule is less than the average *FR* exhibited by the other detection rules.
- The feature-detection rule is the best performing rule because it exhibits the highest average *DR* – *FR* value than the other detection rules.
- The RCHK (Global MHC) is the worst performing rule.

6.7. EXPERIMENTAL RESULTS CONCLUSION

The experiments performed in this section showed that the feature-detection rule was superior (with regards to its *DR* and *FR* values) to the RCHK (MHC), RCHK (No MHC), RCHK (Global MHC) and HD for 4 experiments (Car Evaluation, Wisconsin Breast Cancer, Iris and Mushroom Experiments), better than the RCHK (MHC) rule and equivalent to the HD, RCHK (No MHC) and RCHK (Global MHC) in 1 experiment (Mushroom experiment). The application of a single global permutation mask to a set of pre-generated detectors (under the NSA) consistently produced poor results across all of the data sets, thus reaffirming the assertion made in this paper that permutation masks need to be included in the learning process.

7. CONCLUSION AND FUTURE WORK

The Negative Selection Algorithm, introduced by Forrest et al. (1994) has gained significant momentum in the AIS community due to the fact that it is 1) conceptually simple and 2) not bound to a particular affinity matching function (although this is true for most AIS algorithms). Traditional affinity matching functions employed by the NSA present varying performance in addition to exhibiting holes. Researchers overcome holes by employing a permutation mask to permute the attributes of an antigen before presenting the antigen to a candidate detector. A problem associated with permutation masks is the way in which they are implemented, resulting in researchers misconstruing their value and associated benefits, i.e. a single random permutation mask is applied to a global population of generated detectors.

The viewpoint presented in this paper was to consider a binary problem domain as a set of characteristics/features that need to be learned by the NSA in order to successfully differentiate between self and non-self. By employing this viewpoint, the paper introduced a new detection rule, called the feature-detection rule, which uses a subset of both adjacent and/or non-adjacent features of an antigen in order to determine whether a detector is activated by an antigen.

The paper took the position that multiple relationships, that is, meaningful permutations between features of self/non-self vectors and detectors, exist and that these need to be exploited in order to successfully differentiate between self and non-self data (as opposed to a single randomly generated permutation). In view of the number of possible permutations of features that could exist in a binary data set of length n , the process of determining relevant features is well known and can be approached via a number of means. This paper introduced an elegant manner in which relevant features can be learned by making a simple modification to the NSA, where the selection of relevant features is coupled to the generation of each candidate detector before testing whether the candidate detector is activated by self.

One negative aspect of the feature-detection rule is that it introduces positional bias because it leverages the RCBITS rule, which is overcome by coupling the generation of the feature vector, \mathbf{p} , together with each detector, \mathbf{x} generated by the NSA.

The paper showed that, conceptually, the feature-detection rule is equivalent to the RCHK rule (MHC) if $r = n'$, equivalent to multiple RCHK detectors if $r < n'$, and empirically, the feature-detection rule yields superior performance over the RCHK, RCBITS and HD rule.

The following future work can be undertaken to expand upon the work presented in this paper:

- The NSA was shown to suffer from a severe scaling problem. That is, when P_M , P_f , and n_c are fixed, an exponential increase in n_r can be observed. The feature-detection rule is not immune to this problem, and extensions to the algorithm discussed in section 5.2 can be undertaken to address this issue.
- The applicability of the feature-detection rule can be extended to other AIS paradigms, such as clonal selection theory or network algorithms. Take note, however, that the effect of positional-bias introduced by the feature-detection rule needs to be considered when applying the feature-detection rule to these algorithms.
- The applicability of the feature-detection rule can be extended and investigated within real-valued spaces.
- The execution times associated with the modified NSA algorithm, discussed in section 5.2, can be contrasted with the execution times associated with Forrest's NSA algorithm.
- The feature-detection rule can be extended further, such that rules other than the RCBITS rule are used to ascertain whether a detector is activated by an antigen, in an attempt to explicitly remove the positional bias introduced by the feature-detection rule. For instance, instead of applying the RCBITS rule to a set of features i.e. $f_{RCBITS}(\mathbf{x}, f_{SELECT}(\mathbf{y}, \mathbf{p}))$, other binary rules can be applied and contrasted to the feature-detection rule presented in this paper.
- The selection of relevant features, \mathbf{p} , is vital to the performance of the feature-detection rule. Relevant features were determined in this paper by coupling the generation of \mathbf{p} with a random detector \mathbf{x} , as illustrated in section 5.2. While this approach is valid way to select relevant features, a number of alternative feature selection mechanisms such as principle component analysis should be explored.

