An Optimal Control Approach to Malware Filtering

Michael Bloem[1,2]
NASA Ames Research Center
Moffett Field, CA 94035-1000, USA
michael.j.bloem@nasa.gov

Tansu Alpcan[1]
Deutsche Telekom Laboratories
Technische Universität Berlin
Ernst-Reuter-Platz 7, 10587, Germany
tansu.alpcan@telekom.de

Tamer Başar[2]
Coordinated Science Lab
University of Illinois
1308 West Main Street
Urbana, IL 61801, USA
tbasar@control.csl.uiuc.edu

Abstract— **We study and develop an optimal control theoretic approach to malware filtering in the context of network security. We investigate the malware filtering problem by capturing the tradeoff between increased security on one hand and continued usability of the network on the other. We analyze the problem using a linear control system model with a quadratic cost structure and develop algorithms based on H^{∞}-optimal control theory. A dynamic feedback filter is derived and shown to be an improvement over various heuristic approaches to malware filtering via numerical analysis. The results obtained are verified and demonstrated with packet level simulations on the Ns-2 network simulator.**

I. Introduction

The cost of malicious software and attacks to computer networks is well documented, and these costs only grow as corporations and organizations become more dependent upon networked systems and attacks increase in sophistication. As networks become more complex, preventing and responding to malicious attacks also become more costly. Estimates put the cost of the Code Red virus attack of 2001 at $740 million in cleanup, monitoring, and system checking, and $450 million in lost productivity [1]. Aside from their daunting magnitude, these two types of costs are interesting in the sense that they can be traded off against each other.

Attacks on computer networks, such as worm or denial of services attacks, are expensive in part due to the challenge of preventing them while allowing legitimate network usage. The *base-rate fallacy* captures the essence of this problem. Even if we have low false-negative and false-positive rates in our detection of attacks, there is so much more legitimate network usage than illegitimate usage that we end up with many false alarms [2]. Intrusion detection systems must be constructed with this dilemma in mind, and thus need to be conservative in their operation.

In this paper, we use an optimal control approach to investigate how to dynamically choose an optimal security level high enough to adequately prevent costly attacks but not so high as to excessively prevent legitimate network usage. Specifically, we apply optimal control theory based methods to address the question of how to set up dynamic network traffic filters to prevent attacks or slow the spread of malicious software or *malware* within a single network by protecting sub-networks. Our aim is to develop algorithms and policies for configurable firewalls [3] in order to filter malware traffic such as worms, viruses, spam, and trojans.

We use H^{∞}-optimal control to determine how to dynamically change filtering rules in order to ensure a certain performance level. We note that in H^{∞}-optimal control, by viewing the disturbance as an intelligent maximizing opponent in a dynamic zero-sum game, who plays with knowledge of the minimizer's control action, one evaluates the system under the worst possible conditions. This approach applies naturally to the problem of malware response because the traffic deviation resulting from a malware attack is not merely random noise, but represents the efforts of an intelligent attacker. Therefore, we determine the control action that will minimize costs under these worst circumstances [4].

We study the algorithms developed via simulations in Matlab and Ns-2 network simulator and verify the optimality of our solution in various scenarios. To the best of the knowledge of the authors, this work represents the first application of H^{∞}-optimal control theory to the problem of malware filtering.

A. Related Work

There are several methods of dynamic packet filtering [5]. Perhaps the most common one is to dynamically change which ports are open or closed. Stateful inspection of deeper layers of packets allows for even more detailed filtering by creating and maintaining information about the state of a current connection [3]. Another possibility is to dynamically alter the set of IP addresses from which traffic will be accepted [6].

Implicit to the network traffic filtering problem considered in this paper is the partitioning of a computer network into various sub-networks for administrative and security purposes. This approach is common, and a separate firewall is often assigned to each sub-network. Zou et al. have proposed a "Firewall Network System" based on this very concept in [7]. Cisco recommends their IOS firewalls for defending particular sub-networks or LANs in a corporate network [3]. In [8], quarantining these sub-networks is considered as a strategy to slow the spread of worm epidemics. We note that although the algorithms developed in this paper can be helpful for configuring dynamic firewalls such as the

[1] Research supported in part by Deutsche Telekom AG.

