Technical Report CGR 05-10: Correlation and Classification of Internet Traffic Anomalies

Patrick Macnamara

Graduate Student Mentor:
Olivier Contant

Faculty Advisors:
Stéphane Lafortune
Demosthenis Teneketzis

Department of Electrical Engineering and Computer Science
University of Michigan

August 5, 2005

1. Introduction

In Chapter 4 of his doctoral thesis,[1] Olivier Contant presents a hierarchical approach to monitor, classify, correlate and assess a large number of alerts generated at spatially distributed sites on the Internet. Each site produces an alert whenever a network anomaly – and thus potentially an unknown threat – is detected. With this approach, the network is decomposed into levels and each node is monitored for particular attack profiles, such as a worm or a distributed denial of service (DDoS). At each node in the hierarchy, a risk index is assigned for each type of attack.

Contant uses as a reference the topography of the Internet2 backbone network, Abilene, which is composed of eleven routers distributed across the United States. Figure 2 represents the structure of the Abilene network. The level 0 nodes correspond to the eleven Abilene routers. The level 0 nodes are grouped together in three regional nodes at level 1. Level 2 consists of one node, which represents the entire network. Thus, each risk index generated for a particular node assesses the threat for that level. For example, a high risk index in the level 2 node would represent a threat affecting the entire network.

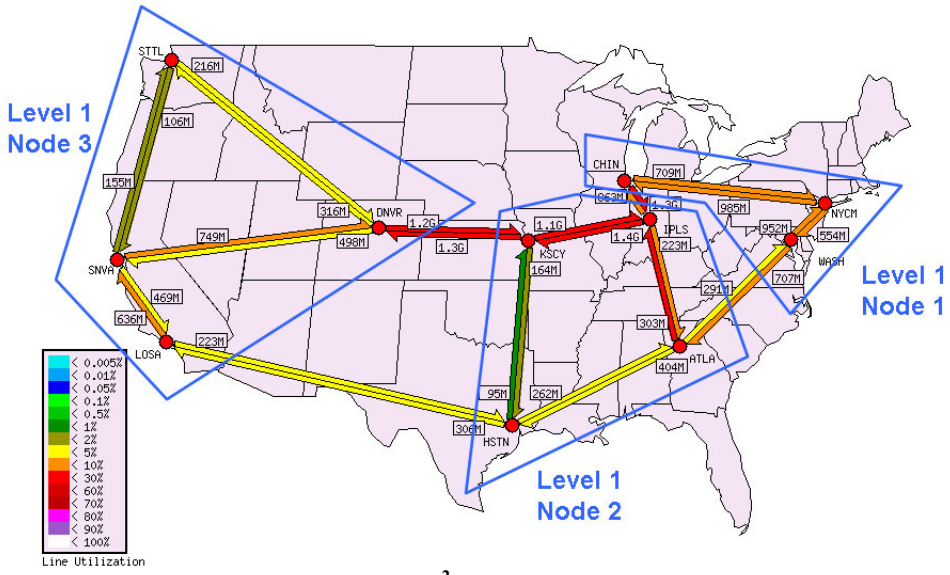

Figure 1: Abilene Internet2 Backbone Network[2] (Note that the formulation of the level 1 nodes differ from those in Contant's thesis.)

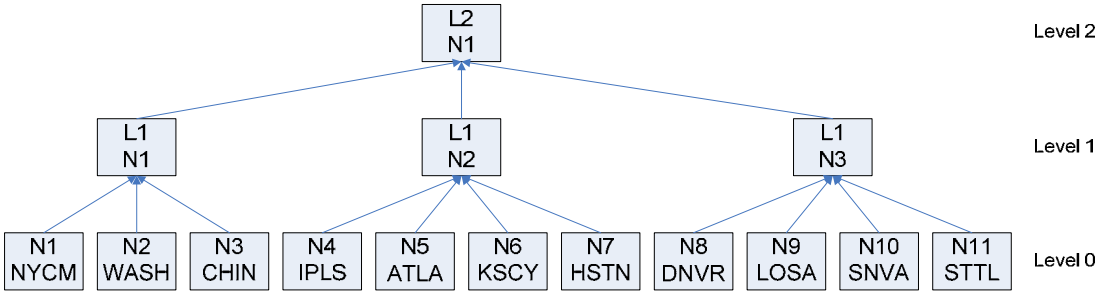

Figure 2: Hierarchical Structure of the Abilene Network

[1] Olivier Contant, "On Monitoring and Diagnosing Classes of Discrete Event Systems"
[2] Source: http://Abilene.internet2.edu/

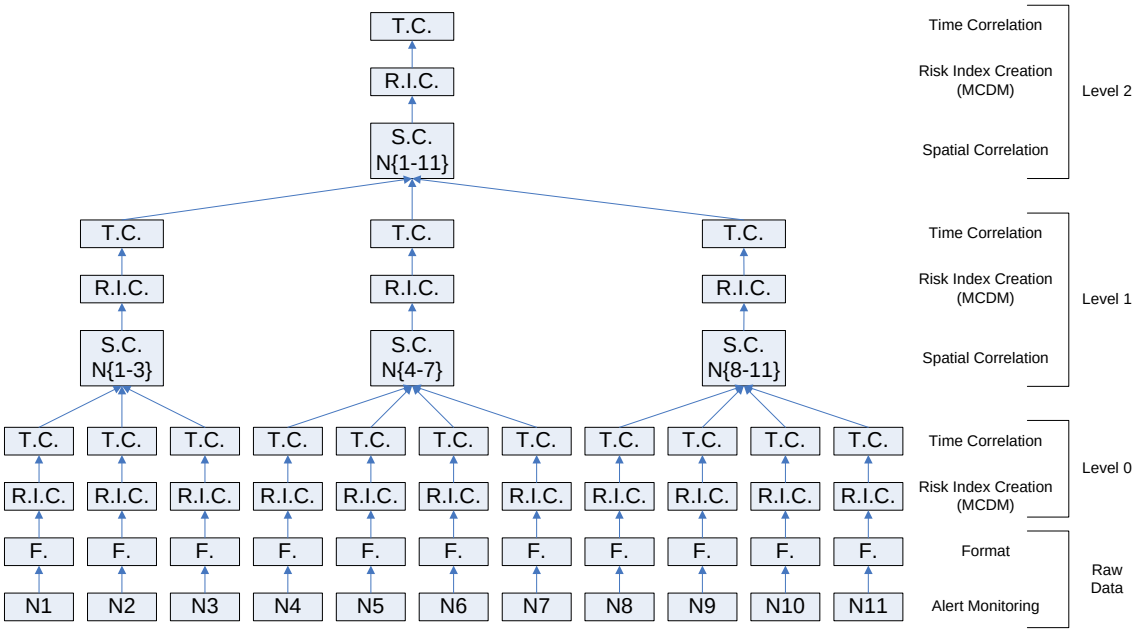

Figure 3: Components of Hierarchical Structure

In Figure 3, the components of each level are shown. The raw data level is the lowest level and it is responsible for formatting alerts generated by the anomaly-based intrusion detection systems (IDSs) into inputs for level 0. When an alert is triggered by the third-party IDS, it is communicated to the lowest level of the hierarchy. In the raw data level, the appropriate information is extracted from the alerts and the netflow data to compute the inputs needed at level 0. At level 0, a risk index is generated using the Multi-Criteria Decision Making (MCDM) approach. This risk index then passes through the time correlation unit, which outputs a new risk index based on the current and previous risk indices output by the MCDM module. In the higher levels, each node uses the MCDM approach to calculate a new risk index, taking into account a spatially correlated index for its child nodes. This new risk index is then time-correlated.

2. Overview of Multi-Criteria Decision Making Tool

In each node, a MCDM approach is used to compute the risk index for each attack profile. Using the MCDM method, numerous and often conflicting criteria can be used to classify a particular attack into a predefined categories. The criteria that the MCDM tool uses to calculate a risk index are often related to each other in a complex way and may sometimes conflict. The goal of MCDM is to locate those potential conflicts and determine a solution.

2.1. Electre Tri Method

The specific multi-criteria problem method used by the tool is Electre Tri, which uses an outranking relation to assess the degree by which an alternative a outranks a profile b_h. Using the outranking relation, one can conclude that a outranks b_h if enough criteria confirm that a is at least as good as b_h (concordance), while no criteria are opposed to that in a "too strong way" (discordance).

In describing the Electre Tri method, we will use the following notation. Let $F = \{1,2,\ldots,m\}$ be the set of indices of criteria $g_1,g_2,\ldots,g_m$. Let $B = \{1,2,\ldots,p\}$ be the set of indices of profiles $b_1,b_2,\ldots,b_p$. These profiles delimit the categories $C_1,C_2,\ldots,C_p,C_{p+1}$, where b_h corresponds to the upper bound of category C_h and lower bound of category C_{h+1}.

For example, using the notation just described, Figure 4 shows how alternatives a_1 and a_2 compare to the five profiles on the three criteria. Clearly, the alternative a_1 can be placed into the category C_2. However, the alternative a_2 cannot be classified so easily. That is where the MCDM approach comes in.