References

- [1] AHA, D., P. MURPHY, C. MERZ, E. KEOGH, C. BLAKE and S.HETTICH. 2008. University of California Irvine Repository of Machine Learning Databases. *University of California Irvine*, <http://www.ics.uci.edu/MLRepository.html>.
- [2] AYARA, M., J. TIMMIS, R. DE LEMOS, L.N. DE CASTRO and R. DUNCAN. 2002. Negative Selection: How to Generate Detectors. In *Proceedings of International Conferences on Artificial Immune Systems*, 89-98.
- [3] BALTHROP, J., F. ESPONDA, S. FORREST and M. GLICKMAN. 2002. Coverage and Generalisation in an Artificial Immune System. In *Proceedings of the Genetic and Evolutionary Computation Conference*, 3-10.
- [4] BUGL, P. 2001. THE IMMUNE SYSTEM. *Department of Mathematics, University of Hartford*, [HTTP://UHAWEB.HARTFORD.EDU/BUGL/IMMUNE.HTM](http://UHAWEB.HARTFORD.EDU/BUGL/IMMUNE.HTM).
- [5] COHN, M.. 2004. AN ALTERNATIVE TO CURRENT THINKING ABOUT POSITIVE SELECTION, NEGATIVE SELECTION AND ACTIVATION OF T-CELLS. In *Immunology* **111**, 375-380.
- [6] DHAESELEER, P., S. FORREST and P. HELMAN. 1996. AN IMMUNOLOGICAL APPROACH TO CHANGE DETECTION: ALGORITHMS, ANALYSIS AND IMPLICATIONS. In *Proceedings of the 9th IEEE Symposium on Computer Security and Privacy*, 110-120.
- [7] DASGUPTA, D.. 2006. ADVANCES IN ARTIFICIAL IMMUNE SYSTEMS. In *IEEE Computational Intelligence Magazine* **1 no. 4**, 40-49.
- [8] DE CASTRO, L.N. and J.I. TIMMIS. 2003. *Artificial Immune Systems as a Novel Soft Computing Paradigm*. SPRINGER.
- [9] ESPONDA, F., S. FORREST and P. HELMAN. 2004. A FORMAL FRAMEWORK FOR POSITIVE AND NEGATIVE DETECTION SCHEMES. In *Proceedings of IEEE Transactions on Systems, Man and Cybernetics* **31 no. 1**, 357-373.
- [10] FARMER, J.D., N.H. PACKARD and A.S. PERELSON. 1986. THE IMMUNE SYSTEM, ADAPTATION AND MACHINE LEARNING. *Physica D* **22**, 187-204.
- [11] FORREST, S., .S. PERELSON, L.ALLEN and R. CHERUKURI. 1994. SELF-NONSELF DISCRIMINATION IN A COMPUTER. In *Proceedings of the 1994 IEEE Symposium on Research in Security and Privacy*, 202-212.
- [12] FREITAS, A. and J. TIMMIS. 2003. REVISITING THE FOUNDATIONS OF ARTIFICIAL IMMUNE SYSTEMS: A PROBLEM ORIENTED PERSPECTIVE. *LNCS* **2787**, 229-241.
- [13] GONZÁLEZ, F., D. DASGUPTA and J.GMEZ. 2003. THE EFFECT OF BINARY MATCHING RULES IN NEGATIVE SELECTION. In *Proceedings of Genetic and Evolutionary Computation Conference* **1**, 195-206.
- [14] GRAAF, A.J.. 2004. *Artificial Immune Systems with Evolved Lymphocytes*. Msc THESIS, DEPT. COMPUTER SCIENCE, UNIVERSITY OF PRETORIA, SOUTH AFRICA.
- [15] HOFMEYER, A. and S. FORREST. 2000. Architecture of an Artificial Immune System. In *Evolutionary Computation Journal* **8 no 4**, 443-473.
- [16] HOFMEYER, A.. 1999. *An Immunological Model of Distributed Detection and its Application to Computer Security*. Ph.D. thesis, University of New Mexico
- [17] HOLLAND, J.H., K.J. HOLYOAK, R.E. NISBETT and T. THAGARD. 1986. *Induction: Processes of Inference, Learning and Discovery*. The MIT Press.
- [18] JANEWAY, C. A, P. TRAVERS, M. WALPORT and M. SHLOMCHIK. 2001. *The Immune System in Health and Disease*. Garland Science Publishing.
- [19] JERNE, N.K.. 1974. Towards a Network Theory of the Immune System. In *Annals of Immunology (Inst. Pasteur)* **125C**, 373-389.
- [20] KIM, J. and P. BENTLEY. 2001. An Evaluation of Negative Selection in an Artificial Immune System for Network Intrusion Detection. In *Proceedings of the Genetic and Evolutionary Computation Conference* 1330-1337.
- [21] MERCK INCORPORATED and S.HETTICH. 2008. Immunology and Allergic Disorders. *The Merck Manual of Diagnosis and Therapy*, <http://www.merck.com/mmpe/sec13.html>.
- [22] PERCUS, J.K., O. PERCUS and A.S. PERELSON. 1996. Probability of Self-nonselF Discrimination In *Theoretical and Experimental Insights into Immunology*.
- [23] PERCUS, J.K., O.E. PERCUS and A.S. PERELSON. 1993. Predicting the Size of the Antibody Combining Region from Consideration of Efficient Self-NonSelf Discrimination. In *Journal of Theoretical Biology* **91**, 645-670.
- [24] PERELSON, A.S. and G.F. OSTER. 1979. Theoretical Studies of Clonal Selection: Minimal Antibody Repertoire Size and Reliability of Self-NonSelf Discrimination. In *Journal of Theoretical Biology* **81**, 645-670.
- [25] STIBOR, T., J. TIMMIS and C.ECKERT. 2006. On Permutation Masks in Hamming Negative Selection. In *Proceedings of the Sixth International Conference of Artificial Immune Systems (ICARIS 2006)*, 122-135.
- [26] STIBOR, T., P. MOHR and J. TIMMIS. 2005. Is Negative Selection Appropriate for Anomaly Detection. In *Proceedings of the 2005 Conference on Genetic and Evolutionary Computation*, 321-328.
- [27] USPENSKY, J.V.. 1937. *Introduction to Mathematical Probability*. McGraw-Hill Book Co., NY.
- [28] WIERZCHOŃ, S.T.. 2000. Discriminative Power of the Receptors activated by the k-contiguous Bits Rule. In *Journal of Computer Science and Technology* **1, no 3** 1-13.