[2] Research supported in part by a grant from the Boeing Company, through the Information Trust Institute at the University of Illinois at Urbana-Champaign.

ones described above, our main objective is to develop general and broadly applicable mathematical foundations and algorithms for future security systems which will be even more configurable and flexible.

Optimization as well as decision and control theory have been applied to network security in several contexts. In [9] and [10], game theory was used for quantitative modeling and to develop decision-making strategies for network intrusion detection and response. Control theory has been applied to the problem of worm propagation by controlling the number of connections made by an infected host [11]. Optimization of system administrator time and efforts has been studied in [12].

The remainder of this paper is structured as follows: Section II discusses the problem of filtering network traffic with dynamic firewalls separating sub-networks. Section III describes Matlab and Ns-2 simulations of the H$^\infty$-optimal controller and demonstrates its performance in comparison with other controllers. Concluding remarks and directions for future research are presented in Section IV.

II. Network Traffic Filtering

In this section we present a linear system model for malware traffic and study the problem of filtering network traffic to prevent malware propagation. Consider a computer network that is divided into sub-networks for administrative and security purposes [3]. The corresponding control framework can be applied to other contexts by re-defining the entities in question.

Let $x(t)$ represent the number of malware packets that traverse a link on their way to the destination sub-network at time t originating from infected sources outside the sub-network. We model this malware flow using a linear differential equation with control and disturbance terms:

$$\dot{x}(t) = a\,x(t) + b\,u(t) + w_a(t), \tag{1}$$

where $u(t)$ represents the number of packets that are filtered at a particular time (t). Usually, only some proportion of the packets filtered are actually malware related. Thus, the parameter b corresponds to that proportion multiplied by -1. On the other hand, $w_a(t)$ represents the number of malware packets added to the link at time t intentionally by malicious sources or unintentionally by hidden software running on hosts, both located outside the sub-network considered. Thus $u(t)$ and $w_a(t)$ represent, for this specific sub-network, the packet filtering rate and malware infiltration rate, respectively. The a value represents the instantaneous proportion of malware packets on the link that are actually delivered to the sub-network and is thus a negative number.

Expanding the dimensions of the model in (1) leads to a set of linear differential equations:

$$\dot{\mathbf{x}}(t) = A\mathbf{x}(t) + B\mathbf{u}(t) + D\mathbf{w}_a(t). \tag{2}$$

In this case both A and B are obtained simply by multiplying the identity matrix by a and b, respectively. The D matrix imposes a propagation model on the attack and quantifies how malware is routed and distributed on this network. For the purposes of this paper, it has zeros for its diagonal terms (intra-sub-network malware traffic does not leave the sub-network), and each column must sum to 1 to ensure conservation of packets. In this version of the problem, the malware being sent to sub-network i is a function of w_j for $j \neq i$, the malicious traffic generated by other sub-networks. This assumption on the propagation of malware inherent to the form given to D allows for a centralized filtering solution that considers network-wide conditions. A decentralized version to this problem is also possible.

Let $\mathbf{y}(t)$ be the number of inbound malicious packets prior to filtering. Inaccuracies in $\mathbf{y}(t)$ are inevitable due to the challenging problem of distinguishing malicious packets from legitimate ones [2]. To capture this uncertainty formally, we define $\mathbf{y}(t)$ as

$$\mathbf{y}(t) := C\mathbf{x}(t) + E\mathbf{w}_n(t), \tag{3}$$

where $\mathbf{w}_n(t)$ is measurement noise of any form. Later, we derive and apply the worst case measurement noise $\mathbf{w}_n(t)$. Additionally, we define $N := EE^T$ and assume that it is positive definite, meaning that the measurement noise impacts each dimension of the measured output. The C matrix models the assumption that $\mathbf{y}(t)$ has a larger magnitude than and is proportional to $\mathbf{x}(t)$. When implemented, entries of this constant matrix could be measured from an analysis of packet filtering and the calculations required for determining the optimal controller could be re-run periodically. Note that we do not make any assumption on how $\mathbf{y}(t)$ is obtained.