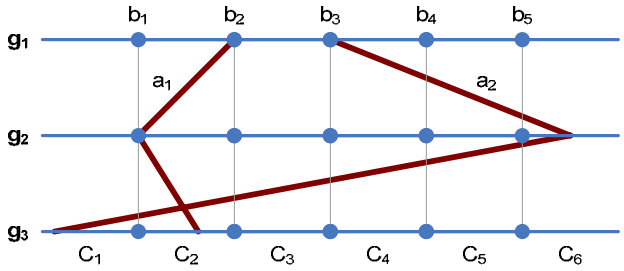

Figure 4: Assignment of Alternatives to Categories

In order to assign alternatives into categories, an outranking relation S is constructed. The relation S confirms or not the statement that "a is at least as good as b_h" or "a outranks b_h" (aSb_h). The indifference, preference, and veto thresholds, $q_j(b_h)$, $p_j(b_h)$, and $v_j(b_h)$ respectively, give some flexibility in how the values of an alternative on the criteria are compared to the profiles. Below are some key definitions:

- The function $g_j(a)$ represents the value of a on criterion g_j.
- The indifference threshold $q_j(b_h)$ is the largest difference $g_j(a) - g_j(b_h)$ that preserves "indifference" between a and b_h on criterion g_j.
- The preference threshold $p_j(b_h)$ is the smallest difference $g_j(a) - g_j(b_h)$ that is compatible with a preference in favor of a on criterion g_j.
- The veto threshold $v_j(b_h)$ is the smallest difference $g_j(b_h) - g_j(a)$ that is incompatible with the assertion aSb_h on criterion g_j.

The outranking relation S is constructed by following these steps:

1. Compute the partial concordance indices $c_j(a,b_h)$ and $c_j(b_h,a)$. The partial concordance index $c_j(a,b_h)$ assesses the statement "a outranks b_h" with respect to the criterion g_j. The same logic holds for $c_j(b_h,a)$.
2. Compute the global concordance indices $c(a,b_h)$ and $c(b_h,a)$. The global concordance index $c(a,b_h)$ assesses the statement "a outranks b_h" with respect to all criteria. The same logic holds for $c(b_h,a)$.
3. Compute the partial discordance indices $d_j(a,b_h)$ and $d_j(b_h,a)$. The partial discordance index $d_j(a,b_h)$ assesses the degree of opposition to the statement "a outranks b_h" with respect to the criterion g_j. The same logic holds for $d_j(b_h,a)$.

4. Compute the "fuzzy" outranking relation based on the credibility indices $\sigma(a,b_h)$ and $\sigma(b_h,a)$. Note that if there is a criterion g_j for which $d_j(a,b_h)$ is high, then $\sigma(a,b_h)$ will be very small. And, if $d_j(a,b_h) = 1$ for any criterion, then it follows that $\sigma(a,b_h)$ will equal zero. Thus, it does not matter how small the corresponding weight coefficient is, an index $d_j(a,b_h)$ close to 1 will cause $\sigma(a,b_h)$ to be very small. The same logic holds for $\sigma(b_h,a)$.
5. Determine a λ-cut of the "fuzzy" relation in order to obtain a "crisp" outranking relation. The cutting level λ is the minimum value the credibility degree must take in order to satisfy the statement aSb_h. Thus, we have
$$\sigma(a,b_h) \geq \lambda \Rightarrow aSb_h$$

The next step is to determine for each profile which of the following binary relations is satisfied:

- **Indifference [*I*]:**
  $aIb_h \Leftrightarrow aSb_h \wedge b_hSa$, or equivalently,
  $aIb_h \Leftrightarrow (\sigma(a,b_h) \geq \lambda) \wedge (\sigma(b_h,a) \geq \lambda)$
- **Preference [$\succ$]:**
  $a \succ b_h \Leftrightarrow aSb_h \wedge \neg b_hSa$
  $a \prec b_h \Leftrightarrow \neg aSb_h \wedge b_hSa$
- **Incomparability [*R*]:**
  $aRb_h \Leftrightarrow \neg aSb_h \wedge \neg b_hSa$

The final step in the MCDM approach is to assign the alternative *a* to a category through one of the two following assignment procedures.

Pessimistic Assignment Procedure

1. Compare *a* successively to b_i, for $i = p, p-1,\ldots, 0$.
2. Let b_h be the first profile such that $a \succ b_h$ and assign *a* to C_{h+1}.

Optimistic Assignment Procedure

1. Compare *a* successively to b_i, for $i = 1, 2,\ldots, p$.
2. Let b_h be the first profile such that $b_h \succ a$ and assign *a* to C_h.

Note, that in the following experiments, the pessimistic assignment procedure is used.

3. Parameters for Multi-Criteria Decision Making Tool

In his thesis, Olivier Contant presented a hierarchical approach to classify network anomalies detected by third-party IDSs. Using this approach, 20 known services and 1 unknown service could be monitored for different types of attacks. The objective was to give a time-correlated risk index for each type of attack at each node for the 21 services. However, for the sake of simplicity, only one service and only DDoS attacks are

considered in this report. The following is a list of the parameters used in the MCDM tool to monitor that one service for DDoS attacks. In addition, we do not claim that the following parameters are the best possible ones, but we do claim that they perform well under the conditions we have examined.

3.1. Level 0 Parameters

3.1.1. Criteria and Profiles

At each time step, the raw data level receives alerts generated by anomaly-based IDSs. It then computes the values of the DDoS criteria that are inputs to the MCDM tool at level 0. The following criteria are used to detect a DDoS anomaly:

1. Traffic increase percentage (as given by the anomaly-based IDS)
2. Total number of unique source IP addresses
3. Number of unique source IP addresses transmitting to 1 unique destination IP addresses

The profiles for the three criteria are shown below.

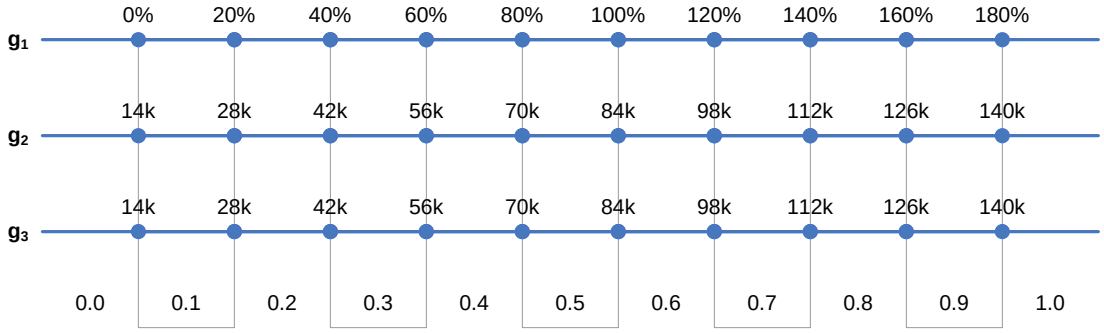

Figure 5: Level 0 Profiles

In addition, the weight coefficients were all set to one. Since all three criteria must increase in order for the anomaly to be considered a DDoS, all three criteria were weighted equally.

k_1	k_2	k_3
1	1	1

Table 1: Level 0 Weight Coefficients

3.1.2. Indifference, Preference, and Veto Thresholds

The indifference, preference, and veto thresholds directly affect the partial discordance and global discordance indices. In addition, when setting the thresholds, the following should hold:

$$v_j(b_h) \geq p_j(b_h) \geq q_j(b_h)$$

With nonzero thresholds, the partial concordance and partial discordance indices will be set as shown below. The value of $g_j(a)$ determines the value of the indices for the profile b_h on the criterion g_j. For example, if $g_j(a) = g_j(b_h)$, then according to the graph below, both partial concordance indices are 1 and both partial discordance indices are 0. In other words, a is at least as good as b_h and b_h is at least as good as a on criteria g_j.

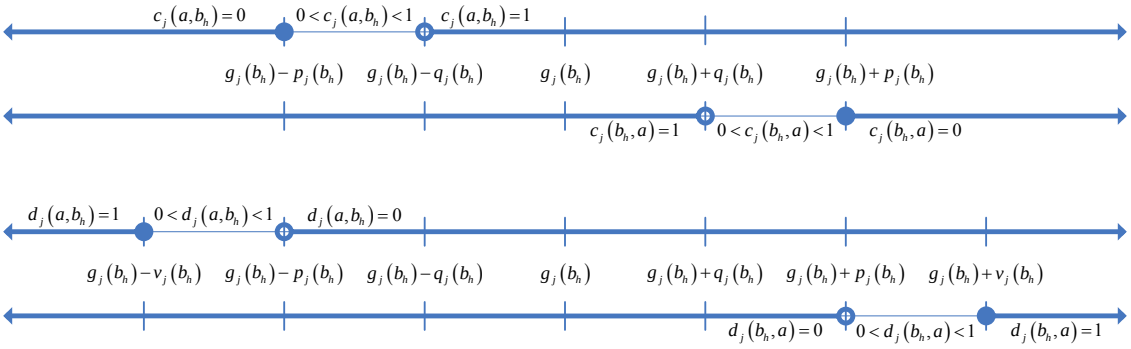

Figure 6: Effect of Thresholds on Partial Concordance and Partial Discordance Indices

If all the thresholds were set to zero, then for the following values of $g_j(a)$, we would have:

- $g_j(a) > g_j(b_h) \rightarrow c_j(a,b_h) = 1$, $c_j(b_h,a) = 0$, $d_j(a,b_h) = 0$, $d_j(b_h,a) = 1$
  (in other words, a outranks b_h on criterion g_j)
- $g_j(a) < g_j(b_h) \rightarrow c_j(a,b_h) = 0$, $c_j(b_h,a) = 1$, $d_j(a,b_h) = 1$, $d_j(b_h,a) = 0$
  (in other words, b_h outranks a on criterion g_j)
- $g_j(a) = g_j(b_h) \rightarrow c_j(a,b_h) = 0$, $c_j(b_h,a) = 0$, $d_j(a,b_h) = 1$, $d_j(b_h,a) = 1$
  (in other words, a does not outrank b_h and b_h does not outrank a on criterion g_j)

Therefore, if the thresholds are all set to zero, then the MCDM tool will be extremely strict. In addition, note that if $g_j(a) < g_j(b_h)$, then $d_j(a,b_h) = 1$. As mentioned earlier, if $d_j(a,b_h) = 1$, then it follows that $\sigma(a,b_h) = 0$. Thus, $\sigma(a,b_h)$ will equal 0 for all the profiles larger than $g_j(a)$. Thus it follows that the smallest criterion will determine the rank. For example, in Figure 4, if the veto threshold is zero, the value of $g_3(a_2)$ would force $\sigma(a,b_h)$ to equal zero for all h.