Appendix A. Figures

The following appendix comprises graphs portraying the average DR and FR values for each experiment conducted in section 6.

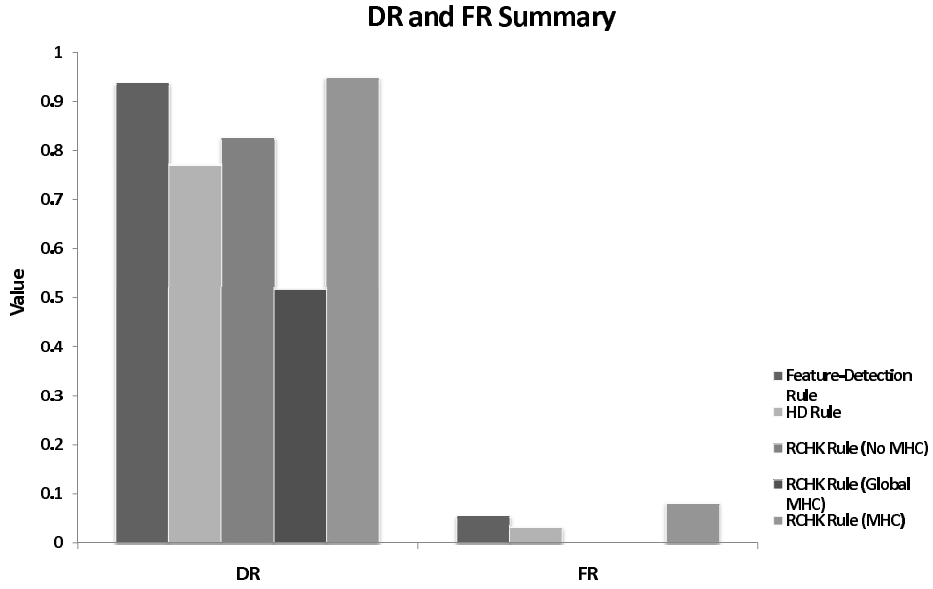


Figure A.3. Car Evaluation Experiment - DR and FR Summary.

