Immune Inspired Approaches to Fault and Intrusion Detection in Ad Hoc Wireless Networks

Von der Fakultät für Elektrotechnik und Informatik der
Gottfried Wilhelm Leibniz Universität Hannover
genehmigte Habilitationsschrift

zur Erlangung der venia legendi für das Fachgebiet Informatik

vorgelegt von

Dr. rer. nat. Martin Drozda

2014

Address: Dr. rer. nat. Martin Drozda
Leibniz Universität Hannover
FG Simulation und Modellierung
Welfengarten 1
30167 Hannover
Germany

Email: drozda@sim.uni-hannover.de

Abstract

In this document we summarize our research efforts related to computational interpretation of two basic immune mechanisms: co-stimulation and priming. This interpretation was done in the scope of misbehavior detection for ad hoc wireless networks. When formulating co-stimulation and priming as a computational approach we abstracted away from the molecular nature of these mechanisms. Instead we concentrated on their logic structure. Needless to say, despite applying such a simplified view, the path to a useful computational interpretation was not straightforward. This fact underscores the conceptual character of our research.

Our computational interpretation of co-stimulation allows any detection system to introduce two desirable properties: energy efficiency and false positives control. Additionally, it introduces the option to exchange energy efficiency for misbehavior detection rate, while keeping the false positives rate unchanged. Priming enables to check whether network operational conditions are within the limits prescribed by priming thresholds. Since priming is based on co-stimulation, it inherits the same properties.

We provide a summary of the challenges related to the design of co-stimulation and priming based architectures. We argue that co-stimulation and priming are rather general paradigms with possible applications in other areas than misbehavior detection. Additionally, we summarize our results related to the efficiency of negative selection in misbehavior detection scenarios. We outline the key features of AIS-Lib, a library for performance evaluation of immune inspired detection systems. Finally, we summarize our results achieved by applying a locality sensitive hashing technique to intrusion detection.

Zusammenfassung

In diesem Dokument fassen wir unsere Forschungsergebnisse zum Thema Fehlverhaltenerkennung für drahtlose ad-hoc Netze zusammen, welche sich auf zwei grundlegende Mechanismen des biologischen Immunsystems (Co-stimulation und Priming) beziehen. In diesem Zusammenhang wurde eine ausführliche Analyse und Interpretation der beiden Mechanismen durchgeführt, um sie erfolgreich im Bereich der Fehlverhaltenerkennung einzusetzen. Bei der Formulierung geeigneter Interpretationen der molekularen Grundlagen (zur Umsetzung in ein technisches System) haben wir uns auf die logische Struktur der beiden Mechanismen konzentriert. Trotz der Verwendung einer auf diese Weise vereinfachten Ansicht, war der Weg zu einer nützlichen Umsetzung nicht direkt. Dieser Sachverhalt unterstreicht den Konzeptcharakter unserer Forschung.

Unsere Umsetzung von Co-stimulation und Priming ermöglicht jedem Fehlverhaltenerkennungsystem zwei wichtige Eigenschaften einzuführen: Steuerung der Energieeffizienz und Minimierung der Anzahl von Falschmeldungen. Zusätzlich bietet sich die Möglichkeit, Energieeffizienz gegen Fehlverhaltenaufklärungsrate auszutauschen, dabei bleibt aber die Anzahl von Falschmeldungen konstant. Priming ermöglicht zu überprüfen, ob die Betriebsbedingungen eines ad-hoc Netzwerkes innerhalb eines eingegrenzten Bereiches bleiben. Diese Begrenzungen werden durch ”priming”-Schwellen (Schwellwerte) definiert. Da Priming auf Co-stimulation basiert, besitzt es die gleichen grundlegenden Eigenschaften wie Co-stimulation.

Wir geben eine Zusammenfassung unserer Forschungsergebnisse, welche sich mit den Problemen und Herausforderungen bezgl. Architekturen die Co-stimulation and Priming verwenden, befassen. Zusätzlich fassen wir unsere Forschungsergebnisse zusammen, die sich auf die Leistungsfähigkeit der negativen Selektion in Bereich der Fehlverhaltenerkennungszenarien beziehen. Wir geben weiterhin einen Überblick über AIS-Lib, einer Bibliothek zur Leistungsbewertung von immuninspirierten Fehlverhaltenerkennungsystemen. Schließlich fassen wir unsere Forschungsergebnisse zusammen, die sich auf Umsetzung von einem ”locality preserving” Hashingverfahren zur Erkennung von Einbrüchen in ein Netzwerk (intrusion detection) beziehen.

Contents

1 Introduction

The Biological immune system (BIS) protects its host against extraneous agents that could cause harm. The BIS can be found in living organisms ranging from plants to vertebrates. Even though the BIS complexity in these various life forms can be very different, the common goal is to sustain survival in often unforeseeable conditions. The BIS can be succinctly described as a protection system that is based both on prior experience possibly accumulated over an evolutionary time span and short-term adaptation that occurs during the lifetime of the host. Whereas for example plants only rely on the former type of protection, *innate immunity*, more complex life forms additionally employ the latter type, *adaptive immunity*. It is however important to note that the existence of adaptive immunity, its internal mechanisms and interplay with innate immunity is also a result of evolutionary priming.

Due to the efficiency in protecting its host, the BIS has become an inspiration for designing protection systems. Besides this area, it has become an inspiration in areas such as machine learning, optimization, scheduling etc. Artificial Immune Systems (AISs), the technical counterpart of the BIS, have become an independent research direction within the field of computational intelligence.

Computer and network security is an area which gained an increased interest, mainly due to the omnipresence of the Internet. It is the role of secure protocols to guarantee data integrity and user authentication. Unfortunately, flaws in secure protocols are continuously being found and exploited as the Internet experience shows [45]. The history of security of home and small mobile computing platforms points out that such attacks can disrupt or even completely interrupt the normal operations of networks [43]. There are several thousand families of viruses and worms recorded. Some of these families consist of several thousands or even tens of thousand viruses that were created by virus kits, a specialized software aimed at automated virus creation.

Ad hoc wireless networks do not rely on any centralized or fixed infrastructure. Instead each participating wireless device acts as a router for forwarding data packets. The advantage of such an architecture is the ease of deployment. This advantage is however at the cost of a more complex maintenance. Additionally, in many scenarios it is expected that the participating wireless devices will be resource restricted due to their reliance on battery power. This further implicates energy aware hardware with a lesser computational throughput.

The above limitations establish the basic motivation for designing autonomous detection and response systems that aim at offering an additional line of defense to the employed secure protocols. Such systems should provide several layers of functionality including the following [16]:

(i) distributed energy efficient self-learning and self-tuning with the aspiration to minimize the need for human intervention and maintenance,

(ii) active response with focus on attenuation and possibly elimination of negative effects of misbehavior on the network.

In the following we will introduce the basic concepts of an energy efficient misbehavior detection system for ad hoc wireless networks. This approach is inspired by the role of co-stimulation and priming in the BIS. We will also discuss the role of features necessary for misbehavior detection. A *feature* in our context is understood to be a performance measure that allows for an efficient reasoning about whether a node misbehaves or works normally.

Document Organization

The rest of this document is organized as follows. In Section 2 we give a short overview of the BIS. We introduce co-stimulation and priming, and describe their role in the BIS. We also explain how the innate and adaptive immune systems cooperate when detecting a pathogen. In Section 3 we review the related work that used co-stimulation as a means for improving misbehavior detection performance. Additionally, we review the state-of-the-art in the area of misbehavior detection. In Section 4 we discuss our computational interpretation of co-stimulation and priming. In Section 5 we provide details on our co-stimulation algorithm and in Section 6 on our priming algorithm. In Section 7 we give an overview of our experimental methodology and give insights on the performance of co-stimulation and priming. In Section 8 we summarize our results. In Section 9 we give concluding remarks. We also argue that co-stimulation and priming are rather general paradigms with possible applications in other areas than misbehavior detection. Finally, Section 10 provides a list of enclosed documents.

2 The Biological Immune System

The Biological immune system (BIS) of vertebrates is a remarkable defense system, able to quickly recognize the presence of foreign microorganisms and thereafter eliminate these *pathogens*. Pathogens are common microorganisms such as viruses, bacteria, fungi or parasites. When confronted with a pathogen, the BIS often relies on a coordinated response from both of its two vital parts:

- the *innate system*: the innate immune system is able to recognize the presence of a pathogen or tissue injury, and is able to signal this to the adaptive immune system.
- the *adaptive system*: the adaptive immune system can develop during the lifetime of its host a specific set of immune responses and provide immunological memory. An immunological memory serves as a basis for a stronger immune response, should a pathogen re-exposure happen.

2.1 Co-stimulation and Priming

The form and amplitude of immune responses is pathogen dependent. Often, an immune response within the BIS is based on a feedback mechanism between the innate and adaptive immune systems. Such a feedback can result in a feedback loop, in which the innate immune systems further stimulates the adaptive immune system, and vice versa [21]. For example, a pathogen gets eliminated, if it was recognized by the adaptive immune system as a pathogen, and at the same time, the innate immune system signals that this pathogen causes some damage to the human organism. Under this specific immune reaction, only damage inflicting or *infectious* cells get eliminated by the BIS.

This demonstrates that a two-way communication, hereafter referred to as *co-stimulation*, between the innate and adaptive immune systems is common. Immunologists such as Frauwirth and Thompson describe co-stimulation as the involvement of *"reciprocal and sequential signals between cells"* in order to fully activate a lymphocyte [21]. The role of lymphocytes is to recognize a specific pathogen, to trigger a corresponding immune reaction and in some forms they are also capable of pathogen elimination.

Priming in the BIS describes the effects of a first encounter of an immune cell with a pathogen. More specifically, immunologists define priming as the activation and clonal expansion of certain

immune cells into effector cells that are then capable of inducing a full immune response against a specific pathogen.

Communication capabilities within the BIS received an increased interest from the Artificial immune systems (AIS) community and evolved into an independent research direction. Several different types of danger, safe and amplifying signals were proposed within the Danger theory due to Aickelin et al. [1].

2.2 Selection Principles

T-cells and B-cells are lymphocytes providing pattern matching capabilities necessary for pathogen detection. Before being released into the body they are subject to a selection process. The goal of this selection process is to produce T-cells and B-cells with a high degree of diversity [35].

T-cells mature in the thymus in two stages called (i) *positive selection* and (ii) *negative selection*.

(i) Immature T-cells in the thymus get first tested on self cell reactivity. T-cells that do not react with self cells at all are subject to removal.

(ii) T-cells that survived the first step are tested on whether they can react with self cells too strongly. If this is true, the result is again removal.