With this in mind, the level 0 thresholds were set so that all three criteria had to be large in order for the MCDM tool to generate a high risk index. Thus, the indifference thresholds were set to be a quarter of the difference between two adjacent profiles. The preference thresholds were set to be a half of that difference, and the veto thresholds were set to be three-quarters of that difference. Therefore, we have the following thresholds.

indifference thresholds				preference thresholds				veto thresholds			
	g_1	g_2	g_3		g_1	g_2	g_3		g_1	g_2	g_3
b_1	5	3500	3500	b_1	10	7000	7000	b_1	15	10500	10500
b_2	5	3500	3500	b_2	10	7000	7000	b_2	15	10500	10500
b_3	5	3500	3500	b_3	10	7000	7000	b_3	15	10500	10500
b_4	5	3500	3500	b_4	10	7000	7000	b_4	15	10500	10500
b_5	5	3500	3500	b_5	10	7000	7000	b_5	15	10500	10500
b_6	5	3500	3500	b_6	10	7000	7000	b_6	15	10500	10500
b_7	5	3500	3500	b_7	10	7000	7000	b_7	15	10500	10500
b_8	5	3500	3500	b_8	10	7000	7000	b_8	15	10500	10500
b_9	5	3500	3500	b_9	10	7000	7000	b_9	15	10500	10500
b_{10}	5	3500	3500	b_{10}	10	7000	7000	b_{10}	15	10500	10500

Table 2: Level 0 Indifference, Preference, and Veto Thresholds

3.1.3. Cutting Level λ

The cutting level for level 0 was set at 0.66. This may not be the best cutting level, but it did produce good results.

3.1.4. Time Correlation

The risk indices output by the MCDM tool are time-correlated and then normalized. In level 0, the number of time steps considered in the correlation, T_0, is 3 and the time correlation weight, ω, is 1.2.

3.2. Level 1-2 Parameters

3.2.1. Criteria and Profiles

At each time step, the upper level nodes received all the time correlated risk indices of their child nodes. Then, it calculated the spatially correlated risk index. The following criteria were used for levels 1 and 2.

1. Normalized spatial correlation
2. Percentage of child nodes with risk greater than 0.15
3. Percentage of child nodes with risk greater than 0.30
4. Percentage of child nodes with risk greater than 0.45
5. Percentage of child nodes with risk greater than 0.60
6. Percentage of child nodes with risk greater than 0.75
7. Percentage of child nodes with risk greater than 0.90

Below are the profiles for the seven criteria.

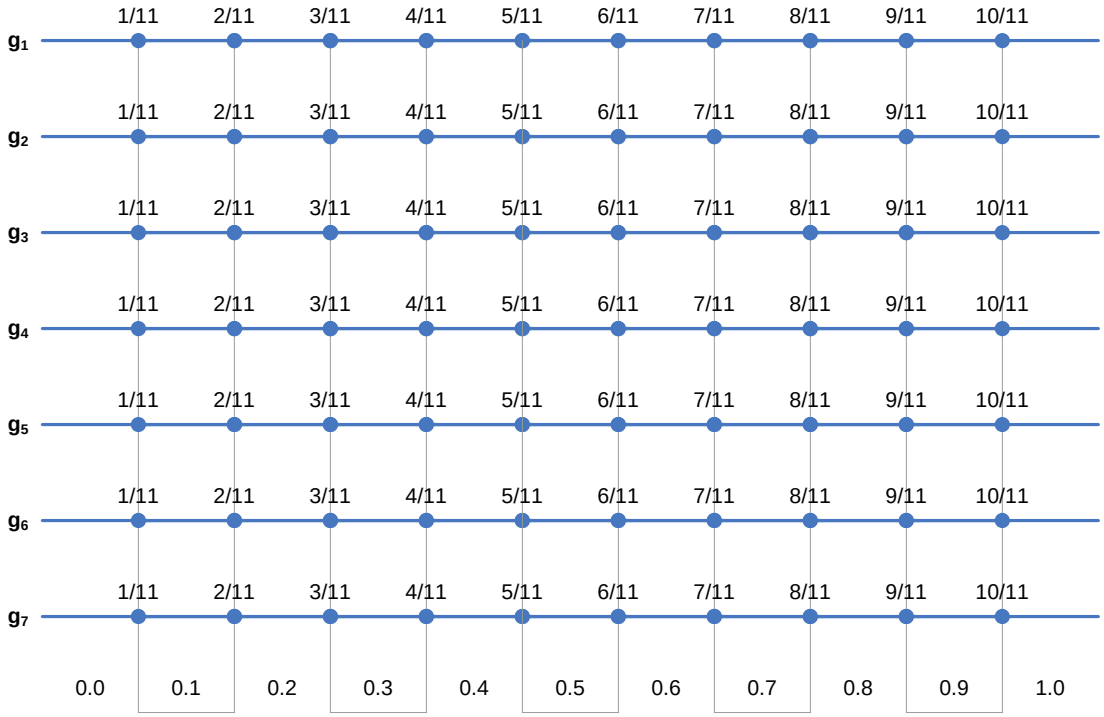

Figure 7: Level 1-2 Profiles

Since criteria 2-7 are all measuring the distribution of risk in the child nodes, the weights for these criteria were reduced relative to criterion 1, the normalized spatial correlation.

k_1	k_2	k_3	k_4	k_5	k_6	k_7
1	1/6	1/6	1/6	1/6	1/6	1/6

Table 3: Level 1-2 Weight Coefficients

3.2.2. Indifference, Preference, and Veto Thresholds

In levels 1 and 2, the veto thresholds were set to be higher than in level 0 because it was much more possible for the upper level criteria to conflict with each other. For example, there may be no nodes with risk greater than 0.90, but all of them may have a risk greater than 0.75. If the veto thresholds are set too low, then the low value on criterion 7 would "veto" the risk index, lowering the risk index too far. Therefore, the following thresholds were set for levels 1 and 2.

indifference thresholds

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
b_1	1/22	1/22	1/22	1/22	1/22	1/22	1/22
b_2	1/11	1/11	1/11	1/11	1/11	1/11	1/11
b_3	3/22	3/22	3/22	3/22	3/22	3/22	3/22
b_4	2/11	2/11	2/11	2/11	2/11	2/11	2/11
b_5	5/22	5/22	5/22	5/22	5/22	5/22	5/22
b_6	3/11	3/11	3/11	3/11	3/11	3/11	3/11
b_7	7/22	7/22	7/22	7/22	7/22	7/22	7/22
b_8	4/11	4/11	4/11	4/11	4/11	4/11	4/11
b_9	9/22	9/22	9/22	9/22	9/22	9/22	9/22
b_{10}	5/11	5/11	5/11	5/11	5/11	5/11	5/11

Table 4: Level 1-2 Indifference Thresholds

preference thresholds

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
b_1	1/44	1/44	1/44	1/44	1/44	1/44	1/44
b_2	1/22	1/22	1/22	1/22	1/22	1/22	1/22
b_3	3/44	3/44	3/44	3/44	3/44	3/44	3/44
b_4	1/11	1/11	1/11	1/11	1/11	1/11	1/11
b_5	5/44	5/44	5/44	5/44	5/44	5/44	5/44
b_6	3/22	3/22	3/22	3/22	3/22	3/22	3/22
b_7	7/44	7/44	7/44	7/44	7/44	7/44	7/44
b_8	2/11	2/11	2/11	2/11	2/11	2/11	2/11
b_9	9/44	9/44	9/44	9/44	9/44	9/44	9/44
b_{10}	5/22	5/22	5/22	5/22	5/22	5/22	5/22

Table 5: Level 1-2 Preference Thresholds

veto thresholds

	g_1	g_2	g_3	g_4	g_5	g_6	g_7
b_1	2	2	2	2	2	2	2
b_2	2	2	2	2	2	2	2
b_3	2	2	2	2	2	2	2
b_4	2	2	2	2	2	2	2
b_5	2	2	2	2	2	2	2
b_6	2	2	2	2	2	2	2
b_7	2	2	2	2	2	2	2
b_8	2	2	2	2	2	2	2
b_9	2	2	2	2	2	2	2
b_{10}	2	2	2	2	2	2	2

Table 6: Level 1-2 Veto Thresholds

3.2.3. Cutting Level λ

The cutting level for levels 1 and 2 was set to 0.50.