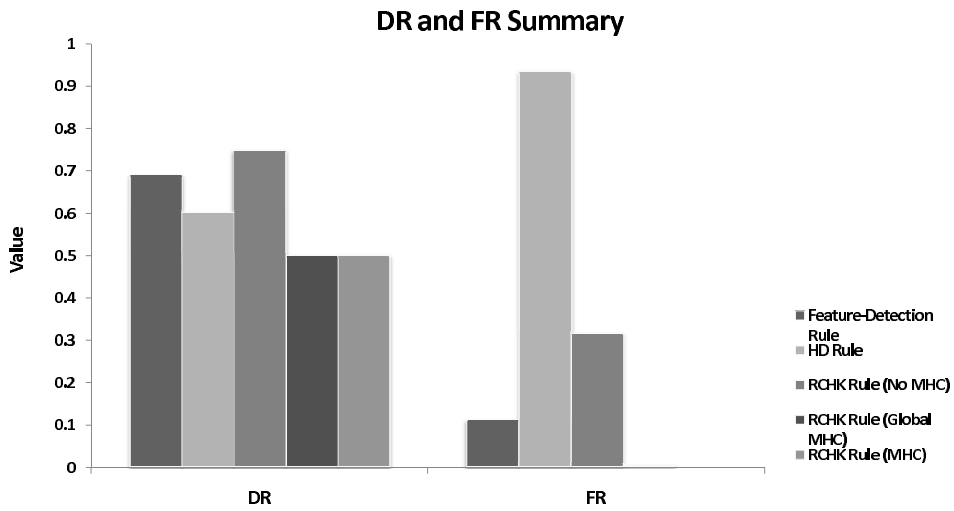


Figure A.4. Iris Experiment - DR and FR Summary

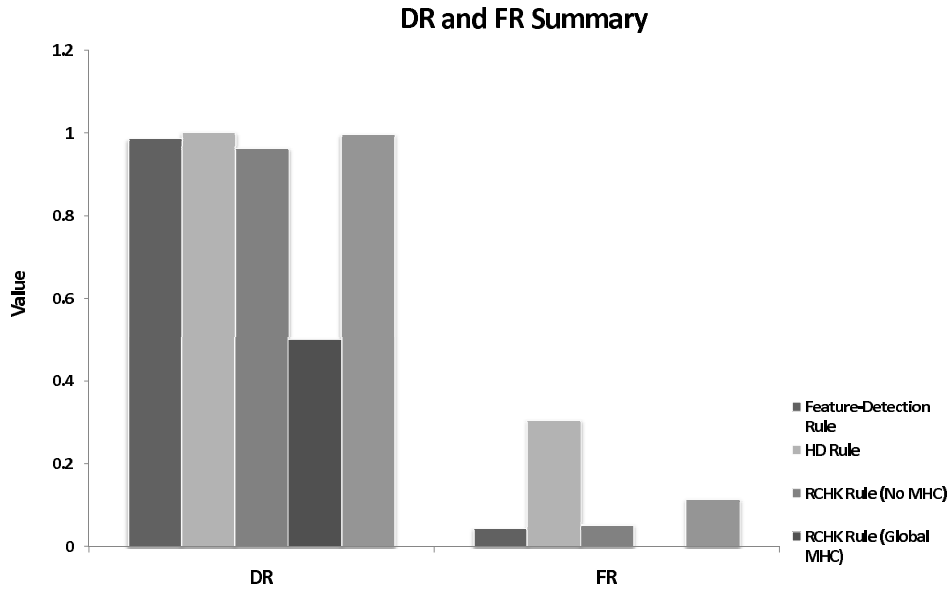


Figure A.5. Wisconsin Breast Cancer Experiment - DR and FR Summary.

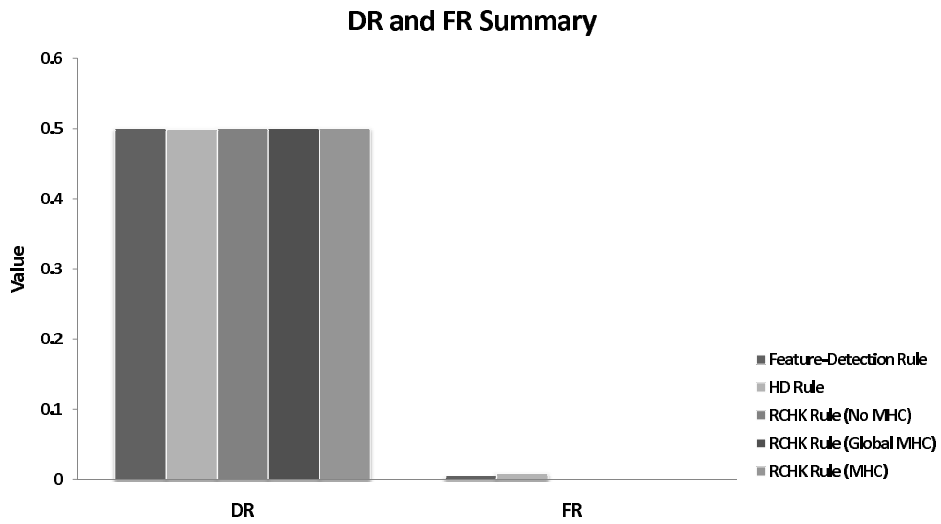


Figure A.6. Glass Experiment - DR and FR Summary.