Remaining T-cells are released into the body. Such T-cells are mature but remain *naive* until their activation. In order these two stages to be achievable, the thymus is protected by a blood-thymic barrier that is able to keep this organ pathogen-free. As a result, mature T-cells are reactive with cells that could not be present in the thymus, i.e. with non-self cells. They are also weakly self reactive but they are unable to get activated by a self cell. Unfortunately, the repertoire of self cells in the thymus does not have to be complete. Additionally, not all self cells are expressed in the thymus at adequate levels to allow for negative selection. This can lead to reactivity with self cells, i.e. to *autoimmune* reactions (false positives). In order to suppress autoimmune reactions, the BIS applies several mechanisms that allow for peripheral immunoregulation [40].

Creating of new B-cells with an improved pathogen recognition ability is done through cloning. Cloning of B-cells allows for some "error". This error is believed to be inversely proportional to the matching ability of a given B-cell. The weaker is the ability, the greater is the error allowance. The process of B-cell cloning with an error allowance is called *somatic hypermutation*. The purpose of somatic hypermutation is to diversify the matching ability of B-cells.

Both T-cells and B-cells that were not able to recognize any pathogen get removed. This happens to most B-cells and T-cells and can be understood as an instance of positive selection (only the fittest stay). Unlike B-cells, T-cells can only recognize pathogen fragments expressed by another type of cells called Antigen presenting cells (APC). Both T-cells and B-cells can mature into *memory* B-cells and T-cells. In contrast to normal T-cells and B-cells, memory cells live much longer, thus allowing the host to keep its immunological memory over a long period.

2.3 Discussion

Our goal is to benefit from the different capabilities of the innate and adaptive immune system. We build on the large body of evidence that indicates that innate and adaptive immune systems tightly cooperate when detecting a pathogen; see e.g. [28, 38]. Liu and Janeway [34] state that pathogen recognition mechanisms that combine T-cell receptor binding and APC based co-stimulation are the most efficient inducers of clonal activity of certain T-cells; see Fig. 1. Medzhitov and Janeway further comment [38]: *"This whole mechanism ensures that a T-cell will normally receive both signals*

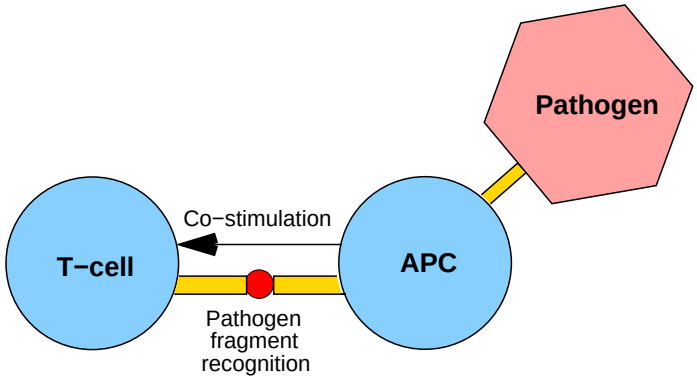


Figure 1: Pathogen recognition: co-stimulation and pathogen presentation are present during lymphocyte activation as well as any subsequent pathogen recognition.

necessary for activation only if the peptide recognized by the T-cell receptor is derived from the pathogen that initially induced the co-stimulation activity". This statement provides an interesting clue on the design of misbehavior detection systems. It suggests that when decreased detection performance is exchanged for increased adaptivity, it is necessary to ensure that any behavior, being currently classified by the adaptive detection system, is identical with the behavior used to train this adaptive system. Regarding misbehavior detection systems, such a strategy could be capable of decreasing the probability that a behavior, having an anomalous or random character, gets incorrectly classified as a known misbehavior type. For this reason, when designing our misbehavior detection system, an efficient implementation of this strategy was one of our key ambitions.

The adaptive immune system benefits from its capability to produce immune cells with clonally distributed receptors. This random approach has two basic drawbacks: i) it cannot avoid creating immune cells that recognize self cells and ii) it cannot avoid creating immune cells that do not recognize any other cell: self or non-self. Therefore this process is combined with negative and/or positive selection in order to remove such immune cells. When deciding whether a cell is self or non-self, the BIS takes advantage of the capability of the innate immune system to recognize pathogen-associated molecular patterns (PAMPs) [38]. PAMPs are invariant molecular structures associated with each pathogen. This feature of the innate immune system has developed over evolutionary time. A successful pathogen could try to avoid being recognized either by changing its PAMP or simply by avoiding such evolutionarily selected receptors. Changing PAMP, even though feasible, requires a genetic recombination of the pathogen. Such a recombination could result in a cell that partially or completely lost its pathogenicity. It is indeed the case that some pathogens managed to avoid being detected by the slowly evolving receptors of the innate immune system. The Nature's answer to this problem seems to be a specialized class of APCs called dendritic cells [4]. Dendritic cells are especially helpful in recognizing the danger connected with pathogens with high mutation rates, e.g. viruses.

When translating the mechanisms of the BIS into a computational approach we decided to focus on the information flow among the various agents of the innate and adaptive immune system. We abstracted away from the particularities of the various selection and learning mechanisms discussed above. We assumed that they can be substituted by general purpose machine learning algorithms such as decision trees, support vector machines etc. We were inspired by the co-stimulation process that is required to activate a lymphocyte. We understood co-stimulation as an interplay between the adaptive and innate immune system when detecting a pathogen. When interpreting priming, we

assumed that the innate immune system can provide strong clues on the pathogenicity of a non-self cell.

Our decision to apply standard machine learning algorithms, instead of their immune inspired counterparts, allowed us to concentrate on the control flow of our co-stimulation and priming approaches. The key question that we aimed at answering was whether co-stimulation and priming can be translated into computational approaches that inherit some of the key characteristics of the BIS: detection efficiency and ability to suppress false positives (autoimmune reactions).

3 The State-of-the-art

In this section we review the few prior approaches that took advantages of co-stimulation. Additionally, we review several other approaches that deal with detecting mainstream types of misbehavior such as data packet dropping or wormholes.

3.1 Co-stimulation Approaches

In one of the first BIS inspired works, Hofmeyr and Forrest [24] described an AIS able to detect anomalies in a wired TCP/IP network. Anomaly detection was based on the negative selection algorithm [20]. After this mechanism detected an anomaly, a message was sent to a human operator. He was given 24 hours to confirm a detected attack. This means, two qualitatively different sorts of classification were used: negative selection and human expertise. The second approach is only applied, if a co-stimulation in the form of a message is received.

Sarafijanović and Le Boudec [41, 33] introduced an AIS for misbehavior detection in mobile ad hoc wireless networks. A local mechanism applied by each node in the network required a co-stimulation in order to classify a neighbor as misbehaving. The origin of co-stimulation was a TCP connection source that perceived data losses. The information about a perceived data loss was forwarded along the connection. The authors observed that their form of co-stimulation reduces the false positives rate. The disadvantage of their approach is its tight coupling with a transport layer protocol (TCP), thus negatively influencing energy consumption.

3.2 Other Misbehavior Detection Approaches

Even though our focus stays on co-stimulation and its suitability for misbehavior detection, we would also like to offer insight into related approaches not taking advantage of this technique.

Marti et al. in [37] introduced two techniques for data dropping detection in ad hoc networks: watchdog and pathrater. A watchdog allows for collecting traffic information about neighboring nodes using promiscuous radio mode. Based on this information, a pathrater can assess route reliability, thus maximizing the chance that data packets get correctly delivered. Due to the energy consumption connected with operation in promiscuous node, the watchdog technique is not suitable for energy constrained ad hoc networks, even though, it offers high data packet dropping detection rates [17].

Hu et al. [25, 26] introduced two approaches using *packet leashes* for wormhole detection in mobile wireless networks. A wormhole is an out-of-band connection between two devices which allows an attacker to manipulate the network topology and thus the routing of packets. A leash either consists of an authenticated timing or location based information. A metric calculating whether a packet has traveled further than allowed (or physically possible) is applied to the leashes. While the first approach requires a tightly time synchronized network, the other one requires GPS (Global

Positioning System) based geographical information. The disadvantage of the former approach is the increased message and authentication complexity, the disadvantage of the latter approach is the increased energy consumption connected with GPS device operation.

Huang and Lee [27] introduced a cooperative approach for intrusion detection in ad hoc networks. It takes advantage of a comprehensive set of 141 data traffic features. The sampling rate for the feature computation was 5 seconds. It applies a decision tree for the classification of traffic samples based on these features. It thus requires a learning period in which the classifier gets trained. To improve the energy efficiency of their approach, a cooperative approach is considered. Under this approach a clique of nodes elects a cluster head. This cluster head does network monitoring on behalf of other clique members until a new cluster head is elected. The disadvantage of such an approach is the necessity to overhear the data traffic in promiscuous mode and to reelect new cluster heads so that the monitoring load gets evenly distributed among the clique members.

Bhuse et al. [8] proposed an approach for data packet dropping detection that does not rely on promiscuous mode. In their approach each connection destination node sends to the source connection node a statistic about received data packets. This is done over an alternative route that is computed by the routing protocol. The received statistic is then compared with a similar statistic computed by the source node. Based on this, the source node can decide whether all nodes on the connection cooperate in data packet forwarding. This approach does not allow to detect the individual misbehaving node. The energy consumption overhead of this approach when the DSR protocol is used for alternative route discovery is 0.6-2.6% for paths of length 3 to 13 hops. The ability of this approach to find an alternative route decreases as the number of data dropping nodes increases as well as with the connection length. For example, when 10% nodes are dropping data packets the reported success rate is about 75% and 25% for route lengths of 3 and 13 hops, respectively.

Gonzales et al. [23] proposed an approach for data dropping detection in ad hoc networks. Their approach is based on the principle of flow conservation. Under this approach, it is assumed that the number of data packets, that a node receives, equals the number of data packets that it forwards excluding the cases when the given node is also the destination node. The success of their approach relies on data traffic overhearing in promiscuous mode. The focus of their experimental evaluation was to determine the ratio of data packets that a node is allowed to drop without being detected. This ratio is measured relative to the maximum number of data packets that a node is allowed (or expected) to drop. Their experimental evaluation shows that this ratio is about 10-15%.

Krishnamurthy et al. [32] introduced a machine learning misbehavior detection approach for wireless sensor networks. They considered several misbehavior types: continuous signal jamming, signal jamming applied only if there is other radio transmission detected, data packet redirecting and data packet dropping. The latter misbehavior type required data traffic observation in promiscuous mode. Classification was done using a linear discriminant analysis and/or a fixed-width clustering with different distance measures. They tested their approach on a sensor network based on Crossbow MicaZ motes and TinyOS. The memory footprint of their approach was 31,342 bytes and 3,500 bytes of flash memory and data memory, respectively. The reported detection accuracy for data packet dropping was nearly 100% with linear discriminant analysis applied.

3.3 Further Comments

Hofmeyr and Forrest [24] described the first immune inspired architecture for intrusion detection in wired networks. Their work has a strong methodological character. Their interpretation of costimulation relies on a human operator to decide whether a detected anomaly is the consequence of an intrusion.