3.2.4. Time and Spatial Correlation

For both levels 1 and 2, the time correlation weight, ω, was 1.2 and the spatial correlation weight, ρ, was 1.2. The number of steps used in the time correlation for level 1 was 4, and for level 2 it was 5. Therefore, $T_1 = 4$ and $T_2 = 5$;

4. Experimental Results and Analysis

Several experiments were performed to do a "proof-of-concept" demonstration of the MCDM approach and also to perform a sensitivity analysis of the tool. One set of experiments used actual historical data containing a DDoS attack. The other set of experiments used arbitrary criteria for the level 0 input to test the sensitivity of the tool.

4.1. Historical Distributed Denial of Service Attack

On Jan 16, 2005, a DDoS anomaly occurred on the Abilene Network, lasting the entire day. The number of packets transmitting to the IP address 128.109.64.0 surged dramatically. All of the new traffic consisted of 48 byte packets being transmitted from either port 1024 or port 3072 to the destination port 80 (web http traffic). The anomaly was the most severe at the NYCM, WASH, and CHIN routers. The anomaly appeared at most, but not all, of the remaining Abilene routers. However, the anomaly at these routers was much more short-lived, lasting only 15 minutes.

Since the anomaly was mainly confined to three routers, we only used the DDoS traffic from the NYCM router and "spread" it to the remaining 10 Abilene routers. To reflect real world conditions, we assumed that the eleven routers did not have exactly the same infection rate. Therefore, when traffic was spread to different routers, only a certain percentage of the DDoS traffic was passed along. Below are the percentages of the anomalous traffic that was "spread" to each router. With these percentages, the attack is strongest in the east and weakest in the west.

EAST	**NYCM N1**	100%
	WASH N2	90%
	CHIN N3	110%
MIDWEST	**IPLS N4**	90%
	ATLA N5	80%
	KSCY N6	70%
	HSTN N7	100%
WEST	**DNVR N8**	50%
	LOSA N9	30%
	SNVA N10	70%
	STTL N11	80%

Table 7: Attack Levels

We considered a three day period from January 15, 2005 through January 17, 2005, divided into five minute bins. However, the first criterion is the percentage increase in

traffic. This is a value that should be retrieved from the third-party IDS generating the alert. In order to simplify the analysis, we assumed that the anomaly-based IDS produced an alert at every possible time step. Thus, an alert was generated for every five minute interval. In addition, we used the three day period from January 8, 2005 through January 10, 2005 as the "normal" traffic. Thus, the first criterion was calculated as the percentage increase in packets from one time step in the "normal" traffic to the corresponding time a week later.

Thus, below are plots of the criteria generated for all the routers, where the horizontal lines are the maximum profiles.

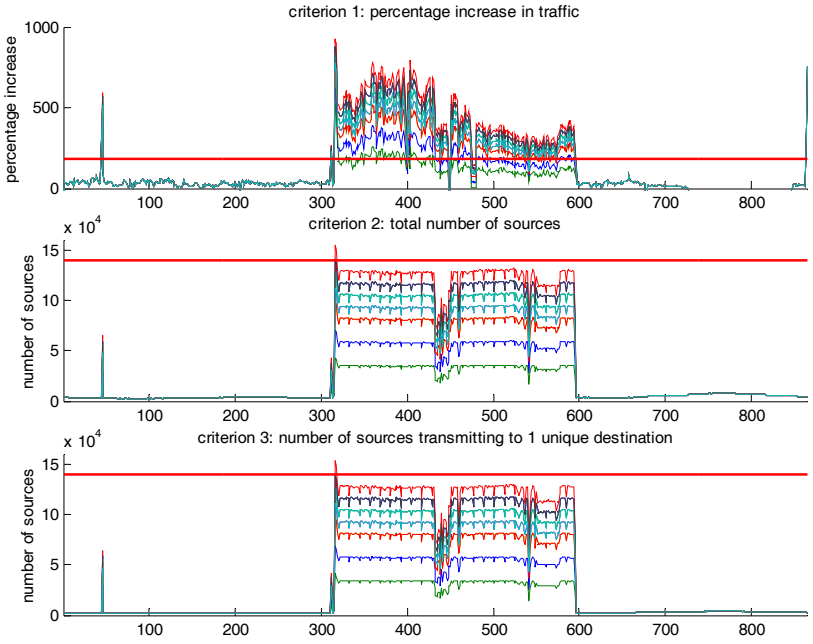

Figure 8: Input Criteria for All Routers

The risk indices of each level can be divided into five attack alert stages. These stages are below. The ranges are the same as the ranges Olivier Contant used for his worm attack alert stages in his thesis.

Alert Stages	Low	Guarded	Elevated	High	Severe
Level 0 Risk Index Ranges	1-14%	15-29%	30-49%	50-79%	80-100%
Level 1 Risk Index Ranges	1-11%	12-24%	25-39%	40-59%	60-100%
Level 2 Risk Index Ranges	1-9%	10-19%	20-29%	30-39%	40-100%

Table 8: DDoS Attack Alert Stages

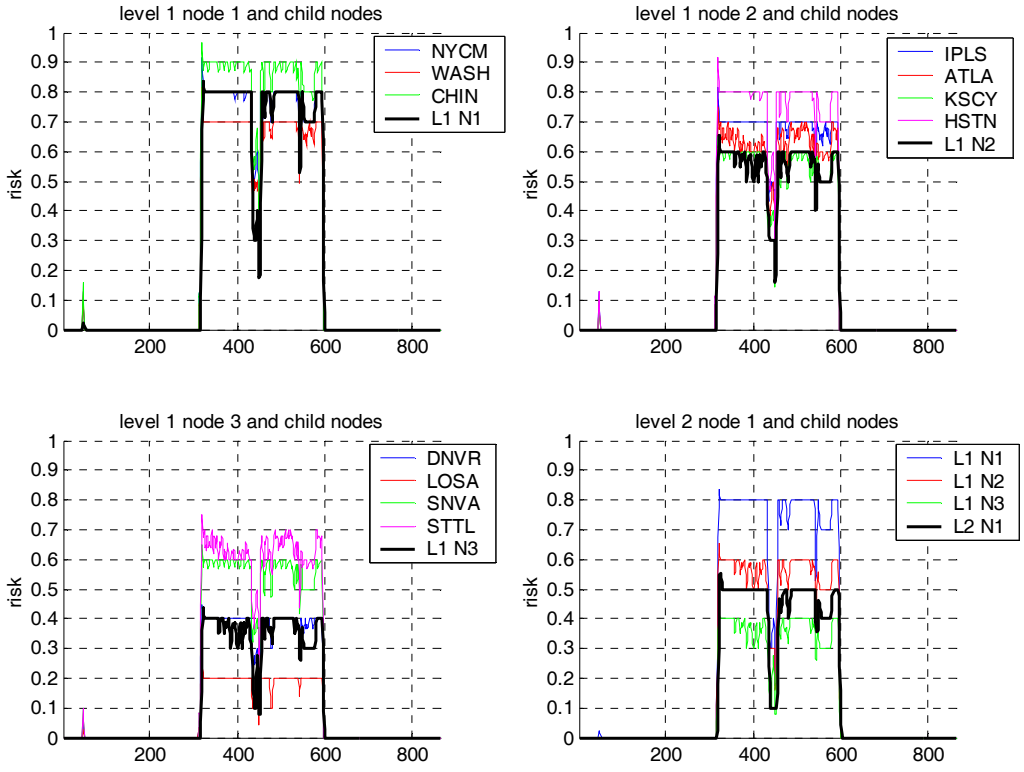

Figure 9: Risk Indices

The output of the MCDM tool is what we would expect. The MCDM approach shows that the risk of a DDoS attack is highest at node 1 of level 1 and lowest at node 3 of level 1. Additionally, the level 2 node shows medium risk of a DDoS attack affecting the *entire* network. Also, the risk indices reached their maxima very quickly. Below is the number of minutes it took to reach each alert stage for each node in the hierarchy.

Alert Stages	Low	Guarded	Elevated	High	Severe
Level 0 Risk Thresholds	**1%**	**15%**	**30%**	**50%**	**80%**
Level 0 Node 1	0	20	20	25	35
Level 0 Node 2	0	20	25	30	35
Level 0 Node 3	0	20	20	25	30
Level 0 Node 4	0	20	25	25	35
Level 0 Node 5	0	20	25	30	X
Level 0 Node 6	0	20	25	30	X
Level 0 Node 7	0	20	20	25	35
Level 0 Node 8	0	25	30	X	X
Level 0 Node 9	0	25	X	X	X
Level 0 Node 10	0	20	25	30	X
Level 0 Node 11	0	20	25	30	X
Level 1 Risk Thresholds	**1%**	**12%**	**25%**	**40%**	**60%**
Level 1 Node 1	20	30	30	35	40
Level 1 Node 2	20	30	30	35	45
Level 1 Node 3	20	30	35	45	X
Level 2 Risk Thresholds	**1%**	**10%**	**20%**	**30%**	**40%**
Level 2 Node 1	30	40	45	45	50

Table 9: Number of Minutes Required to Reach Each Alert Stage after an Attack

4.2. Historical Distributed Denial of Service Attack Delayed

In this experiment, the same historical NYCM DDoS data is "spread" to the other routers. The strength of each router's DDoS attack is determined by the attack levels in Table 7. However, for the routers in the midwest (level 1 node 2), the criteria are not allowed to reach their maxima for 8 hours. And for the routers on the west coast (level 1 node 3), the delay is 16 hours. This represents the DDoS attack propagating from the east to the west.