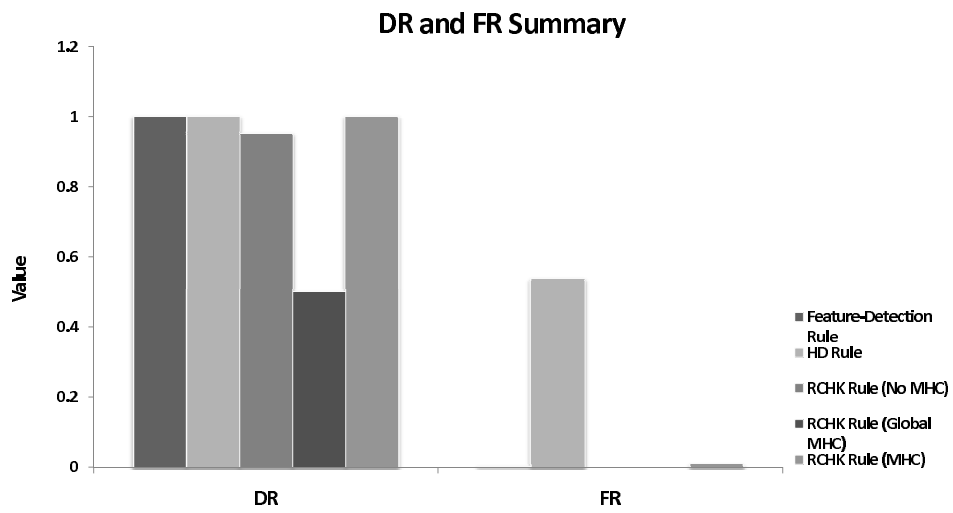


Figure A.7. Mushroom Experiment - DR and FR Summary.

Appendix B. Mann-Whitney Tables

Table B.2. Car Evaluation Experiment - DR and FR Results

Affinity Matching Function	DR	FR	DR - FR	Parameters
Feature Detection Rule	0.9375 ± 0.0119	0.0547 ± 0.0357	0.8836 ± 0.0316	$n' = 6, r = 5$
HD Rule	0.7686 ± 0.0146	0.0313 ± 0.0197	0.7274 ± 0.0225	$r = 9$
RCHK (No MHC) Rule	0.8258 ± 1.2124	0.0 ± 0.0	$0.8258 \pm 5.64E-16$	$r = 3$
RCHK (Global MHC) Rule	0.5155 ± 0.0473	0.0 ± 0.0	0.5155 ± 0.0473	$r = 3$
RCHK (MHC) Rule	0.9482 ± 0.0163	0.0814 ± 0.0491	0.8713 ± 0.0365	$r = 5$

Table B.3. Iris Experiment - DR and FR Table

Affinity Matching Function	DR	FR	DR - FR	Parameters
Feature Detection Rule	0.6896 ± 0.0412	0.1111 ± 0.0931	0.5464 ± 0.1076	$n' = 3, r = 2$
HD Rule	0.6 ± 0.4982	0.9333 ± 0.0631	0.1177 ± 0.5131	$r = 12$
RCHK (No MHC) Rule	0.7476 ± 0.0703	0.3155 ± 0.1594	0.4714 ± 0.1406	$r = 4$
RCHK (Global MHC) Rule	0.4996 ± 0.0052	0.0044 ± 0.0169	0.4330 ± 0.2583	$r = 21$
RCHK (MHC) Rule	0.5002 ± 0.0015	0.0 ± 0.0	0.5002 ± 0.0015	$r = 21$

Table B.4. Cancer Experiment - DR and FR Table

Affinity Matching Function	DR	FR	DR - FR	Parameters
Feature Detection Rule	0.9856 ± 2.5094	0.0433 ± 0.0169	0.9438 ± 0.0159	$n' = 3, r = 2$
HD Rule	1.0 ± 0.0	0.3051 ± 0.0478	0.7672 ± 0.0281	$r = 18$
RCHK (No MHC) Rule	0.9624 ± 0.0198	0.0527 ± 0.0186	0.9109 ± 0.0193	$r = 5$
RCHK (Global MHC) Rule	0.5 ± 0.0	0.0 ± 0.0	0.5 ± 0.0	$r = 37$
RCHK (MHC) Rule	0.9959 ± 0.0068	0.1136 ± 0.0304	0.8942 ± 0.02439	$r = 3$

Table B.5. Glass Experiment - DR and FR Table

Affinity Matching Function	DR	FR	DR - FR	Parameters
Feature Detection Rule	0.4999 ± 0.0047	0.0047 ± 0.0145	0.3999 ± 0.3092	$n' = 23, r = 23$
HD Rule	0.4989 ± 0.0050	0.0095 ± 0.0193	0.3263 ± 0.3671	$r = 40$
RCHK (No MHC) Rule	0.5 ± 0.0	0.0 ± 0.0	0.5 ± 0.0	$r = 30$
RCHK (Global MHC) Rule	0.5 ± 0.0	0.0 ± 0.0	0.5 ± 0.0	$r = 30$
RCHK (MHC) Rule	0.5 ± 0.0	0.0 ± 0.0	0.5 ± 0.0	$r = 30$