The approach of Sarafijanović and Le Boudec [41, 33] avoids the necessity to invoke a human response after a possible misbehavior in mobile ad hoc network was detected. As mentioned above, this was done by coupling negative selection with a transport layer protocol. The main methodological drawback of their approach is that they combined two immune inspired mechanisms, negative selection and co-stimulation, without having first understood the efficiency of each approach in isolation. As a consequence, the performance of their approach did not afford for a clear interpretation. The related insights on the efficiency of negative selection were published only recently by Elberfeld and Textor [18].

Even though, co-stimulation was cited as a possible approach for gaining control over false positives in intrusion/misbehavior detection [24, 41, 33], this mechanism received only very limited attention. For this reason, the process of translating co-stimulation into a computational approach required from our side a substantial amount of conceptual effort.

With respect to the above reviewed work, we followed two objectives. We aimed at benefiting from the capability of co-stimulation to suppress false positives. At the same time, due to energy efficiency concerns, we also aimed at minimizing any overhearing in promiscuous mode. When evaluating the energy efficiency of our approach, we compare against watchdog based misbehavior detection. Watchdog based misbehavior detection offers solid detection performance in scenarios where the ambition is to identify a specific node executing data dropping, data delaying or a similar misbehavior type [17]. Such a solid detection performance makes it straightforward to understand the trade-off between detection performance and energy efficiency, i.e. to understand whether any decrease in detection performance was matched by an increase in energy efficiency.

The main discerning factor between our approach and the approaches reviewed above is that our approach develops a conceptual framework for misbehavior detection in ad hoc wireless networks. The reviewed approaches concentrate on achieving an acceptable misbehavior detection performance, whereas we attempt to provide a means that offers the possibility to influence the detection rate, the false positives rate and the energy efficiency in a controlled manner.

4 Co-stimulation and Priming: A Computational Interpretation

Before discussing the different approaches which take advantage of co-stimulation and priming, we define and discuss co-stimulation and priming in a more general way, so that a computational interpretation with focus on network security is straightforward [11, 12, 15].

Definition Co-stimulation is an auxiliary signal indicating that a misbehavior causes damage to the system.

Definition Priming is a process for specifying the alertness levels of a system with respect to a misbehavior class.

Notice that co-stimulation, with respect to the above definition, implies that at least two detection mechanisms are applied when detecting misbehavior. Clearly, co-stimulation can be implemented in several ways. Our approach aims not only at improving the misbehavior detection precision by using two distinct detection mechanisms, but also by requiring that these two detection mechanisms have very different energy efficiency. In our case, the less energy efficient type of classification is only used upon receiving a co-stimulatory signal from a more energy efficient type of classification.

The co-stimulation based misbehavior classification approach, that we next discuss, resembles the "cascading classifiers" of Kaynak and Alpaydin introduced in [30]. Their approach is based on

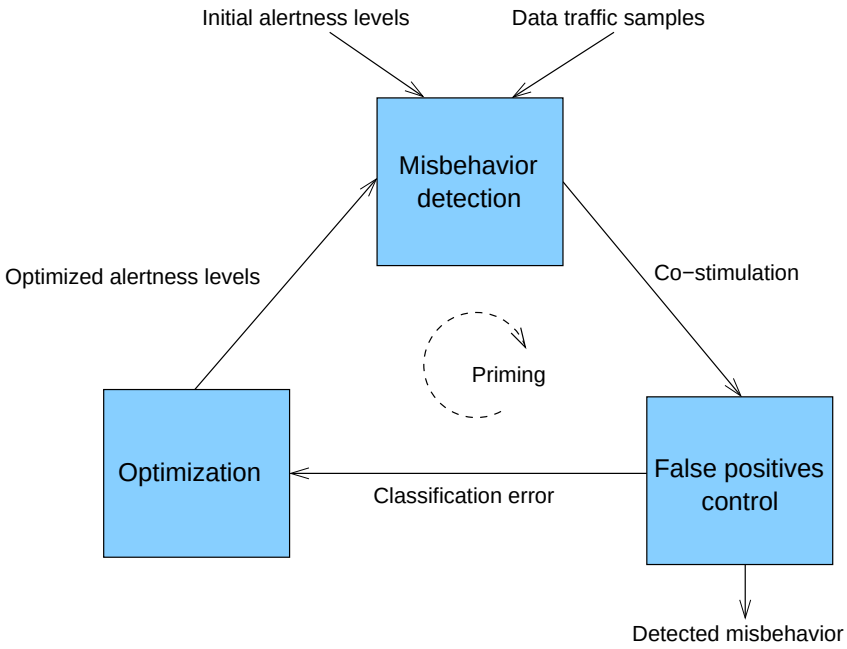


Figure 2: Misbehavior detection architecture.

a sequential application of several classifiers such that *"at the next stage, using a costlier classifier, we build a more complex rule to cover those uncovered patterns of the previous stage"*. Our BIS inspired approach can be seen as an instance of cascading classification. It is however not the complexity of classification rules that is increased at each step but the energy cost connected with observing additional states and events necessary for a more precise reasoning about misbehavior.

Cascading classifiers were empirically studied by Gama and Brazdil in [22]. They combined several types of classifiers: Bayes classifier, C4.5 and linear discriminant function. Their focal point was to investigate whether cascading classification could offer some classification performance improvement over classification using a single classifier. They tested this approach on several standard datasets from the UCI Repository [3]. Their results showed that cascading classifiers offer some improvement in classification precision, however, this improvement was not significant enough to initiate additional follow-up research. Their results also did not point out that cascading classifiers could offer some advantage in distributed environments such as ad hoc networks.

When translating priming into a computational approach, we decided to couple several basic quality of service indicators, such as the number of data packets dropped or delayed, with a mechanism allowing for an efficient detection of their critical levels. Such critical levels can be set by the human operator with the intent to monitor the network. Reaching or even surpassing such critical levels can be interpreted as being undesired, i.e. as misbehavior. Since continuous monitoring of a node by its neighbors can be considered energy inefficient, we decided to take advantage of co-stimulation, most notably of its ability to suppress false positives. This property allowed us to design a simple energy efficient detection mechanism that relies on co-stimulation for false positives control.

When interpreting co-stimulation and priming, our central goal was to mimic the capability of two-way communication within the BIS. When implementing such a two-way communication, the strategy discussed in Section 2.3 provided helpful clues. Since our computational interpretation is

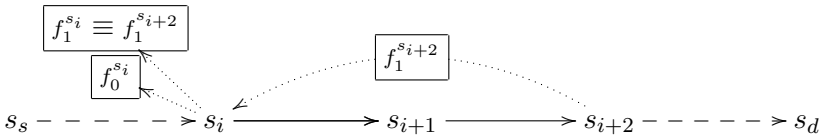


Figure 3: Data traffic measurement model.

based on two distinct detection mechanisms, we tended towards mapping one detection mechanism to the adaptive immune system and the other one to the innate immune system. We considered the detection mechanism that relies on continuous monitoring in promiscuous mode to be more similar to the innate immune system. The other detection mechanism was, due to its greater flexibility conveyed by the underlying machine learning algorithms, assigned to represent the adaptive immune system. In retrospection, we still consider such a mapping useful, since it helped to develop a conceptual picture of our misbehavior detection architecture.

Co-stimulation and priming are key mechanisms of our misbehavior detection architecture shown in Fig. 2. The purpose of the misbehavior detection module is to provide an initial separation of data traffic samples stemming from either normal behavior or misbehavior. This module can consist of several detection algorithm, for example, supervised or unsupervised machine learning algorithms. The false positives control module is only applied, if a suspicious data traffic sample was detected. A classification error feedback allows for a computation of optimized alertness levels, these in turn allow for a more efficient separation of data traffic samples.

5 Energy Efficient Co-stimulation Approach

Our first goal was to introduce the capability of co-stimulation into our misbehavior detection architecture [17]. Since the purpose of the misbehavior detection module and the false positives rate control module is different, it was necessary to identify several feature sets, each offering a different perspective on a node's behavior.

Our co-stimulation approach is based on the local connection-oriented traffic measurement model depicted in Fig. 3. Let s_{i+1} be a misbehaving node and let s_i and s_{i+2} be its neighbors. These nodes are lying on a data connection from s_s to s_d. Let us further assume that each node can measure local traffic using two distinct feature sets f_0 and f_1. Let us denote their numerical *instances* as $\hat{f}_0$ and $\hat{f}_1$.

After the node s_{i+2} computes $\hat{f}_1$, this feature set instance is proliferated in the upstream connection direction (towards connection source s_s). The node s_i then compares its own $\hat{f}_1$ sample with the sample received from s_{i+2}. Based on this, a behavior classification with respect to the misbehaving node s_{i+1} is done. If s_i classifies s_{i+1} as misbehaving, it will additionally use the feature set f_0 for classification. If misbehavior is detected again, s_{i+1} is finally classified as misbehaving.

To simplify the notation let us define: $\mathcal{F}_1 = f_1^{s_i} \cup f_1^{s_{i+2}}$. Similarly, assuming that $\hat{f}_0$ and $\hat{f}_1$ are represented as vectors, we define $\hat{\mathcal{F}}_1 = \hat{f}_1^{s_i} \circ \hat{f}_1^{s_{i+2}}$, where $\circ$ is the operator of vector concatenation. With respect to this notation, our approach to co-stimulation can be succinctly expressed as:

$$\mathcal{F}_1 \xrightarrow{co-stimulation} f_0 \tag{1}$$

The next task in implementing co-stimulation was to populate the feature sets $\mathcal{F}_1$ and f_0 with suitable features. When choosing the features, we aimed at the following objectives: i) the relation-

ship between the feature sets $\mathcal{F}_1$ and f_0 should be well defined, ii) computation of $\mathcal{F}_1$ should be energy more efficient than computation of f_0 and iii) classification based on f_0 should offer higher precision than classification precision based on $\mathcal{F}_1$.

To simplify the search for suitable features, we decided that the feature set f_0 would be dominated by watchdog features, i.e. features computed in promiscuous mode. Since watchdog based features benefit from explicit overhearing of each data packet, the classification precision that they convey is very high. Since promiscuous mode prevents the wireless interface from entering idle state, operation in promiscuous mode can be considered energy inefficient. According to [19], power consumption in idle and receive states is about 12-20 higher than in sleep mode.

Including watchdog features, we identified 24 candidate features. These features were divided into three groups with respect to their energy requirements and protocol assumptions. A wrapper approach [31] was used to identify features with a *statistically significant contribution* to the detection of the three considered misbehavior types. More specifically, each of the 24 features was tested whether it significantly decreases the classification error with respect to these misbehavior types. The feature that decreased the classification error the most was chosen for the final feature set. The process was then repeated with the remaining features until there was no other feature that could significantly decrease the classification error. This was coupled with cross-validation in order to obtain a robust estimate of the classification error in each round.