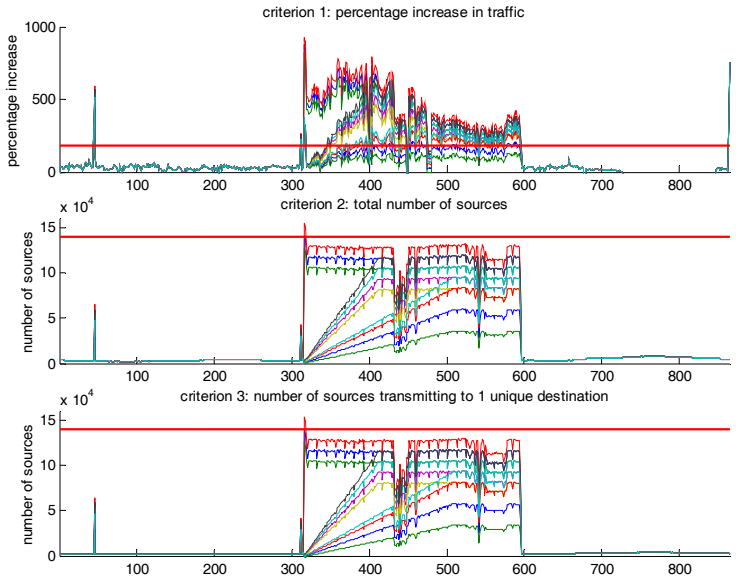

Figure 10: Input Criteria for All Routers

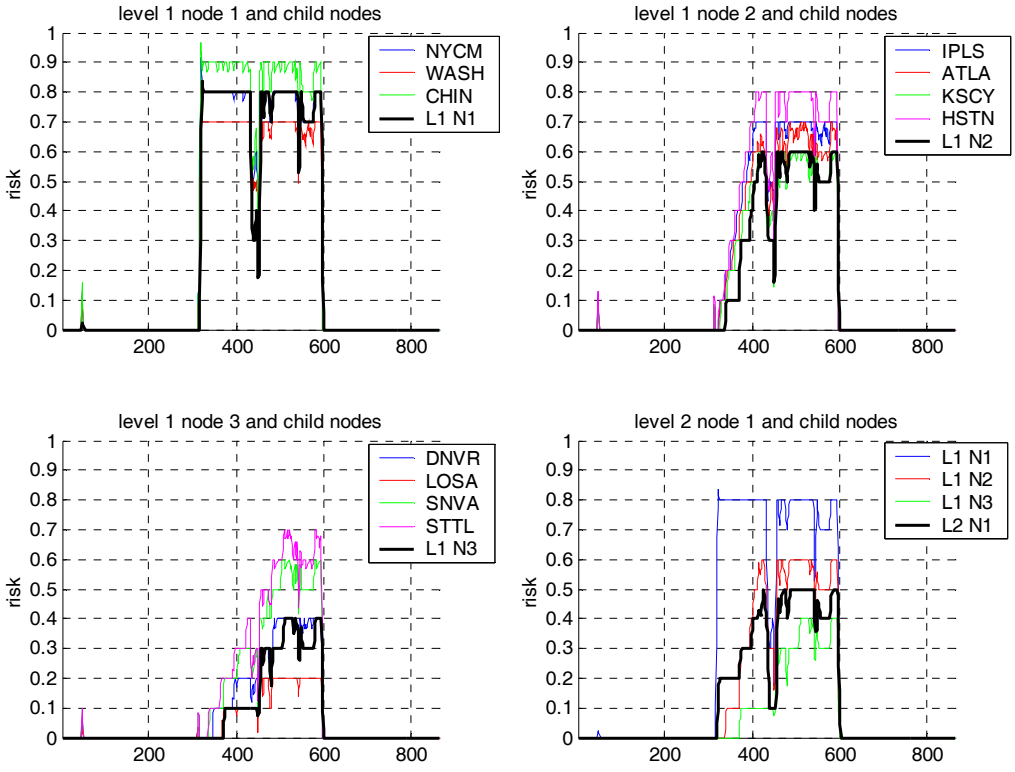

Figure 11: Risk Indices

Again, the output of the MCDM tool is what we would expect. Below is the number of minutes it took to reach each alert stage for each node in the hierarchy.

Alert Stages	Low	Guarded	Elevated	High	Severe
Level 0 Risk Thresholds	**1%**	**15%**	**30%**	**50%**	**80%**
Level 0 Node 1	0	20	20	25	35
Level 0 Node 2	0	20	25	30	35
Level 0 Node 3	0	20	20	25	30
Level 0 Node 4	0	130	205	335	X
Level 0 Node 5	0	145	230	375	X
Level 0 Node 6	0	160	255	425	X
Level 0 Node 7	0	120	185	305	485
Level 0 Node 8	0	410	710	X	X
Level 0 Node 9	0	705	X	X	X
Level 0 Node 10	0	295	485	810	X
Level 0 Node 11	0	260	425	725	X
Level 1 Risk Thresholds	**1%**	**12%**	**25%**	**40%**	**60%**
Level 1 Node 1	20	30	30	35	40
Level 1 Node 2	130	295	310	430	555
Level 1 Node 3	295	710	725	995	X
Level 2 Risk Thresholds	**1%**	**10%**	**20%**	**30%**	**40%**
Level 2 Node 1	35	50	70	320	455

Table 10: Number of Minutes Required to Reach Each Alert Stage after an Attack

As expected, the time it takes to reach higher alert stages went up for the nodes whose traffic was delayed. Below is a table of the change in these times in hours.

Alert Stages	Low	Guarded	Elevated	High	Severe
Level 0 Risk Thresholds	**1%**	**15%**	**30%**	**50%**	**80%**
Level 0 Node 1	0.000	0.000	0.000	0.000	0.000
Level 0 Node 2	0.000	0.000	0.000	0.000	0.000
Level 0 Node 3	0.000	0.000	0.000	0.000	0.000
Level 0 Node 4	0.000	1.833	3.000	5.167	X
Level 0 Node 5	0.000	2.083	3.417	5.750	X
Level 0 Node 6	0.000	2.333	3.833	6.583	X
Level 0 Node 7	0.000	1.667	2.750	4.667	7.500
Level 0 Node 8	0.000	6.417	11.333	X	X
Level 0 Node 9	0.000	11.333	X	X	X
Level 0 Node 10	0.000	4.583	7.667	13.000	X
Level 0 Node 11	0.000	4.000	6.667	11.583	X
Level 1 Risk Thresholds	**1%**	**12%**	**25%**	**40%**	**60%**
Level 1 Node 1	0.000	0.000	0.000	0.000	0.000
Level 1 Node 2	1.833	4.417	4.667	6.583	8.500
Level 1 Node 3	4.583	11.333	11.500	15.833	X
Level 2 Risk Thresholds	**1%**	**10%**	**20%**	**30%**	**40%**
Level 2 Node 1	0.083	0.167	0.417	4.583	6.750

Table 11: Change in Number of Hours Required to Reach Each Alert Stage

In level 1 node 2, the increase in time required to reach the severe alert stage was 8.5 hours. This compares well with the delay of 8 hours that was added to the Midwestern nodes. In level 1 node 3, the increase in time required to reach the high alert stage was 15.833 hours. Again, this compares well with the delay of 16 hours that was added to the west nodes. The increase in time for the lower alert stages was less because the anomaly traffic was not completely removed during the delay.

Thus, the MCDM tool does not predict attacks. Delaying network anomalies in one part of the network and not in another does not help the MCDM tool assess the threat quicker in the delayed part of the network. The only way the MCDM method could be predictive is if the criteria used are predictive. For example, one might consider using criteria in the higher levels that detects similarities in traffic patterns between different nodes. Thus, if a traffic pattern is detected in a node that is similar to earlier anomalies in other nodes, then that particular node could be isolated early on.

4.3. Linearly Increasing Criteria

In this experiment and the following, the criteria being input into the level 0 nodes are arbitrarily chosen. In this case, the NYCM criteria are chosen to increase linearly just past the maximum profiles. The criteria for the remaining ten routers were determined by multiplying the NYCM criteria by the attack levels in Table 7. Since all three criteria are increasing simultaneously, this represents the threat of a DDoS anomaly steadily rising.