Similarly, $\mathbf{w}_a(t)$ represents a worm attack, expressed in terms of the number of malware packets sent from a sub-network to other sub-networks at each time instant. However, we do not assume any form on the attack. To simplify notation, let $\mathbf{w} := \left[\mathbf{w}_a^T \; \mathbf{w}_n^T\right]^T$.

A. H^∞-Optimal Controller Design

Our objective now is to design an algorithm or controller for traffic filtering given this imperfect measure of inbound malicious packets. As part of the H$^\infty$-optimal control analysis and design we introduce first the *controlled output*

$$\mathbf{z}(t) := H\mathbf{x}(t) + G\mathbf{u}(t), \tag{4}$$

where we assume that G^TG and H^TH are positive definite, and that no cost is placed on the product of control actions and states: $H^TG = 0$. Here, H represents a cost on malicious packets arriving at a sub-network. For this H$^\infty$ optimal control theory to apply, $(A,\, B)$ and $(A,\, D)$ must be stabilizable, and $(A,\, H)$ and $(A,\, C)$ be detectable – conditions which readily hold in our case.

Recall that b specifies the proportion of filtered traffic that is malware-related. Thus, $(1-b)$ is the proportion of filtered traffic that is legitimate. If f_l is the cost of filtering legitimate packets when malware packets are on the link and f_a is the cost of the filtering action itself, the components g of G can be specified as $g = f_l(1-b) + f_a$.

The cost of this system for the purpose of H$^\infty$ analysis is defined by

$$L(\mathbf{x}, \mathbf{u}, \mathbf{w}) = \frac{\|\mathbf{z}\|}{\|\mathbf{w}\|}, \tag{5}$$

where $\|\mathbf{z}\|^2 := \int_{-\infty}^{\infty} |\mathbf{z}(t)|^2 dt$ and a similar definition applies to $\|\mathbf{w}\|^2$. This is a cost ratio rather than an actual cost, but we will refer to it as the cost for simplicity. It captures the proportional changes in $\mathbf{z}$ due to changes in $\mathbf{w}$. More intuitively, it is the ratio of the cost incurred by the system to the corresponding attacker and measurement noise "effort".

H$^\infty$-optimal control theory not only applies very directly and appropriately to the problem of worm response, but also guarantees that a performance factor (the H$^\infty$ norm) will be met. This norm can be thought of as the worst possible value for the cost L and is bounded above by

$$\gamma^* := \inf_{\mathbf{u}} \sup_{\mathbf{w}} L(\mathbf{u}, \mathbf{w}), \qquad (6)$$

which can also be viewed as the optimal performance level in this H$^\infty$ context.

In order to actually solve for the optimal controller $\mu(\mathbf{y})$, the number of packets to filter as a function of the inaccurately measured number of inbound malicious packets, a corresponding differential game is defined between the attackers and the malware filtering system, which is parameterized by γ, where $\gamma > \gamma^*$:

$$J_\gamma(\mathbf{u}, \mathbf{w}) = \|\mathbf{z}\|^2 - \gamma^2 \|\mathbf{w}\|^2. \qquad (7)$$

The malicious attackers try to maximize this cost function by varying $\mathbf{w}$ while the malware filtering algorithm minimizes it via the controller $\mathbf{u}$. A similar application of game theory, where attackers and intrusion detection/prevention system are modeled as players in a security game, has been investigated in [9].