The output from the above procedure were three feature sets, each with a different degree of energy efficiency. Since the statistical significance of each feature within a feature set was known, it allowed us to understand how these three feature sets are related. Given this knowledge, we decided to concentrate only on the least and most energy efficient feature set that formed f_0 and f_1, respectively. The wrapper based approach showed that in addition to watchdog features, the f_0 feature set must also include several topology related features in order to facilitate detection of wormholes.

5.1 Definitions

We evaluate misbehavior classification performance in terms of detection rate and false positives (FP) rate. We assume that a classifier $\mathcal{K}(f)$ computed by a learning algorithm is used in the classification process, where $f \in \{f_0, \mathcal{F}_1\}$ is the feature set used to reason about the behavior of a node. The classifier $\mathcal{K}(f)$ is then used to classify the objects $\Omega = \{o_1, ..., o_p\}$, where p is the number of these objects. The two measures are then computed as follows:

$$det.\ rate^{\Omega}_{c_j}(\mathcal{K}(f)) = \frac{c_{c_j}}{n_{c_j}} \times 100.0\% \qquad FP\ rate^{\Omega}_{c_j}(\mathcal{K}(f)) = \frac{FP_{c_j}}{FP_{c_j} + c_{c_j}} \times 100.0\% \quad (2)$$

where c_j is the j-th class. n_{c_j} is the number of objects labeled with the class c_j; note that $n_{c_j} > 0$ in all our experiments. c_{c_j} is the number of objects that were correctly classified by the learning algorithm as belonging to the class c_j. FP_{c_j} is the number of objects incorrectly predicted as belonging to c_j.

The overall misclassification rate is evaluated by means of the classification error:

$$\epsilon^{\Omega}(\mathcal{K}(f)) = \frac{\sum_{c_j} FP_{c_j}}{\sum_{c_j} n_{c_j}} \times 100.0\% \quad (3)$$

Since the focus of our work is on the features being used in the classification process, rather than on the type of learning algorithm, we simplify the notation by using $det.\ rate^{\Omega}_{c_j}(f)$, $FP\ rate^{\Omega}_{c_j}(f)$ and $\epsilon^{\Omega}(f)$ instead of $FP\ rate^{\Omega}_{c_j}(\mathcal{K}(f))$, $FP\ rate^{\Omega}_{c_j}(\mathcal{K}(f))$ and $\epsilon^{\Omega}(\mathcal{K}(f))$, respectively. We also omit the superscript Ω, whenever there is a single set of objects to be classified.

5.2 Properties of the f_0 and $\mathcal{F}_1$ Feature Sets

Our experimental results [17] show that with respect to the classification error ϵ and the energy cost ξ, the properties of $\mathcal{F}_1$ and f_0 can be summarized as follows:

1. For $win.\ size \to 0$, it holds:

$$\lim_{win.\ size \to 0} \epsilon(\mathcal{F}_1) \approx \epsilon(f_0) \tag{4}$$

   This characterizes the relationship between watchdog and $\mathcal{F}_1$ based misbehavior detection. It points out that instead of observing *each* data packet's delivery in promiscuous mode by the node s_i, it can be equally well done in a cooperative way by s_i and s_{i+2}, if $win.\ size \to 0$.

2. For $win.\ size \gg 0$, it holds:

$$\epsilon(\mathcal{F}_1) > \epsilon(f_0) \tag{5}$$

$$\xi(\mathcal{F}_1) < \xi(f_0) \tag{6}$$

   The measure of energy cost ξ includes feature computation costs as well as all induced communication costs. The communication costs for f_0 are related to overhearing in promiscuous mode. The communication costs for $\mathcal{F}_1$ are related to the necessity to transmit $\hat{f}_1^{s_{i+2}}$ over two hops to s_i.

   The first inequality reflects the fact that overhearing each data packet in promiscuous mode gives a better base for classification than a classification based on features computed by two distinct nodes and aggregated over a time window. The other inequality reflects the fact that operation in promiscuous mode is inherently energy inefficient.

The features in the sets f_0 and $\mathcal{F}_1$ can be computed without much computational overhead. Our misbehavior detection results could have been better if a more complex Fourier or wavelets analysis of the packet stream had been done. As the results by Barford et al. point out [5], this could lead to good anomaly detection rates.

5.3 Co-stimulation and its Misbehavior Detection Efficiency

Let Ω be the set of all vectors subject to misbehavior classification. The vectors in Ω represent the behavior of monitored nodes (or in general, any other type of objects). In our case, a vector $v_m^{s_i} \in \Omega$ has two components: $v_m^{s_i} = \hat{\mathcal{F}}_1^{s_i} \circ \hat{f}_0^{s_i}$, where m identifies the time window in which $\hat{\mathcal{F}}_1^{s_i}$ was computed. Dependent upon the evaluation of $\hat{\mathcal{F}}_1^{s_i}$, the computation of $\hat{f}_0^{s_i}$ is started in the time period following the time window m. $\mathcal{F}_1$ and f_0 based classification of $v_m^{s_i}$ is done using the $\hat{\mathcal{F}}_1^{s_i}$ component and the $\hat{f}_0^{s_i}$ component, respectively.

Let $\Omega_{\mathcal{F}_1} \subseteq \Omega$ be the subset of vectors that were marked as representing suspicious behavior, after the $\mathcal{F}_1$ based classification was done. Let us first assume that $\epsilon^{\Omega}(f_0) = 0$, where $\epsilon^{\Omega}(f_0)$ is the classification error of f_0 based classification applied to the vector set Ω. If f_0 based classification is applied to $\Omega_{\mathcal{F}_1}$, it clearly holds:

$$\epsilon^{\Omega_{\mathcal{F}_1}}(f_0) = 0 \tag{7}$$

This implies, for the final FP rate and a given misbehavior class c_j, after co-stimulation is applied, it holds:

$$FP\ rate_{c_j}^{\Omega}(\mathcal{F}_1 \to f_0) = 0 \tag{8}$$

In other words, the conditional application of f_0 based classification removes all misclassified vectors. Furthermore, the final detection rate for the given class is determined by the detection rate after the $\mathcal{F}_1$ based classification:

$$det.\ rate_{c_j}^{\Omega}(\mathcal{F}_1 \rightarrow f_0) = det.\ rate_{c_j}^{\Omega}(\mathcal{F}_1) \tag{9}$$

Since to achieve $\epsilon^{\Omega}(f_0) = 0$ may not be possible, Eqs. 8 and 9 translate for $\epsilon^{\Omega}(f_0) \neq 0$ to:

$$FP\ rate_{c_j}^{\Omega}(\mathcal{F}_1 \rightarrow f_0) = FP\ rate_{c_j}^{\Omega}(f_0) \tag{10}$$

$$det.\ rate_{c_j}^{\Omega}(\mathcal{F}_1 \rightarrow f_0) \leq det.\ rate_{c_j}^{\Omega}(\mathcal{F}_1) \tag{11}$$

This means, the final FP rate is only depending on the efficiency of f_0 based classification. Eq. 10 is based on the assumption that when classifying Ω and $\Omega_{\mathcal{F}_1}$ using f_0 based classification, the following holds:

$$FP\ rate_{c_j}^{\Omega_{\mathcal{F}_1}}(f_0) = FP\ rate_{c_j}^{\Omega}(f_0) \tag{12}$$

To what extent such an assumption is reasonable, was one of the goals of our experimental analysis. Notice that co-stimulation, being a classification approach where f_0 based classification is applied conditionally, can result in two groups of vectors: i) vectors subject only to $\mathcal{F}_1$ based classification and ii) vectors subject to both $\mathcal{F}_1$ and f_0 based classification. This asymmetric situation provides the basis for energy efficient misbehavior detection.

When detecting misbehavior, f_0 based detection will only get used, if (i) $\mathcal{F}_1$ based classification detects a true positive or (ii) $\mathcal{F}_1$ based classification outputs a false positive. Since it is reasonable to assume that any ad hoc network will work reliably, most of the time, our focus stays on the energy cost analysis for misbehavior free ad hoc networks. With respect to the above said, the overall energy cost of co-stimulation ξ in a misbehavior free ad hoc network can be modeled as:

$$\xi = \xi(\mathcal{F}_1) + g(FP\ rate_{c_j}^{\Omega}(\mathcal{F}_1)) \times \xi(f_0) \tag{13}$$

where $g(.)$ is the rate at which f_0 based classification is mistakenly applied in a misbehavior free ad hoc network. $\xi(f_0)$ is dominated by the cost of overhearing in promiscuous mode. The options for decreasing this cost are limited and therefore it can be considered fixed. When applying $\mathcal{F}_1$ based classification that allows for a trade-off between its classification precision and $\xi(\mathcal{F}_1)$, and it holds that $\xi(f_0) \gg \xi(\mathcal{F}_1)$, then the effects of such a trade-off are further amplified. Increasing the precision of $\mathcal{F}_1$ based classification can thus lead to a substantial decrease of co-stimulation cost ξ, and vice versa. Additionally, if we assume that Eq. 12 holds, then this trade-off translates for a given misbehavior class c_j into a trade-off between detection rate and energy efficiency, while keeping the FP rate unchanged.

6 Towards Priming against Misbehavior

Co-stimulation, as presented in the previous sections, requires that two distinct $\mathcal{F}_1$ and f_0 based classifiers get computed. This implies that two sets of $\hat{\mathcal{F}}_1$ and $\hat{f}_0$ feature vectors, labeled with the classes under consideration, must be available for training. Next, we discuss how this can be avoided.

In this section, we delve into the mechanisms of error propagation [11, 12]. Error propagation is the opposite of co-stimulation, i.e. in a short form it can be expressed as:

$$\mathcal{F}_1 \xleftarrow{error\ propagation} f_0 \tag{14}$$

Eq. 4 states that for $win.\ size \rightarrow 0$, the $\mathcal{F}_1$ and f_0 based classification approaches offer the same classification accuracy. This implies, if $\epsilon^{\Omega}(f_0) = 0$, for the $\hat{\mathcal{F}}_1$ and $\hat{f}_0$ feature vectors stemming from the same time window, the classification outcome will be (nearly) the same. This motivates the following strategy: compute $\hat{\mathcal{F}}_1$ and $\hat{f}_0$ feature vectors, classify the $\hat{f}_0$ feature vector according to a predefined threshold, and then, label the $\hat{\mathcal{F}}_1$ feature vector with the same label as the $\hat{f}_0$ based vector. This allows us to build a labeled $\hat{\mathcal{F}}_1$ based feature vector set necessary for the computation of an $\mathcal{F}_1$ based classifier.