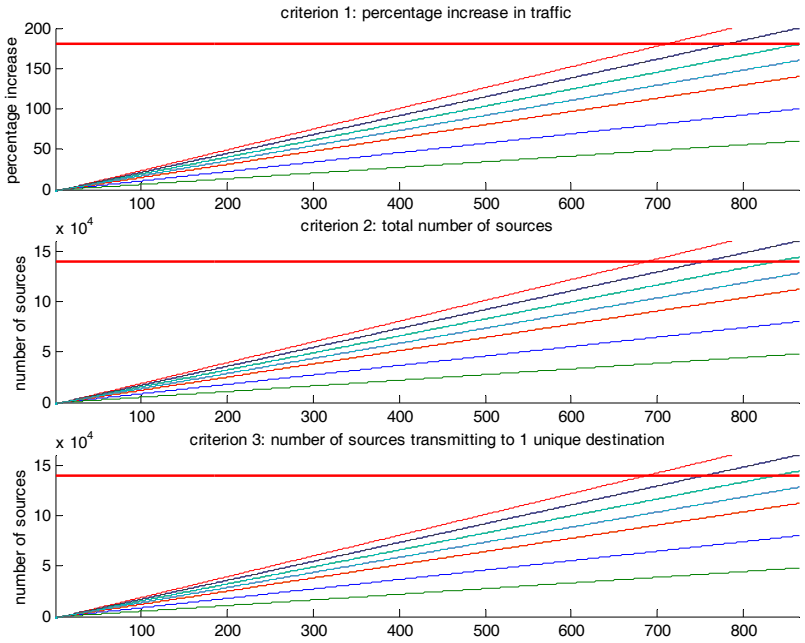

Figure 12: Level 0 Input Criteria for All Routers

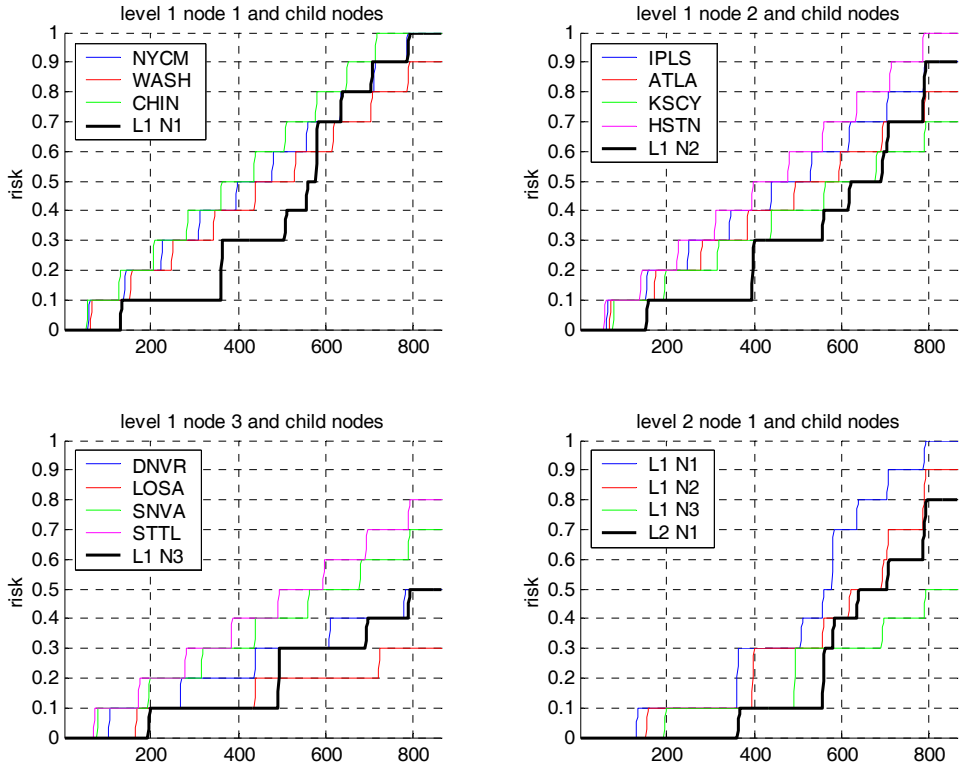

Figure 13: Risk Indices

The MCDM tool generally performed as expected. With all three criteria rising, the risk index rose as well. However, in the higher level nodes, and *not* in the level 0 nodes, the risk indices did not grow at a consistent rate. Sometimes the rate was faster and sometimes it was slower. The likely causes for this are the criteria and weight coefficients used in the higher levels. The criteria used in the higher levels have different weights and they tend to conflict much more than the level 0 criteria.

4.4. Linearly Increasing Criteria – Higher Slope

This experiment is the same as the last, except that the slope at which the criteria increase is three times larger. Compared to the previous simulation, this represents a situation in which the threat of a DDoS grows three times faster.

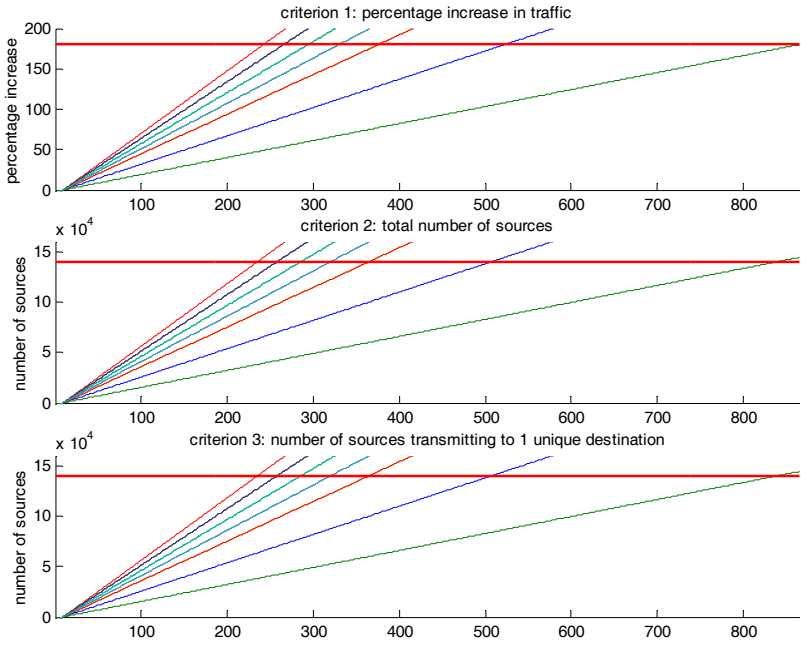

Figure 14: Level 0 Input Criteria for All Routers

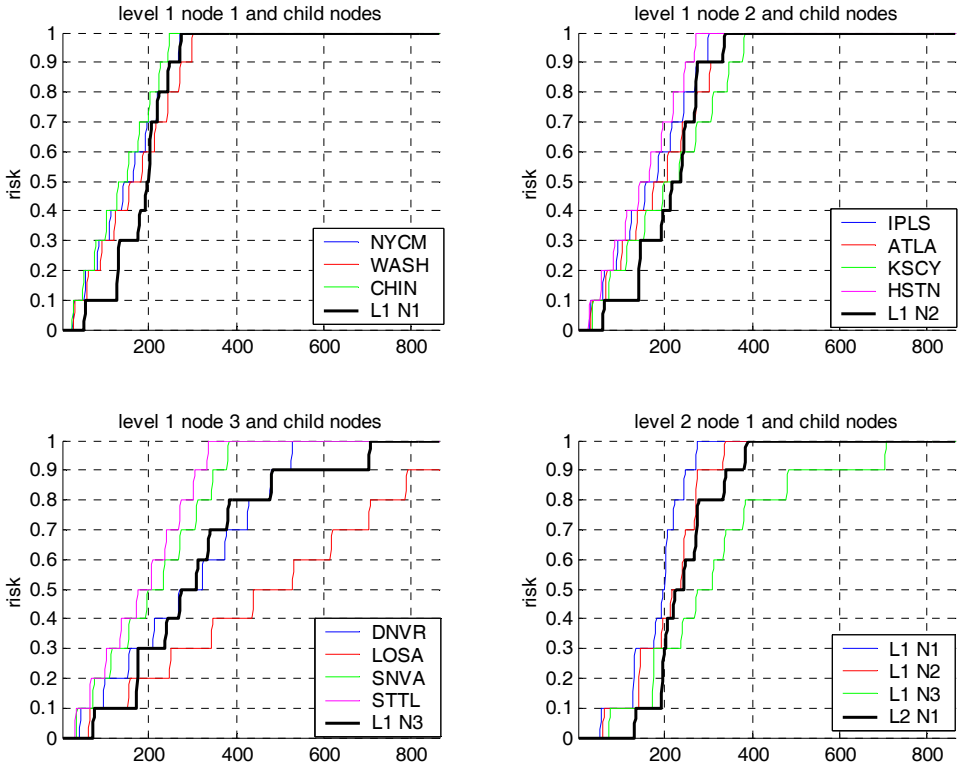

Figure 15: Risk Indices

Again, the MCDM tool performed as expected. The higher slope of the input criteria translated into a higher slope of the output risk indices. This meant that it took roughly one-third of the time for the risk to reach its maximum. For example, in the first simulation it took 793 time steps for the risk index of level 1 node 1 to reach its maximum. However, when the slope of the input criteria was tripled, that time reduced to 275 time steps. In addition, in contrast with the level 0 risk indices, the higher level risk indices did not increase in a smooth and consistent fashion.

4.5. Linearly Increasing Criteria – One Small Criterion

In this simulation, two of the three criteria linearly increase to a maximum while the other criterion remains small. As in the previous simulations, the criteria are scaled according to the attack levels in Table 7. Thus, this is a situation where it may look like the threat of a DDoS is growing, but the growth in some of the criteria may be due to some other type of anomaly, such as a worm. For example, if a worm were spreading rapidly, it would most likely cause the traffic and the total number of sources to increase. However, since the nature of a worm is to attempt to spread itself to new hosts, then it is not likely that a worm would cause much of an increase in the number of sources transmitting to *one* unique destination. Thus, this is a case where only two of the three criteria increase. Thus, the MCDM tool should be tuned so that it will not generate a false positive in this situation.