Table B.6. Mushroom Experiment - DR and FR Table

Affinity Matching Function	DR	FR	DR - FR	Parameters
Feature Detection Rule	1.0 ± 0.0	0.0037 ± 0.0039	0.9963 ± 0.0039	$n' = 4, r = 3$
HD Rule	1.0 ± 0.0	0.5375 ± 0.0187	0.6504 ± 0.0079	$r = 29$
RCHK (No MHC) Rule	0.9509 ± 0.1497	0.0 ± 0.0	0.9509 ± 0.1497	$r = 4$
RCHK (Global MHC) Rule	0.5000 ± 3.4879	0.0 ± 0.0	0.5000 ± 0.0003	$r = 29$
RCHK (MHC) Rule	1.0 ± 0.0	0.0096 ± 0.0057	0.9904 ± 0.0056	$r = 4$

Table B.7. Car Evaluation Experiment - Mann-Whitney U Test Results

Hypothesis	z	Outcome
$H_0 : \mu_{FDR}^{DR} \neq \mu_{HD}^{DR}$	6.6545	Reject
$H_{0a} : \mu_{FDR}^{DR} \geq \mu_{HD}^{DR}$	6.6545	Accept
$H_1 : \mu_{FDR}^{FR} \neq \mu_{HD}^{FR}$	1.4791	Accept
$H_{1a} : \mu_{FDR}^{FR} < \mu_{HD}^{FR}$	1.4791	Reject
$H_2 : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \neq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	6.6545	Reject
$H_{2a} : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \geq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	6.6545	Accept
$H_3 : \mu_{FDR}^{DR} \neq \mu_{RCHK(NoMHC)}^{DR}$	7.1121	Reject
$H_{3a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(NoMHC)}^{DR}$	7.1121	Accept
$H_4 : \mu_{FDR}^{FR} \neq \mu_{RCHK(NoMHC)}^{FR}$	6.5578	Reject
$H_{4a} : \mu_{FDR}^{FR} < \mu_{RCHK(NoMHC)}^{FR}$	6.5578	Accept
$H_5 : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \neq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	6.6380	Reject
$H_{5a} : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \geq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	6.6380	Accept
$H_6 : \mu_{FDR}^{DR} \neq \mu_{RCHK(GlobalMHC)}^{DR}$	6.9784	Reject
$H_{6a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(GlobalMHC)}^{DR}$	6.9784	Accept
$H_7 : \mu_{FDR}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{FR}$	6.5578	Reject
$H_{7a} : \mu_{FDR}^{FR} < \mu_{RCHK(GlobalMHC)}^{FR}$	6.5578	Accept
$H_8 : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	6.9784	Reject
$H_{8a} : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \geq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	6.9784	Accept
$H_9 : \mu_{FDR}^{DR} \neq \mu_{RCHK(MHC)}^{DR}$	2.4988	Reject
$H_{9a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(MHC)}^{DR}$	2.4988	Accept
$H_{10} : \mu_{FDR}^{FR} \neq \mu_{RCHK(MHC)}^{FR}$	2.5509	Reject
$H_{10a} : \mu_{FDR}^{FR} < \mu_{RCHK(MHC)}^{FR}$	2.5509	Accept
$H_{11} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \neq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	6.6533	Reject
$H_{11a} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \geq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	6.6533	Accept