The application of such thresholds is hereafter referred to as *priming*. The general goal of priming is to introduce a well-defined level of consistency, when detecting misbehavior in an ad hoc network. With respect to the above said, we define misbehavior as a violation of the priming thresholds $\mathcal{P} = \{p_1, p_2, ..., p_l\}$, where l is the number of priming thresholds. A priming threshold can be for example the maximum allowed data packet loss at a node or the maximum allowed data packet processing delay at a node. If any priming threshold is violated, the corresponding $\hat{\mathcal{F}}_1$ and $\hat{f}_0$ feature vectors will be labeled as representing misbehavior.

In order to achieve a good level of energy efficiency, it is desirable to apply $win.\ size \gg 0$. Since however a larger time window size introduces a loss in misbehavior detection precision, error propagation must be followed by co-stimulation:

$$\mathcal{F}_1 \underset{error\ propagation}{\overset{co-stimulation}{\rightleftarrows}} f_0 \begin{array}{l} \nearrow \epsilon \\ \nwarrow \mathcal{P} \end{array} \tag{15}$$

Co-stimulation is achieved by computing a fresh $\hat{f}_0$ feature vector and comparing it with $\mathcal{P}$. This approach is also schematically depicted in Fig. 5(a).

Error propagation and co-stimulation is executed within the same node; see Fig. 4. Notice that some nodes in the example network, for the shown data flows, are unable to compute $\hat{\mathcal{F}}_1$ since they do not have any two-hop neighbor s_{i+2}. On the other hand, several nodes receive multiple $\hat{f}_1^{s_{i+2}}$, e.g. s_1 from s_3 and s_5.

Let us now formulate the effects of these two phases in more detail. Let us assume that $win.\ size \gg 0$. For simplicity, let us also assume that $\epsilon^{\Omega}(f_0) = 0$.

1. *After the error propagation phase:* it holds that $\epsilon^{\Omega}(\mathcal{F}_1) > 0$, i.e. the precision of $\mathcal{F}_1$ based classification is for $win.\ size \gg 0$ lower than the precision of f_0 based classification. This is a direct consequence of the property expressed in Eq. 5. $\epsilon^{\Omega}(\mathcal{F}_1) > 0$ implies that there exists a class c_j for which at least one of the following holds:

$$det.\ rate^{\Omega}_{c_j}(\mathcal{F}_1) = a < 100\%$$

$$FP\ rate^{\Omega}_{c_j}(\mathcal{F}_1) = b > 0\%$$

2. *After the co-stimulation phase:* co-stimulation applied after error propagation decreases the FP rate of the class c_j to zero, while keeping the detection rate unchanged:

$$det.\ rate^{\Omega}_{c_j}(\mathcal{F}_1 \rightarrow f_0) = a$$

$$FP\ rate^{\Omega}_{c_j}(\mathcal{F}_1 \rightarrow f_0) = 0\%$$

This is a direct consequence of the properties expressed in Eqs. 8 and 9.

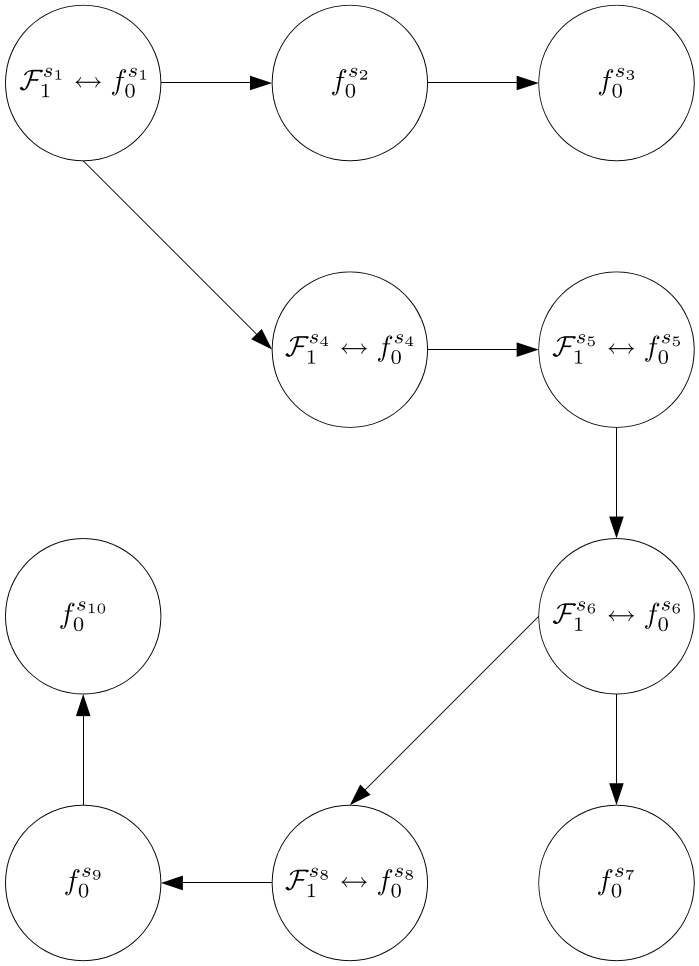
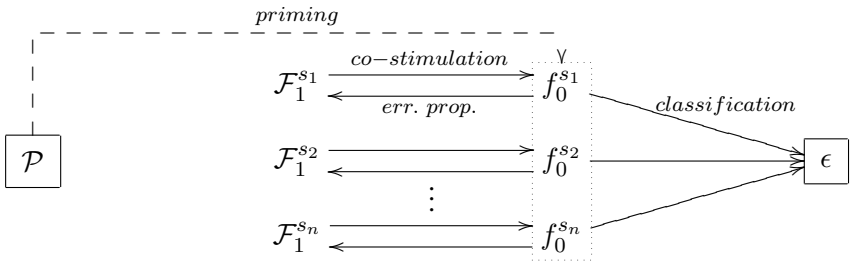


Figure 4: A 10-node ad hoc network with priming. s_1 is the only data flow source node. s_3, s_7 and s_{10} are sink nodes for three distinct data flows.

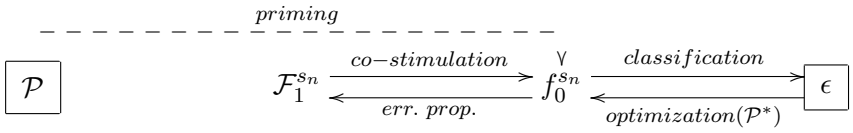
Since to achieve $\epsilon^{\Omega}(f_0) = 0$ may not be feasible, Eqs. 10 and 11 apply. This means that the final FP rate for a class c_j is determined by $FP\ rate_{c_j}^{\Omega_{\mathcal{F}_1}}(f_0)$. Similarly, the final detection rate has an upper bound equal to $det.\ rate_{c_j}^{\Omega}(\mathcal{F}_1)$. Notice that $win.\ size$ also influences the time delay for detecting a misbehavior, i.e. it should reflect the requirements prescribed for the given misbehavior detection system.

6.1 Error Propagation Algorithm with Optimization

The error propagation algorithm can be extended with an optimization phase; see Fig. 5(b). With respect to the classification outcome, the individual priming threshold values for each node can be optimized, i.e. the classification error can be minimized. This can be done by a repeated application of the error propagation and co-stimulation phases, while adjusting the priming thresholds for each node, until a termination condition is met; see Fig. 5. This approach can be described as priming with influences of e.g. noise being considered locally. More formally, new optimized priming thresholds $\mathcal{P}^* = \{p_1^*, p_2^*, ..., p_l^*\}$ for each node s_i will be found, so that the final classification error for each node s_i is minimized. The repeated application of the error propagation and co-



(a) An approach with error propagation and co-stimulation.



(b) An approach extended with optimization.

Figure 5: Error propagation algorithm.

stimulation phases is inspired by the feedback loop between the innate and adaptive immune systems as discussed in Section 2.

Our extended approach bears a certain similarity to the *backpropagation algorithm* for artificial neural networks [2]. The backpropagation algorithm takes advantage of two steps, feed-forward and error backpropagation. These two steps are repeated in order to minimize an error function. A notable difference between our extended approach and the backpropagation algorithm is, co-stimulation phase is designed to revert the FP rate to the levels before any error propagation was done. Since the FP rate is expected to stay unchanged, the search for suitable values for priming thresholds is less complex.

7 Co-stimulation and Priming: Experimental Results

A performance evaluation of co-stimulation and priming was done using the JiST/SWANS network simulator [6]. JiST/SWANS was chosen over other alternative such as ns2 due to its better scalability with the experiment size, both in terms of event processing speed and memory overhead [6]. JiST/SWANS is about two orders of magnitude faster than ns2 with Tcl.

JiST/SWANS does not offer any ready-to-use misbehavior implementation. We took advantage of AIS-Lib, a library for performance evaluation of AIS based detection systems for ad hoc wireless networks, introduced by Drozda et al. in [16]. AIS-Lib implements several types of misbehavior.

The experiments were based on a network with 1718 nodes. The number of connections were 50. The length of a connection was 7 hops. Each connection expired after 15-20 minutes and was substituted with a new connection with randomly chosen source and destination nodes. Three different types of misbehavior were considered: data packet dropping, data packet delaying and wormholes. Wormholes are private (out-of-band) links between one or several pairs of nodes. They

are added by an attacker in order to attract data traffic into them to gain control over packet routing and other network operations.

The classification based on $\mathcal{F}_1$ or f_0 was done using a decision tree classifier. The reason for using a decision tree classifier are the results that we report in [7]. Therein we compared several learning algorithms such as decision tree classifier, naive Bayes classifier, support vector machines, neural networks and negative selection with respect to their applicability in misbehavior detection scenarios. We concluded that for the considered scenarios, the classification results based on decision tree classifier dominate the classification results obtained by the other learning algorithms. Additionally, since decision tree classifier is less complex than the other considered classification approaches, it is more suitable for ad hoc networks with limited (battery) resources.

When estimating the energy efficiency of the co-stimulation and priming approaches, we considered several wireless devices: TI CC2420, Wistron CM9 miniPCI card, Ubiquiti XR2 card and Lucent 2Mbps IEEE802.11 wireless card. For the reasons stated in Sect. 3, we compared the energy consumption of our approaches with the energy consumption of misbehavior detection done in promiscuous mode. The parameters used in our simulations are summarized in Figure 6.

The results for co-stimulation reported in [17] show, for a scenario with $win.\ size = 500s$ and with respect to the general misbehavior class mis (in bimodal normal-misbehavior classification), the following:

$$FP\ rate^{\Omega}_{mis}(\mathcal{F}_1 \rightarrow f_0) = 1.67 \pm 2.59\% \ \cong\ FP\ rate^{\Omega}_{mis}(f_0) = 1.77 \pm 0.66\% \tag{16}$$

$$det.\ rate^{\Omega}_{mis}(\mathcal{F}_1 \rightarrow f_0) = 78.89 \pm 1.71\% \ \cong\ det.\ rate^{\Omega}_{mis}(\mathcal{F}_1) = 76.40 \pm 2.53\% \tag{17}$$

The achieved $FP\ rate^{\Omega}_{mis}(\mathcal{F}_1)$ was $15.09 \pm 2.27\%$. Comparing the results for the $\mathcal{F}_1$ and $\mathcal{F}_1 \rightarrow f_0$ based classification, it is clear that $\mathcal{F}_1 \rightarrow f_0$ offers a detection rate comparable to $\mathcal{F}_1$, however, a much lower false positives rate.