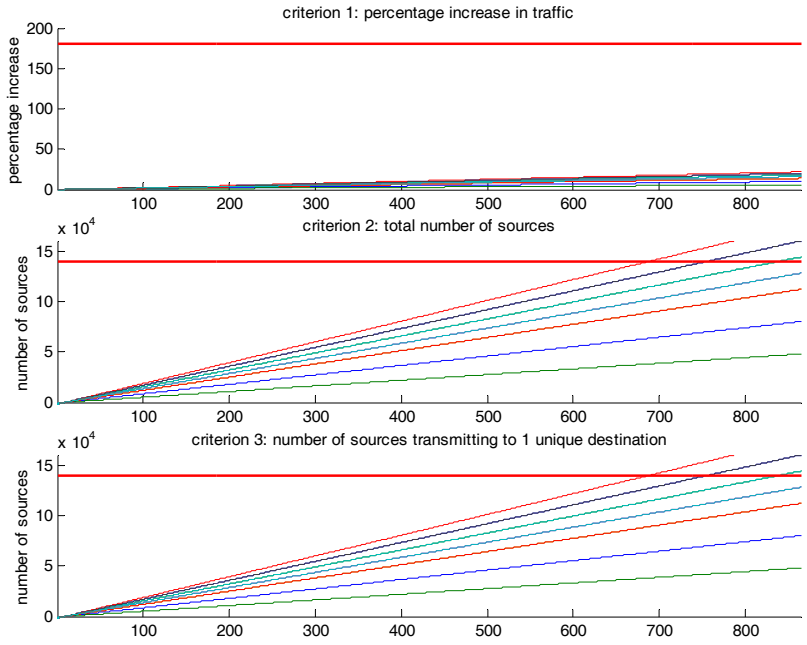

Figure 16: Level 0 Input Criteria for All Routers

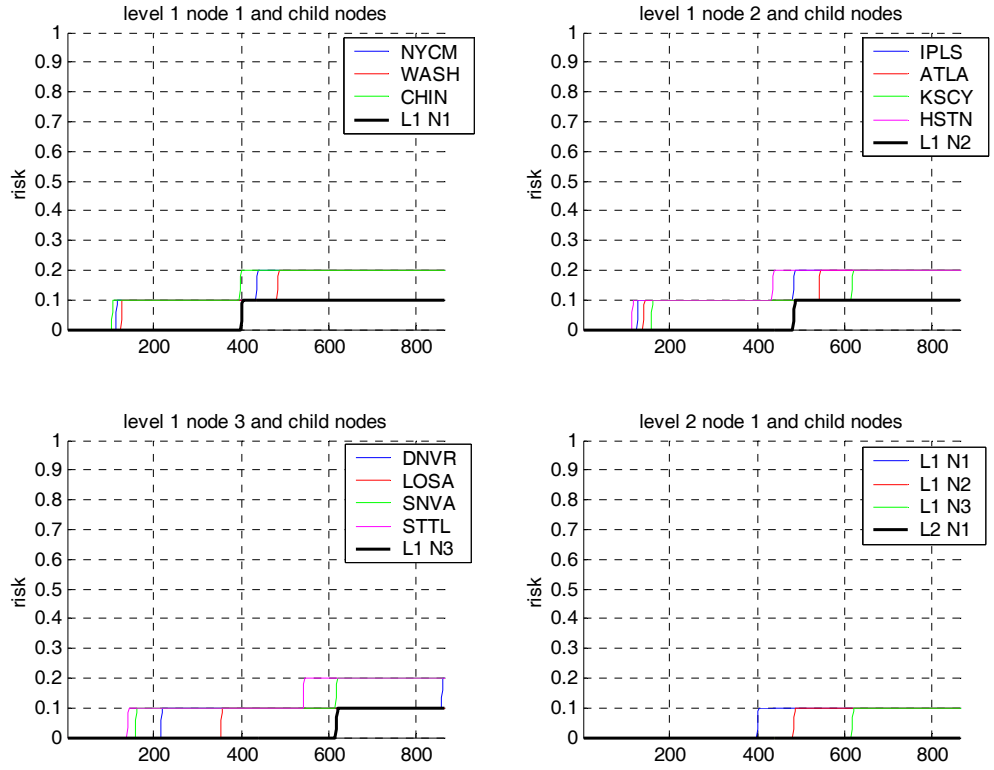

Figure 17: Risk Indices

In this simulation, the MCDM approach performed well. The risk indices were always very small. Therefore, when faced with conflicting criteria, the MCDM tool did not classify this anomaly as a DDoS.

4.6. Linearly Increasing Criteria – Two Small Criteria

In this simulation, only one of the three criteria linearly increase to a maximum while the other two criteria remain small. This represents another situation where another type of attack may be causing some of the DDoS criteria to rise. Thus the goal is to ensure that the MCDM tool does not classify this as a DDoS.

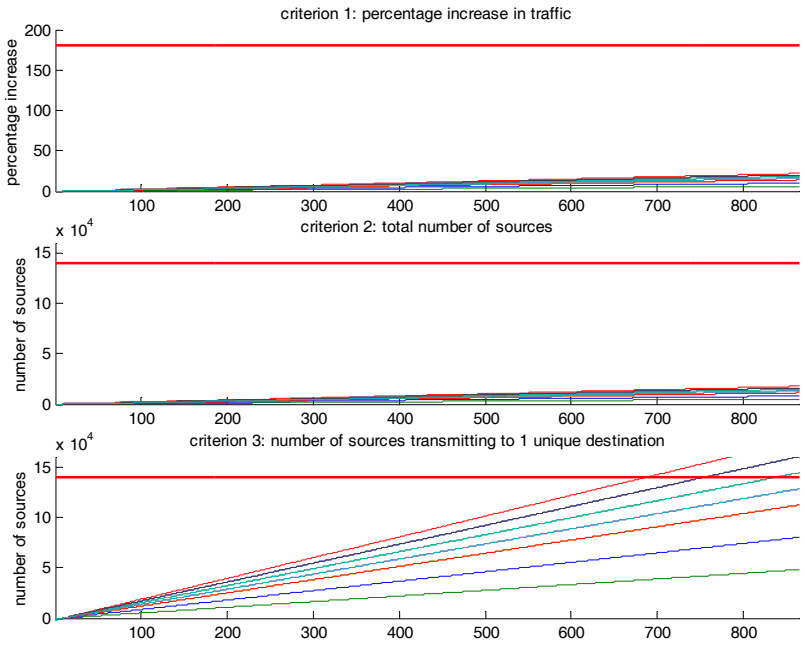

Figure 18: Level 0 Input Criteria for All Routers

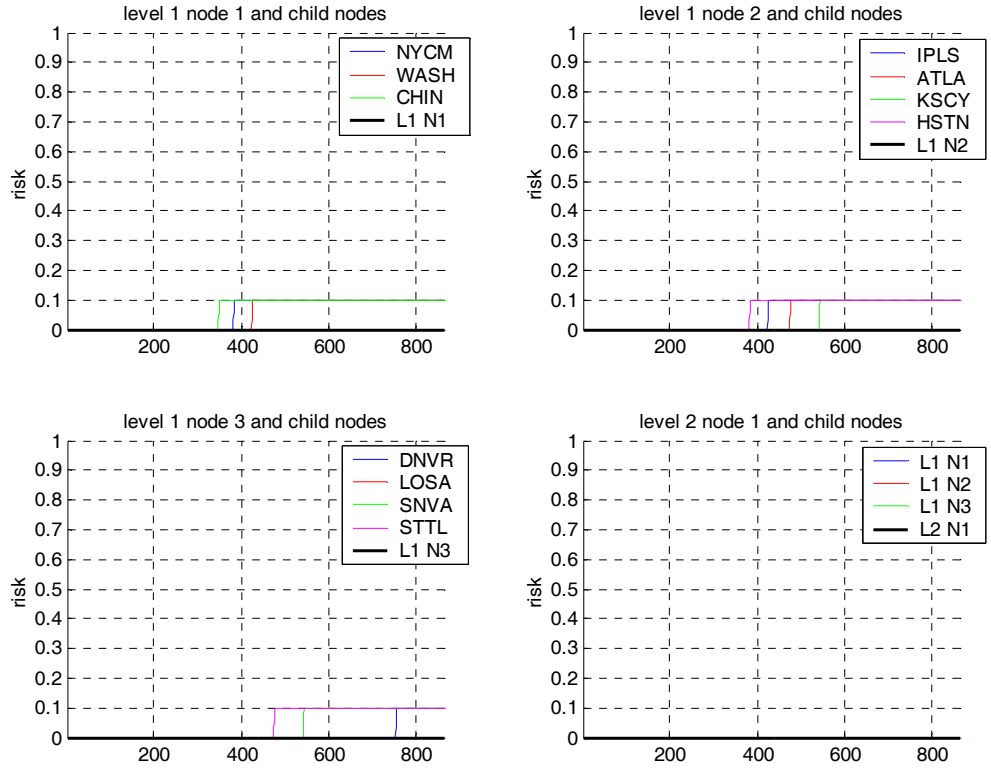

Figure 19: Risk Indices

Again, the MCDM tool performed as expected. Since only one of the three criteria was large, the risk indices were even lower than in the experiment of the previous section. Thus, the MCDM approach will not classify such an anomaly as a DDoS.

4.7. Linearly Increasing Criteria – Varying Sizes

In this simulation, one of the three criteria linearly increases to a maximum, another criterion linearly increases to a medium level, and the other criterion stays small. As in the previous experiments, the criteria are scaled according to the attack levels in Table 7. This is another situation in which another type of anomaly may cause some of the DDoS criteria to increase. The goal is to see that this anomaly is *not* classified as a DDoS.