Table B.8. Iris Experiment - Mann-Whitney U Test Results

Hypothesis	z	Outcome
$H_0 : \mu_{FDR}^{DR} \neq \mu_{HD}^{DR}$	1.3561	Accept
$H_{0a} : \mu_{FDR}^{DR} \geq \mu_{HD}^{DR}$	1.3561	Reject
$H_1 : \mu_{FDR}^{FR} \neq \mu_{HD}^{FR}$	6.7068	Reject
$H_{1a} : \mu_{FDR}^{FR} < \mu_{HD}^{FR}$	6.7068	Accept
$H_2 : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \neq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	3.4875	Reject
$H_{2a} : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \geq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	3.4875	Accept
$H_3 : \mu_{FDR}^{DR} \neq \mu_{RCHK(NoMHC)}^{DR}$	3.6362	Reject
$H_{3a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(NoMHC)}^{DR}$	3.6362	Accept
$H_4 : \mu_{FDR}^{FR} \neq \mu_{RCHK(NoMHC)}^{FR}$	4.4925	Reject
$H_{4a} : \mu_{FDR}^{FR} < \mu_{RCHK(NoMHC)}^{FR}$	4.4925	Accept
$H_5 : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \neq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	2.0203	Reject
$H_{5a} : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \geq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	2.0203	Accept
$H_6 : \mu_{FDR}^{DR} \neq \mu_{RCHK(GlobalMHC)}^{DR}$	6.9170	Reject
$H_{6a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(GlobalMHC)}^{DR}$	6.9170	Accept
$H_7 : \mu_{FDR}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{FR}$	5.3663	Reject
$H_{7a} : \mu_{FDR}^{FR} < \mu_{RCHK(GlobalMHC)}^{FR}$	5.3663	Accept
$H_8 : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	3.2883	Reject
$H_{8a} : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \geq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	3.2883	Accept
$H_9 : \mu_{FDR}^{DR} \neq \mu_{RCHK(MHC)}^{DR}$	7.0732	Reject
$H_{9a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(MHC)}^{DR}$	7.0732	Accept
$H_{10} : \mu_{FDR}^{FR} \neq \mu_{RCHK(MHC)}^{FR}$	6.3810	Reject
$H_{10a} : \mu_{FDR}^{FR} < \mu_{RCHK(MHC)}^{FR}$	6.3810	Accept
$H_{11} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \neq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	7.1197	Reject
$H_{11a} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \geq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	7.1197	Accept

Table B.9. Cancer Experiment - Mann-Whitney U Test Results

Hypothesis	z	Outcome
$H_0 : \mu_{FDR}^{DR} \neq \mu_{HD}^{DR}$	7.1265	Reject
$H_{0a} : \mu_{FDR}^{DR} \geq \mu_{HD}^{DR}$	7.1265	Accept
$H_1 : \mu_{FDR}^{FR} \neq \mu_{HD}^{FR}$	6.6685	Reject
$H_{1a} : \mu_{FDR}^{FR} < \mu_{HD}^{FR}$	6.6685	Accept
$H_2 : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \neq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	6.6685	Reject
$H_{2a} : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \geq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	6.6685	Accept
$H_3 : \mu_{FDR}^{DR} \neq \mu_{RCHK(NoMHC)}^{DR}$	5.6531	Reject
$H_{3a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(NoMHC)}^{DR}$	5.6531	Accept
$H_4 : \mu_{FDR}^{FR} \neq \mu_{RCHK(NoMHC)}^{FR}$	2.4079	Reject
$H_{4a} : \mu_{FDR}^{FR} < \mu_{RCHK(NoMHC)}^{FR}$	2.4079	Accept
$H_5 : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \neq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	5.5641	Reject
$H_{5a} : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \geq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	5.5641	Accept
$H_6 : \mu_{FDR}^{DR} \neq \mu_{RCHK(GlobalMHC)}^{DR}$	7.1265	Reject
$H_{6a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(GlobalMHC)}^{DR}$	7.1265	Accept
$H_7 : \mu_{FDR}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{FR}$	7.1265	Reject
$H_{7a} : \mu_{FDR}^{FR} < \mu_{RCHK(GlobalMHC)}^{FR}$	7.1265	Accept
$H_8 : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	7.1265	Reject
$H_{8a} : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \geq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	7.1265	Accept
$H_9 : \mu_{FDR}^{DR} \neq \mu_{RCHK(MHC)}^{DR}$	3.3505	Reject
$H_{9a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(MHC)}^{DR}$	3.3505	Accept
$H_{10} : \mu_{FDR}^{FR} \neq \mu_{RCHK(MHC)}^{FR}$	6.4245	Reject
$H_{10a} : \mu_{FDR}^{FR} < \mu_{RCHK(MHC)}^{FR}$	6.4245	Accept
$H_{11} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \neq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	6.6674	Reject
$H_{11a} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \geq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	6.6674	Accept