An example of the detection performance and the energy efficiency is shown in Table 1, where $win.\ size$ is the time window size applied by $\mathcal{F}_1$ based classification, $FP\ rate$ is the false positives rate achieved after $\mathcal{F}_1$ based classification is done, ξ is the total energy consumption of co-stimulation, γ is the reduction in energy consumption achieved by co-stimulation compared to overhearing in promiscuous mode, det. rate is the detection of co-stimulation for a given misbehavior class and FP rate is false postives rate achieved by co-stimulation for the same misbehavior class. It can be seen that co-stimulation establishes a trade-off between detection rate and energy efficiency, while keeping the false positives rate almost unchanged.

The results achieved applying priming are in many respects similar to those obtained by co-stimulation. Again, a very good reduction in energy consumption could be achieved. The peak value was 97.31%. Additionally to several misbehavior detection scenarios, we also evaluated the case when priming is used to check whether network operational conditions are within limits prescribed by the priming thresholds. We could achieve the final false positives rate equal to 0%.

8 Summary of Results and Research Challenges

Even though, a large amount of our research effort went into establishing co-stimulation and priming as computational approaches, we also delved into other research areas. Most notably, we investigated the suitability of negative selection for misbehavior detection. In order to facilitate the complex simulation based experiments executed when evaluating co-stimulation and priming in realistic settings, we designed AIS-Lib, a library for performance evaluation of AIS based misbehavior approaches. Additionally, we investigated the suitability of a locality sensitive hashing scheme for

1. **Classification algorithm:** $\mathcal{F}_1$ based and f_0 based classification was implemented as decision tree. To decide whether a node within the decision tree should be further split (impurity measure), we used the information gain measure [2]. We used the decision tree implementation from the Rapidminer tool [39]. k-fold cross-validation with $k = 20$ was used to estimate classification performance.
2. **Network topology:** Snapshot of movement modeled by random waypoint mobility model i.e. it is a static network. There were 1,718 nodes. The area was a square of 2,900m×2,950m. The transmission range of transceivers was 100 meters.
3. **Misbehavior:** data packet dropping, data packet delaying and wormholes. In data packet dropping and data packet delaying scenarios, there were 20-30 nodes concurrently exercising misbehavior.
    - Co-stimulation experiments: 30% data packets dropped, 30% data packets delayed 100ms, 20 wormholes. Data packet dropped or delayed were chosen uniformly at random.
    - Priming experiments: 30% or 10% data packets dropped, 30% data packets delayed 100ms or 20ms, 20 wormholes. Data packet dropped or delayed were chosen uniformly at random.
4. **Time window size:** 50, 100, 250 and 500 seconds.
5. **Connections:** 50 CBR (constant bit rate) connections. Each connection expires after 15-20 minutes and is then substituted with a new source-destination pair. The connection length is 7 hops.
6. **Injection rate:** 0.5 packet/second. **Packet size:** 68 bytes. **MAC protocol**: IEEE 802.11 DCF. **Routing protocol**: AODV. **Other parameters:** Channel frequency: 2.4 GHz. Network protocol: IPv4. Connection type: UDP. Data rate: co-stimulation: 2Mbit/s; priming: 54Mbit/s and 250kbit/s.
7. The number of independent simulation runs for normal traffic and each misbehavior type: 20. The simulation time was 4 hours.
8. **Simulator used:** JiST/SWANS; hardware used: 20× Linux (SuSE 10.0) PC with 2GB RAM and Pentium 4 3GHz microprocessor.
9. Wireless devices considered when evaluating energy efficiency:
    - co-stimulation: Lucent 2Mbps IEEE802.11 card.
    - priming: TI CC2420, Wistron CM9, Ubiquiti XR2.

Figure 6: Parameters used in the experiments.

$win.\ size$ $[s]$	$FP\ rate$ [%] $\mathcal{K}(\mathcal{F}_1)$	ξ [mJ]	γ [%]	Det.rate [%] $\mathcal{K}(\mathcal{F}_1) \rightarrow \mathcal{K}(f_0)$	FP rate [%] $\mathcal{K}(\mathcal{F}_1) \rightarrow \mathcal{K}(f_0)$
50	3.55	23.29	82.73	93.42	0.98
100	4.65	12.39	90.81	90.00	1.26
250	9.28	6.20	95.40	84.70	1.28
500	15.09	3.89	97.12	78.89	1.67

Table 1: Co-stimulation: energy consumption and misbehavior detection performance.

intrusion detection. Next, we summarize the results related to these efforts. However before doing so, we provide a short summary of results for the co-stimulation and priming approaches. We also discuss several challenges [11, 12] related to these approaches:

1. *Understanding watchdog based misbehavior detection:* Since watchdog based misbehavior detection (f_0) is energy inefficient, it was mandatory to design a different type of misbehavior detection ($\mathcal{F}_1$). The latter misbehavior detection approach is based on traffic measurements by neighboring nodes. Our experimental results show that the properties of f_0 and $\mathcal{F}_1$ based classification can be expressed as in Eqs. 4, 5 and 6. This means, not only is computation of $\mathcal{F}_1$ feature set more energy efficient, but for $win.\ size \rightarrow 0$, it can substitute watchdog based misbehavior detection. Additionally, the detection precision of $\mathcal{F}_1$ based misbehavior detection decreases as $win.\ size$ increases. This is a key property since it allows us to substitute watchdog based misbehavior detection with its $\mathcal{F}_1$ based counterpart.

2. *Co-stimulation vs priming:* Co-stimulation requires that two distinct classifiers get computed. This necessitates that two sets of labeled feature vectors are available. Priming does not require any set of labeled feature vectors as input. Instead it relies on a set of priming thresholds that define the limits for network operational conditions. The focus of priming is different from the focus of co-stimulation. The goal of co-stimulation is to provide a means for energy efficient misbehavior detection, whereas the goal of priming is to provide a means for energy efficient detection of priming threshold violation. A violation of priming thresholds can either be interpreted as misbehavior or simply as an event which requires logging and analysis.

3. *Single detection system instance per node:* We only apply a single instance of the detection mechanism per node. This is different from the approach used by Sarafijanović and Le Boudec [41], where one instance was used to reason about the behavior of each neighbor. Under our approach, $\mathcal{F}_1$ based classification detects that one of the neighbors is misbehaving. The identity of the misbehaving node is later determined by f_0, i.e. by watchdog features that observe whether data packets get correctly forwarded by a specific node.

4. *Independence of detection rate and FP rate:* Co-stimulation allows that the final detection rate and false positives rate are influenced by two distinct mechanisms: $\mathcal{F}_1$ and f_0 based classification. This fact is documented in Eqs. 10 and 11. This property allows for tuning the detection system with respect to these two measures in isolation. We showed that in scenarios, where the only goal is to detect whether a specific performance threshold was violated, it is feasible to achieve $\epsilon^{\Omega}(f_0) = 0$. For this reason, with respect to Eq. 8, it was possible to decrease the final false positives rate to zero.

5. *Energy efficiency:* The costly f_0 (watchdog) based classification is only applied if the energy more efficient $\mathcal{F}_1$ classification detects a misbehavior. Under our experimental setup, the peak energy consumption reduction was 92.7% compared to the energy cost of exclusive f_0 based classification. Co-stimulation allows us to choose a *trade-off* between detection performance and energy efficiency. A higher level of energy efficiency can be achieved by increasing the time window size. This results in a lower detection rate. Increasing energy efficiency has however only a limited influence on the FP rate.

6. *Source of co-stimulation:* The co-stimulation approach due to Sarafijanović and Le Boudec [41] benefits from the information about data packet delivery provided by TCP (Transmission Control Protocol). If a data packet is not delivered, then the connection source

does not receive the corresponding packet acknowledgment. This can be then used as a co-stimulation for another form of data packet loss detection. The transport layer can thus serve as a potent source of co-stimulation for any lower layer detection mechanisms. This is however not always possible, since e.g. sensor networks are expected to operate with a reduced set of transport layer services. The reason for such a reduction is also the energy consumption related to transport layer services. The goal to increase energy efficiency of misbehavior detection was our motivation for investigating a distinctly different form of co-stimulation.

7. *Co-stimulation avoiding watchdog (f_0) features:* Watchdog features can only be computed, if omnidirectional antennas are being used. Using a directional antenna by s_{i+1} could preclude any packet overhearing by s_i. A similar effect can be observed, if s_{i+1} is capable of dynamic radio radius adjustment. As Eq. 4 suggests, watchdog features can be substituted by $\mathcal{F}_1$ based features, if $win.\ size \to 0$. This implies, a co-stimulation based on $\mathcal{F}_1$ with a small window size could be a viable substitute for an f_0 based co-stimulation. For example, an $\mathcal{F}_1$ based classification with a 25-second time window could be used instead:

$$\mathcal{F}_1(500s) \xrightarrow{co-stimulation} \mathcal{F}_1(25s) \tag{18}$$

This sort of co-stimulation would require an adaptive approach for requesting an $\hat{f}_1^{s_{i+2}}$ sample based on a smaller time window size. It also limits the option of *negative co-stimulation* [12]. Negative co-stimulation is executed by s_i, if s_{i+1} decides not to cooperate in forwarding $\hat{f}_1^{s_{i+2}}$. Under such circumstances, s_i will switch, after a time-out, to the f_0 feature set computation. Unlike $\mathcal{F}_1$ feature set computation, f_0 feature set computation can be applied by s_i at any time, unimpeded by the willingness of its neighbors to cooperate.

We next discuss a few challenges related to co-stimulation or priming:

1. *Convergence of the extended procedure with optimization:* In Section 6.1 we discussed priming extended with an optimization phase. This optimization phase allows for finding the optimal thresholds for $\mathcal{P}^*$. It is currently unclear, which optimization approach would suit best this purpose, i.e. delivering the best performance with respect to the optimization convergence.

2. *Minimizing the rate of undecided vectors:* Undecided vectors are the vectors that were after $\mathcal{F}_1$ based classification marked as representing a misbehavior, but this could not be confirmed by f_0 based classification. An especially severe case happens, if $\mathcal{F}_1$ based classification correctly classifies a vector as representing a misbehavior, but this classification result will get incorrectly rejected by f_0 based classification. One of the possibilities how the decrease the rate of undecided vectors is to present f_0 based classification with vectors that this classification approach can correctly classify with a high success rate. This implies that $\epsilon^{\Omega_{\mathcal{F}_1}}(f_0) \ll \epsilon^{\Omega}(f_0)$, i.e. the task to classify the vectors in $\Omega_{\mathcal{F}_1}$ is much less complex than the task to classify the vectors in Ω. The challenge is to tune $\mathcal{F}_1$ based classification in such a way that it outputs a vector set $\Omega_{\mathcal{F}_1}$ such that $\epsilon^{\Omega_{\mathcal{F}_1}}(f_0) \to 0$.