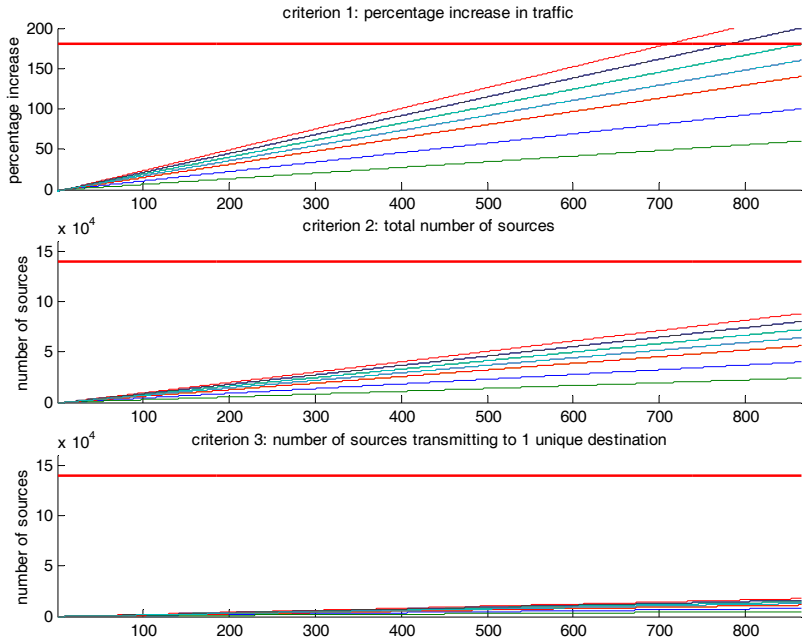

Figure 20: Level 0 Input Criteria for All Routers

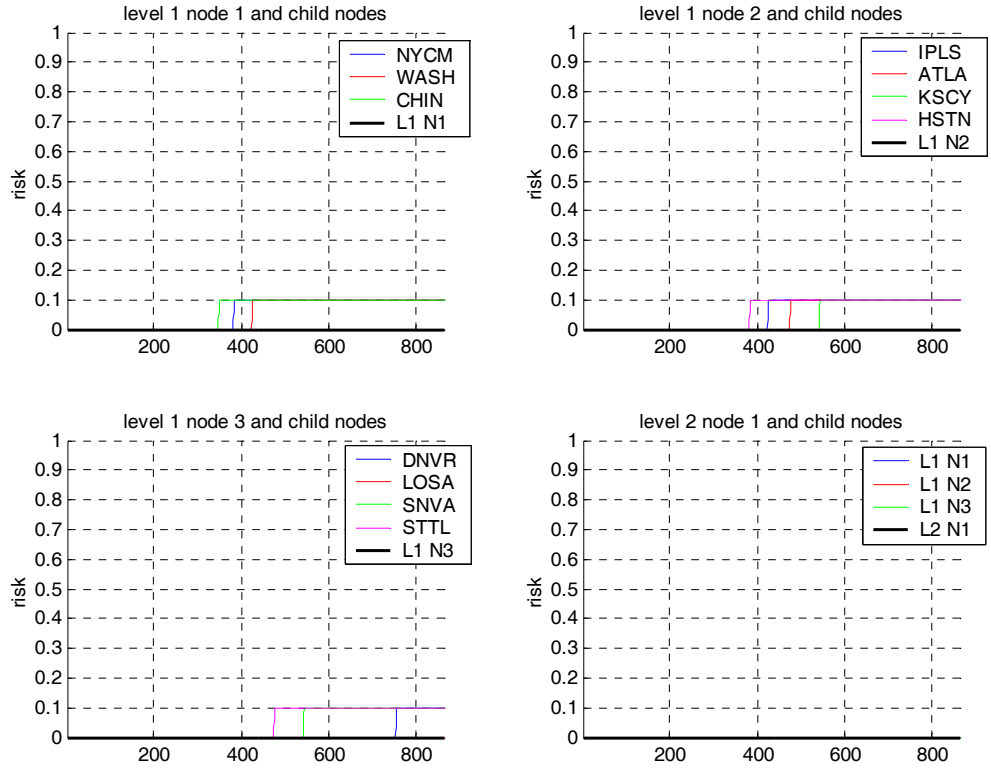

Figure 21: Risk Indices

In this simulation, the MCDM approach did not classify this anomaly as a DDoS. Again, when the input criteria conflicted with each other, the MCDM tool output very low risk indices.

4.8. Constant and Sinusoidal Criteria

In this last simulation, two of the criteria for the NYCM router remain constant while the other oscillates near the maximum profile. Again, the scale of the criteria for the remaining routers is determined by the attack levels in Table 7. The goal of this experiment is to see how sensitive the MCDM tool is to small variations in the criteria.

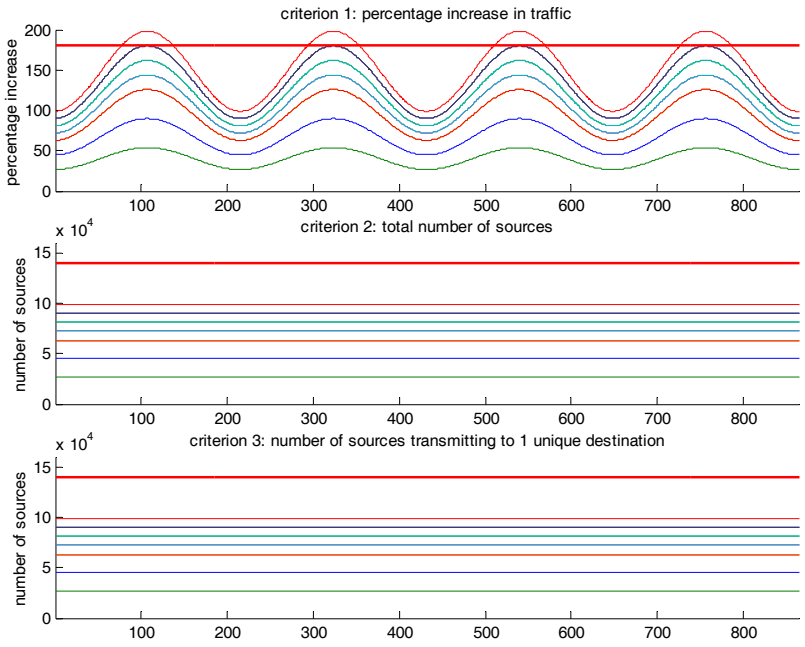

Figure 22: Level 0 Input Criteria for All Routers

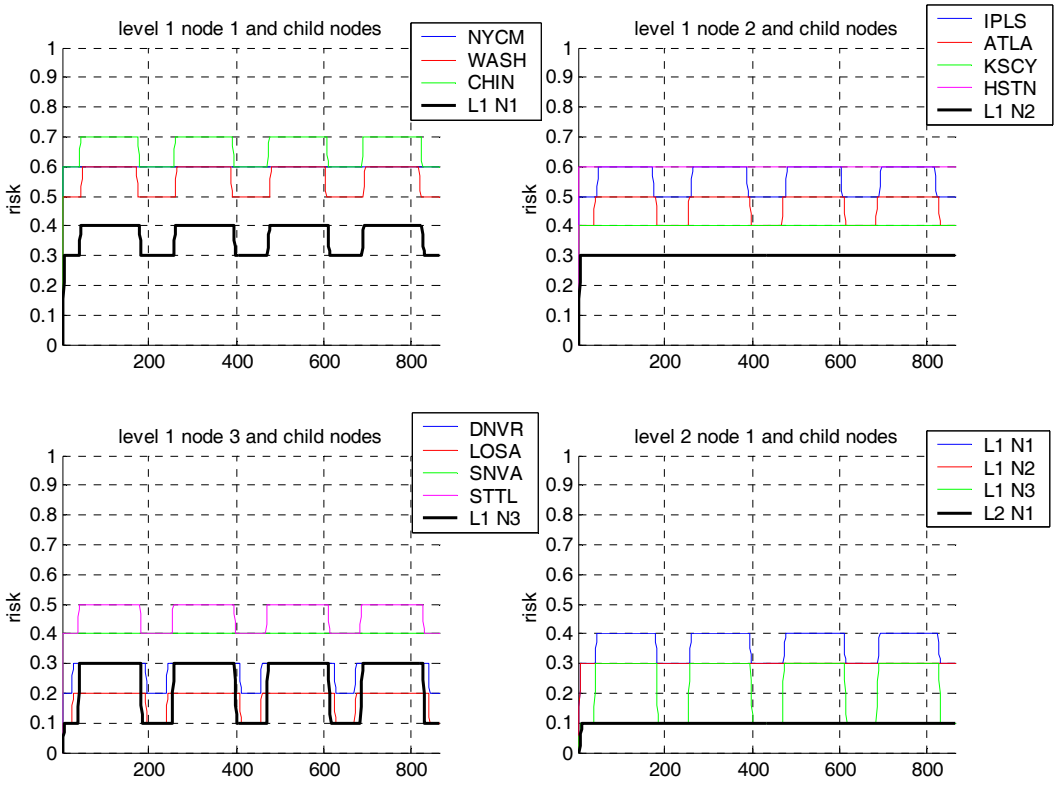

Figure 23: Risk Indices

In this simulation, the variation in criterion 1 for level 0 never caused a variation in the level 0 risk index of more than 0.1, even though criterion 1 oscillated between the midpoint and the maximum profile for some routers. For example, criterion 1 for the HSTN router varied between 90% and 180%, yet the HSTN router's risk index was a constant 0.6. By contrast, however, criterion 1 for LOSA only varied between 27% and 54%, but its risk index oscillated back and forth between 0.1 and 0.2. Therefore, the relationship between the input and output has some discrete properties. Sometimes, only a small change in one input is needed to increase or decrease the risk index by 0.1.

5. Conclusions

These results are encouraging. We have shown that the MCDM approach can detect a DDoS attack on a network. Moreover, the parameters of the MCDM approach were tuned to reduce the possibility of false positives.