Table B.10. Glass Experiment - Mann-Whitney U Test Results

Hypothesis	z	Outcome
$H_0 : \mu_{FDR}^{DR} \neq \mu_{HD}^{DR}$	1.0797	Accept
$H_{0a} : \mu_{FDR}^{DR} \geq \mu_{HD}^{DR}$	1.0797	Reject
$H_1 : \mu_{FDR}^{FR} \neq \mu_{HD}^{FR}$	1.0015	Accept
$H_{1a} : \mu_{FDR}^{FR} < \mu_{HD}^{FR}$	1.0015	Reject
$H_2 : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \neq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	1.0797	Accept
$H_{2a} : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \geq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	1.0797	Reject
$H_3 : \mu_{FDR}^{DR} \neq \mu_{RCHK(NoMHC)}^{DR}$	1.0724	Accept
$H_{3a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(NoMHC)}^{DR}$	1.0724	Reject
$H_4 : \mu_{FDR}^{FR} \neq \mu_{RCHK(NoMHC)}^{FR}$	1.7621	Reject
$H_{4a} : \mu_{FDR}^{FR} < \mu_{RCHK(NoMHC)}^{FR}$	1.7621	Accept
$H_5 : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \neq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	1.0724	Accept
$H_{5a} : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \geq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	1.0724	Reject
$H_6 : \mu_{FDR}^{DR} \neq \mu_{RCHK(GlobalMHC)}^{DR}$	1.0724	Accept
$H_{6a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(GlobalMHC)}^{DR}$	1.0724	Reject
$H_7 : \mu_{FDR}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{FR}$	1.7621	Reject
$H_{7a} : \mu_{FDR}^{FR} < \mu_{RCHK(GlobalMHC)}^{FR}$	1.7621	Accept
$H_8 : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	1.0724	Accept
$H_{8a} : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \geq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	1.0724	Reject
$H_9 : \mu_{FDR}^{DR} \neq \mu_{RCHK(MHC)}^{DR}$	1.0724	Accept
$H_{9a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(MHC)}^{DR}$	1.0724	Reject
$H_{10} : \mu_{FDR}^{FR} \neq \mu_{RCHK(MHC)}^{FR}$	1.7621	Reject
$H_{10a} : \mu_{FDR}^{FR} < \mu_{RCHK(MHC)}^{FR}$	1.7621	Accept
$H_{11} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \neq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	5.8377	Reject
$H_{11a} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \geq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	5.8377	Accept

Table B.11. Mushroom Experiment - Mann-Whitney U Test Results

Hypothesis	z	Outcome
$H_0 : \mu_{FDR}^{DR} \neq \mu_{HD}^{DR}$	0	Accept
$H_{0a} : \mu_{FDR}^{DR} \geq \mu_{HD}^{DR}$	0	Reject
$H_1 : \mu_{FDR}^{FR} \neq \mu_{HD}^{FR}$	6.6664	Reject
$H_{1a} : \mu_{FDR}^{FR} < \mu_{HD}^{FR}$	6.6664	Accept
$H_2 : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \neq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	6.6529	Reject
$H_{2a} : \mu_{FDR}^{DR} - \mu_{HD}^{FR} \geq \mu_{HD}^{DR} - \mu_{HD}^{FR}$	6.6529	Accept
$H_3 : \mu_{FDR}^{DR} \neq \mu_{RCHK(NoMHC)}^{DR}$	1.7621	Reject
$H_{3a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(NoMHC)}^{DR}$	1.7621	Accept
$H_4 : \mu_{FDR}^{FR} \neq \mu_{RCHK(NoMHC)}^{FR}$	5.4698	Reject
$H_{4a} : \mu_{FDR}^{FR} < \mu_{RCHK(NoMHC)}^{FR}$	5.4698	Accept
$H_5 : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \neq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	3.9838	Reject
$H_{5a} : \mu_{FDR}^{DR} - \mu_{RCHK(NoMHC)}^{FR} \geq \mu_{RCHK(NoMHC)}^{DR} - \mu_{RCHK(NoMHC)}^{FR}$	3.9838	Accept
$H_6 : \mu_{FDR}^{DR} \neq \mu_{RCHK(GlobalMHC)}^{DR}$	7.5134	Reject
$H_{6a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(GlobalMHC)}^{DR}$	7.5134	Accept
$H_7 : \mu_{FDR}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{FR}$	5.4698	Reject
$H_{7a} : \mu_{FDR}^{FR} < \mu_{RCHK(GlobalMHC)}^{FR}$	5.4698	Accept
$H_8 : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \neq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	6.9932	Reject
$H_{8a} : \mu_{FDR}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR} \geq \mu_{RCHK(GlobalMHC)}^{DR} - \mu_{RCHK(GlobalMHC)}^{FR}$	6.9932	Accept
$H_9 : \mu_{FDR}^{DR} \neq \mu_{RCHK(MHC)}^{DR}$	0	Accept
$H_{9a} : \mu_{FDR}^{DR} \geq \mu_{RCHK(MHC)}^{DR}$	0	Reject
$H_{10} : \mu_{FDR}^{FR} \neq \mu_{RCHK(MHC)}^{FR}$	4.1972	Reject
$H_{10a} : \mu_{FDR}^{FR} < \mu_{RCHK(MHC)}^{FR}$	4.1972	Accept
$H_{11} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \neq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	6.6682	Reject
$H_{11a} : \mu_{FDR}^{DR} - \mu_{RCHK(MHC)}^{FR} \geq \mu_{RCHK(MHC)}^{DR} - \mu_{RCHK(MHC)}^{FR}$	6.6682	Accept