At last, we discuss several results related to performance evaluation of misbehavior detection. We also outline our preliminary on applying locality preserving hashing in intrusion detection.

1. *AIS-Lib:* AIS-Lib, a library for performance evaluation of AIS based detection systems for ad hoc wireless networks, was introduced by Drozda et al. in [16]. This library is based on the Jist/SWANS network simulator [6]. It implements several misbehavior models such as data packet dropping, data packet delaying or wormholes [26]. The implementation of this library was an important step towards the feasibility of the simulation based performance evaluation presented in [17, 12].

2. *Applicability of negative selection for misbehavior detection:* We investigated the suitability of negative selection for misbehavior detection in ad hoc networks. We applied the computational approach to negative selection due to D'haeseleer et al. [10]. This approach considers bit-vector representation of features. It randomly produces a set of detectors that are used to detect anomalous behavior. Our results show that, under our experimental setup, less than 5% detectors were used in detecting misbehavior [13, 14].

   Additionally, we pointed out that simpler data traffic models such as the Constant bit rate (CBR) model may pose a challenge to misbehavior detection systems due to their synchronized nature, i.e. if data packets collide there is a high chance that data packet at a given node will again collide. This introduces a certain amount of noise into the system, and therefore constitutes a practical option in detection performance evaluation [42].

   The work presented in [13, 14, 42] was our starting point in the area of misbehavior detection research. Therein we formulated several key objectives related to the design of misbehavior detection systems, most notably, energy efficiency and ability to rely on a single instance of a learning algorithm. We also argued that it is necessary to expand our knowledge about features suitable for misbehavior detection. An evaluation of existing misbehavior detection approaches with respect to these objected led to the design of our co-stimulation and priming inspired misbehavior detection architectures.

   In [7] we compared negative selection with several standard machine learning algorithms such as decision tree classifier, naive Bayes classifier, support vector machines and neural networks. The performance evaluation was done using the same data sets as [13, 14]. Our results show that these machine learning algorithm offer a better misbehavior detection performance. With respect to this conclusion, we decided to focus on decision tree classifier when classifying the data obtained in [17, 12].

3. *Intrusion Detection in High Dimensional Space:* Our goal was to investigate the suitability of a locality sensitive hashing technique due to Datar et al. [9] for intrusion detection. This hashing technique allows for finding the approximate solution to the k-nearest neighbors problem, it can thus be used as a basis for a k-nearest neighbors classifier. In [36] we evaluate suitability of this technique to intrusion detection. We use several standard benchmark files, most notably the KDD'99 data files. These files were used in a competition that took place during the KDD'99 conference. Our results are comparable with those of the KDD'99 winner. The advantage of our approach is its applicability for classification when the number of features is high, i.e. when learning algorithms such as decision tree classifier, support vector machines or neural networks cannot cope with the feature vector dimensionality.

9 Conclusions

In this document we summarize our research efforts related to computational interpretation of two basic immune mechanisms: co-stimulation and priming. This interpretation was done in the scope

of misbehavior detection for ad hoc wireless networks. When formulating co-stimulation and priming as a computational approach we abstracted away from the molecular nature of these mechanisms. Instead we concentrated on their logic structure. Needless to say, despite applying such a simplified view, the path to a useful computational interpretation was not straightforward. This fact underscores the conceptual character of our research.

Our computational interpretation of co-stimulation allows any detection system to introduce two desirable properties: energy efficiency and false positives control. Additionally, it introduces the option to exchange energy efficiency for misbehavior detection rate, while keeping the false positives rate unchanged. Priming builds on the capability of co-stimulation to suppress false positives. Co-stimulation allows priming to learn several priming thresholds without paying attention to the efficiency of this leaning process. The increased false positives rate is decreased by a subsequent application of co-stimulation.

Even though, we concentrated on the applicability of co-stimulation and priming to misbehavior detection in ad hoc networks, we believe these methods are of general interest. An important characteristics of application scenarios, that could take advantage of these two methods, is the high cost connected with the relocation of information, services or resources, and at the same time, the feasibility to improve their control or allocation through one or several cost related parameters (such as time window size).

An example where co-stimulation and priming could be applicable is information broadcasting. When broadcasting information to users, the demand for any type of information can only be estimated [44]. Since the users cannot send feedback directly to the broadcast source (e.g. satellite), the feedback is sent intermittently using other paths for delivery (often a mix of wireless and wired delivery paths). The feedback frequency can increase the accuracy, it is however connected with a certain cost. It thus gives sense to track the demand with some prediction accuracy exchanged for cost efficiency.

A limiting factor in translating the functionality of the BIS to technical systems remains our narrow knowledge of the BIS. Immunology is a research area with a large number of experimental results published every year. These results are however often very specific to the life form in investigation. As an example, we cite a recent result on priming of plant immunity [29].

The success of the BIS in protecting its host often benefits from the efficiency of underlying chemical reactions at molecular level. Such an efficiency has often no obvious computational parallel. For example, the negative selection process required 15 years until an efficient computational counterpart could be presented [18].

10 List of Enclosed Documents

The results discussed in the previous sections were originally presented in the documents that we list below. We grouped these documents thematically into several groups. The first document is a review of immune inspired computing with focus on ad hoc wireless networks. The other documents present our results related to misbehavior detection in ad hoc wireless networks.

Introduction to Immune Inspired Computing with Focus on Ad Hoc Wireless Networks

Martin Drozda, Sven Schaust, Helena Szczerbicka. Immuno-Inspired Knowledge Management for Ad Hoc Wireless Networks. *In* Smart Information and Knowledge Management, Ngoc Thanh Nguyen and Edward Szczerbicki (Eds.), *Studies in Computational Intelligence*, vol. 260, pp. 1-26,

Springer, 2010. URL http://dx.doi.org/10.1007/978-3-642-04584-4_1

Co-stimulation and Priming

Martin Drozda, Sven Schaust, Sebastian Schildt, Helena Szczerbicka. Priming: Making the Reaction to Intrusion or Fault Predictable. *Natural Computing*, 32 pages, Springer, 2010. URL http://dx.doi.org/10.1007/s11047-010-9219-8. In press.

Martin Drozda, Sebastian Schildt, Sven Schaust, Helena Szczerbicka. An Immuno-Inspired Approach to Misbehavior Detection in Ad Hoc Wireless Networks. *Computing Research Repository (CoRR)*, 15 pages, 2010. URL http://arxiv.org/abs/1001.3113

Performance Evaluation of Immune Inspired Approaches

Martin Drozda, Sebastian Schildt, Sven Schaust, Sandra Einhellinger, Helena Szczerbicka. A Tool for Prototyping AIS Based Protection Systems for Ad hoc and Sensor Networks. *Cybernetics and Systems*, vol. 39, no. 7, pp. 719-742, Taylor&Francis, 2008. URL http://dx.doi.org/10.1080/01969720802257964

Comparing Negative Selection with Machine Learning Algorithms

Matthias Becker, Martin Drozda, Sven Schaust, Sebastian Bohlmann, Helena Szczerbicka. On Classification Approaches for Misbehavior Detection in Wireless Sensor Networks. *Journal of Computers*, vol. 4, no. 5, pp. 357-365, Academy Publisher, 2009. URL http://www.academypublisher.com/jcp/vol04/no05/jcp0405357365.html

Performance of Negative Selection in Misbehavior Detection Scenarios

Martin Drozda, Sven Schaust, Helena Szczerbicka. AIS for Misbehavior Detection in Wireless Sensor Networks: Performance and Design Principles. In *Proc. of IEEE Congress on Evolutionary Computation (CEC'07)*, pp. 3719-3726, Singapore, 2007. URL http://dx.doi.org/10.1109/CEC.2007.4424955

Martin Drozda, Sven Schaust, Helena Szczerbicka. Is AIS Based Misbehavior Detection Suitable for Wireless Sensor Networks? In *Proc. of IEEE Wireless Communications and Networking Conference (WCNC'07)*, pp. 3130-3135, Hong Kong, 2007. URL http://dx.doi.org/10.1109/WCNC.2007.578

Sven Schaust, Martin Drozda. Influence of Network Payload and Traffic Models on the Detection Performance of AIS. In *Proc. of 2008 International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'08)*, pp. 44-51, IEEE Press, Edinburgh, UK, 2008.

Suitability of Locality Preserving Hashing for Intrusion Detection

Stanislav Marcek, Martin Drozda, Gabriel Juhas, Fedor Lehocki. Intrusion Detection in High Dimensional Space. In *Proc. of Second International Symposium on Applied Sciences in Bio-Medical and Communication Technologies (ISABEL'09)*, 6 pages, IEEE Press, Bratislava, Slovakia, 2009. URL http://dx.doi.org/10.1109/ISABEL.2009.5373652

References

[1] Aickelin, U., Bentley, P., Cayzer, S., Kim, J., McLeod, J.: Danger theory: The link between ais and ids? In: Timmis, J., Bentley, P.J., Hart, E. (eds.) ICARIS '03: Proceedings of the International Conference on Artificial Immune Systems. Lecture Notes in Computer Science, vol. 2787, pp. 147–155. Springer Berlin / Heidelberg, Edinburgh, UK (2003), URL http://dx.doi.org/10.1007/978-3-540-45192-1_15

[2] Alpaydin, E.: Introduction To Machine Learning. MIT Press (2004)

[3] Asuncion, A., Newman, D.: UCI machine learning repository (2007), URL http://www.ics.uci.edu/~mlearn/MLRepository.html

[4] Banchereau, J., Briere, F., Caux, C., Davoust, J., Lebecque, S., Liu, Y., Pulendran, B., Palucka, K.: Immunobiology of dendritic cells. Annual review of immunology 18(1), 767–811 (2000), URL http://dx.doi.org/10.1146/annurev.immunol.18.1.767

[5] Barford, P., Kline, J., Plonka, D., Ron, A.: A signal analysis of network traffic anomalies. In: IMW '02: Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurement. pp. 71–82. ACM, Marseille, France (2002), URL http://doi.acm.org/10.1145/637201.637210

[6] Barr, R., Haas, Z., van Renesse, R.: JiST: an efficient approach to simulation using virtual machines. Software Practice and Experience 35(6), 539–576 (2005), URL http://dx.doi.org/10.1002/spe.647

[7] Becker, M., Drozda, M., Schaust, S., Bohlmann, S., Szczerbicka, H.: On Classification Approaches for Misbehavior Detection in Wireless Sensor Networks. Journal of Computers 4(5), 357 (2009), URL http://dx.doi.org/10.4304/jcp.4.5.357-365

[8] Bhuse, V., Gupta, A., Lilien, L.: DPDSN: Detection of packet-dropping attacks for wireless sensor networks. In: Proceedings of the Fourth International Trusted Internet Workshop. Goa, India (2005)

[9] Datar, M., Immorlica, N., Indyk, P., Mirrokni, V.: Locality-sensitive hashing scheme based on p-stable distributions. In: Proceedings of the twentieth annual symposium on Computational geometry. pp. 253–262. ACM New York, NY, USA (2004), URL http://dx.doi.org/10.1145/997817.997857

[10] D'haeseleer, P., Forrest, S., Helman, P.: An Immunological Approach to Change Detection: Algorithms, Analysis and Implications. In: Proc. of IEEE Symposium on Security and Privacy. pp. 110–119. Los Angeles: IEEE computer society (1996), URL http://dx.doi.org/10.1109/SECPRI.1996.502674

[11] Drozda, M., Schaust, S.: Costimulation and Priming: Can it Help Protect Ad Hoc Wireless Networks? In: Proceedings of the 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL'09) (2009), URL http://dx.doi.org/10.1109/ISABEL.2009.5373649

[12] Drozda, M., Schaust, S., Schildt, S., Szczerbicka, H.: Priming: Making the Reaction to Intrusion or Fault Predictable. Natural Computing (2010), URL http://dx.doi.org/10.1007/s11047-010-9219-8, to appear

[13] Drozda, M., Schaust, S., Szczerbicka, H.: AIS for Misbehavior Detection in Wireless Sensor Networks: Performance and Design Principles. In: Proc. IEEE Congress on Evolutionary Computation (CEC). pp. 3719–3726 (2007), URL http://dx.doi.org/10.1109/CEC.2007.4424955

[14] Drozda, M., Schaust, S., Szczerbicka, H.: Is AIS Based Misbehavior Detection Suitable for Wireless Sensor Networks? In: Proc. of IEEE Wireless Communications and Networking Conference (WCNC'07). pp. 3130–3135 (2007), URL http://dx.doi.org/10.1109/WCNC.2007.578

[15] Drozda, M., Schaust, S., Szczerbicka, H.: Immuno-Inspired Knowledge Management for Ad Hoc Wireless Networks. In: Smart Information and Knowledge Management, Ngoc Thanh Nguyen and Edward Szczerbicki (Eds.), Studies in Computational Intelligence, vol. 260, pp. 1–26. Springer (2010), URL http://dx.doi.org/10.1007/978-3-642-04584-4_1

[16] Drozda, M., Schildt, S., Schaust, S., Einhellinger, S., Szczerbicka, H.: A tool for prototyping AIS based protection systems for ad hoc and sensor networks. Cybernetics and Systems 39(7), 719–742 (2008), URL http://dx.doi.org/10.1080/01969720802257964

[17] Drozda, M., Schildt, S., Schaust, S., Szczerbicka, H.: An Immuno-Inspired Approach to Misbehavior Detection in Ad Hoc Wireless Networks. Computing Research Repository (CoRR) (2010), URL http://arXiv.org/abs/1001.3113

[18] Elberfeld, M., Textor, J.: Efficient Algorithms for String-Based Negative Selection. In: Proc. of International Conference on Artificial Immune Systems (ICARIS). pp. 109–121 (2009), URL http://dx.doi.org/10.1007/978-3-642-03246-2_14

[19] Feeney, L., Nilsson, M.: Investigating the energy consumption of a wireless network interface in an ad hoc networking environment. In: INFOCOM 2001: Proceedings of Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies. vol. 3, pp. 1548–1557. Anchorage, Alaska (2001), URL http://dx.doi.org/10.1109/INFCOM.2001.916651

[20] Forrest, S., Perelson, A., Allen, L., Cherukuri, R.: Self-nonself discrimination in a computer. In: Proceedings of IEEE Computer Society Symposium on Research in Security and Privacy. pp. 202–212. Oakland, CA, USA (1994), URL http //dx.doi.org/10.1109/RISP.1994.296580

[21] Frauwirth, K.A., Thompson, C.B.: Activation and inhibition of lymphocytes by costimulation. The Journal of Clinical Investigation 109(3), 295–299 (2 2002), URL http://www.jci.org/articles/view/14941

[22] Gama, J., Brazdil, P.: Cascade generalization. Machine Learning 41(3), 315–343 (2000), URL http://dx.doi.org/10.1023/A:1007652114878

[23] Gonzalez, O., Howarth, M., Pavlou, G.: Detection of packet forwarding misbehavior in mobile ad-hoc networks. In: Boavida, F., Monteiro, E., Mascolo, S., Koucheryavy, Y. (eds.) Proceedings of the International Conference on Wired/Wireless Internet Communications. Lecture Notes in Computer Science, vol. 4517, pp. 302–314. Springer Berlin / Heidelberg, Coimbra, Portugal (2007), URL http://dx.doi.org/10.1007/978-3-540-72697-5_26

[24] Hofmeyr, S., Forrest, S.: Immunity by design: An artificial immune system. In: GECCO '99: Proceedings of Genetic and Evolutionary Computation Conference. vol. 2, pp. 1289–1296. Morgan Kaufmann, Orlando, Florida, USA (1999)

[25] Hu, Y., Perrig, A., Johnson, D.: Packet leashes: a defense against wormhole attacks in wireless networks. In: INFOCOM 2003: Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies. vol. 3, pp. 1976–1986. IEEE, San Francisco, CA, USA (2003), URL http://dx.doi.org/10.1109/INFCOM.2003.1209219

[26] Hu, Y., Perrig, A., Johnson, D.: Wormhole attacks in wireless networks. IEEE Journal on Selected Areas in Communications 24(2), 370–380 (2006), URL http://dx.doi.org/10.1109/JSAC.2005.861394

[27] Huang, Y., Lee, W.: A cooperative intrusion detection system for ad hoc networks. In: SASN '03: Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks. pp. 135–147. ACM, Fairfax, Virginia (2003), URL http://doi.acm.org/10.1145/986858.986877

[28] Janeway Jr, C.: The immune system evolved to discriminate infectious nonself from noninfectious self. Immunology Today 13(1), 11–16 (1992), URL http://dx.doi.org/10.1016/0167-5699(92)90198-G

[29] Jung, H.W., Tschaplinski, T.J., Wang, L., Glazebrook, J., Greenberg, J.T.: Priming in Systemic Plant Immunity. Science 324(5923), 89–91 (2009), URL http://dx.doi.org/10.1126/science.1170025

[30] Kaynak, C., Alpaydin, E.: Multistage cascading of multiple classifiers: One man's noise is another man's data. In: Proc. 17th International Conf. on Machine Learning. pp. 455–462 (2000)

[31] Kohavi, R., John, G.: Wrappers for feature subset selection. Artificial Intelligence 97(1-2), 273–324 (1997), URL http://dx.doi.org/10.1016/S0004-3702(97)00043-X

[32] Krishnamurthy, S., Thamilarasu, G., Bauckhage, C.: Malady: A machine learning-based autonomous decision-making system for sensor networks. In: Proc. of IEEE International Conference on Computational Science and Engineering. vol. 2, pp. 93–100. IEEE Computer Society, Vancouver, Canada (2009), URL http://dx.doi.org/10.1109/CSE.2009.246

[33] Le Boudec, J., Sarafijanovic, S.: An Artificial Immune System Approach to Misbehavior Detection in Mobile Ad-Hoc Networks. In: Proc. of Bio-ADIT. pp. 96–111. Springer (2004), URL http //dx.doi.org/10.1007/978-3-540-27835-1_29

[34] Liu, Y., Janeway, C.: Cells that present both specific ligand and costimulatory activity are the most efficient inducers of clonal expansion of normal CD4 T cells. Proceedings of the National Academy of Sciences of the United States of America 89(9), 3845 (1992)

[35] Mahajan, V., Leskov, I., Chen, J.: Homeostasis of T cell diversity. Cell Mol Immunol 2(1), 1–10 (2005)

[36] Marcek, S., Drozda, M., Juhas, G., Lehocki, F.: Network Intrusion Detection in High Dimensional Space. In: 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL) (2009), URL http://dx.doi.org/10.1109/ISABEL.2009.5373652

[37] Marti, S., Giuli, T.J., Lai, K., Baker, M.: Mitigating routing misbehavior in mobile ad hoc networks. In: MobiCom '00: Proceedings of the 6th annual international conference on Mobile computing and networking. pp. 255–265. ACM, Boston, Massachusetts, United States (2000), URL http://doi.acm.org/10.1145/345910.345955

[38] Medzhitov, R., Janeway, C.: Innate immunity: impact on the adaptive immune response. Current opinion in immunology 9(1), 4–9 (1997), URL http://dx.doi.org/10.1016/S0952-7915(97)80152-5

[39] Mierswa, I., Wurst, M., Klinkenberg, R., Scholz, M., Euler, T.: Yale: Rapid prototyping for complex data mining tasks. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. pp. 935–940. ACM, Philadelphia, PA, USA (2006)

[40] Piccirillo, C., Thornton, A.: Cornerstone of peripheral tolerance: naturally occurring CD4+ CD25+ regulatory T cells. Trends in immunology 25(7), 374–380 (2004), URL http://dx.doi.org/10.1016/j.it.2004.04.009

[41] Sarafijanovic, S., Le Boudec, J.Y.: An artificial immune system for misbehavior detection in mobile ad-hoc networks with virtual thymus, clustering, danger signal, and memory detectors. In: Nicosia, G., Cutello, V., Bentley, P.J., Timmis, J. (eds.) ICARIS '04: Proceedings of the International Conference on Artificial Immune Systems. Lecture Notes in Computer Science, vol. 3239, pp. 342–356. Springer Berlin / Heidelberg, Catania, Sicily (2004), URL http://dx.doi.org/10.1007/978-3-540-30220-9_28

[42] Schaust, S., Drozda, M.: Influence of Network Payload and Traffic Models on the Detection Performance of AIS. In: Procedings of International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS). pp. 44–51. IEEE Press, Edinburgh, UK (2008)

[43] Szor, P.: The art of computer virus research and defense. Addison-Wesley Professional (2005)

[44] Vaidya, N., Hameed, S.: Scheduling data broadcast in asymmetric communication environments. Wireless Networks 5(3), 171–182 (1999), URL http://dx.doi.org/10.1023/A:1019142809816

[45] Yegneswaran, V., Barford, P., Ullrich, J.: Internet intrusions: global characteristics and prevalence. In: SIGMETRICS '03: Proceedings of the 2003 ACM SIGMETRICS international conference on Measurement and modeling of computer systems. pp. 138–147. ACM, San Diego, CA, USA (2003), URL http://doi.acm.org/10.1145/781027.781045