The optimal filtering strategy $\mathbf{u} = \mu_\gamma(\mathbf{y})$ can be determined from this differential game formulation for any $\gamma > \gamma^*$. It is given by [4]

$$\mu_\gamma(\mathbf{y}) = -(G^T G)^{-1} B^T \bar{Z}_\gamma \hat{\mathbf{x}}, \qquad (8)$$

where $\bar{Z}_\gamma$ is solved from

$$A^T Z + Z A - Z(B(G^T G)^{-1} B^T - \gamma^{-2} D D^T) Z + H^T H = 0, \qquad (9)$$

as its unique minimal positive definite solution, and $\hat{\mathbf{x}}$ is given by

$$\begin{aligned} \dot{\hat{\mathbf{x}}} &= \left[A - (B(G^T G)^{-1} B^T - \gamma^{-2} D D^T) \bar{Z}_\gamma\right] \hat{\mathbf{x}} \\ &\quad + \left[I - \gamma^{-2} \bar{\Sigma}_\gamma \bar{Z}_\gamma\right]^{-1} \bar{\Sigma}_\gamma C^T N^{-1} (\mathbf{y} - C\hat{\mathbf{x}}), \qquad (10) \end{aligned}$$

where $\bar{\Sigma}_\gamma$ is the unique minimal positive definite solution of

$$A\Sigma + \Sigma A^T - \Sigma(C^T N^{-1} C - \gamma^{-2} H^T H)\Sigma + D D^T = 0. \qquad (11)$$

Here $\hat{\mathbf{x}}$ is an estimate for $\mathbf{x}$. This is a linear feedback controller operating on a state estimate. Note that γ^* is also the smallest γ such that the spectral radius $\rho(\bar{\Sigma}_\gamma \bar{Z}_\gamma) < \gamma^2$.

Remark 1. *The H$^\infty$-optimal controller derived here in (8) is a centralized control solution due to the D matrix, which imposes a specific malware propagation model. However, we can apply the same framework to each sub-network separately by using (1) for each. This leads to a decentralized solution consisting of independent scalar H$^\infty$-optimal controllers.*

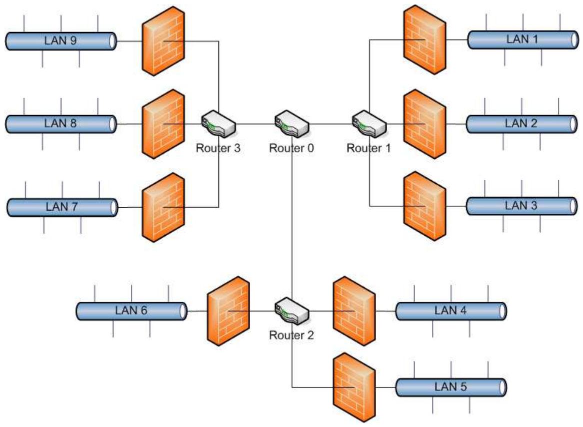

Fig. 1. Sample computer network to be analyzed.

III. Numerical Analysis of Network Traffic Filtering

Consider the representative computer network shown in Fig. 1. In this simple network configuration, each sub-network or LAN has a dynamic firewall that filters incoming network traffic. Each firewall communicates its malicious packet measure $\mathbf{y}$ to all other firewalls, where filtering decisions are made. No centralized server is overseeing the filtering activity.

A. Matlab Simulations

We first conduct a numerical analysis in Matlab. We consider a simplified model of malware that is spreading between sub-networks within this network. Several variations of this modeled malware are simulated on this network topology in order to compare the H$^\infty$-optimal controller with other controllers. As a simplification, a sub-network is assumed to be either infected or not infected. An infected sub-network sends malware to other sub-networks. Sub-networks become infected with some probability once they have received a certain threshold number of malware packets. This probability increases when higher thresholds are met.

Four types of malware attacks are considered: no attack (A1), a high-traffic, slow spreading attack (A2), a low-traffic, slow-spreading attack (A3), and a low-traffic, fast-spreading attack (A4). In each attack, one sub-network is initially infected and sends malware to all other sub-networks.

Five response types are applied to each of these attack types: no response (R1), the H$^\infty$-optimal controller response (R2), a threshold-based controller that implements a filter of some fixed magnitude when a certain amount of malicious packets are detected (R3), a controller that removes all suspicious packets ($\mathbf{y}(t)$) from each link (R4), and an optimal controller that minimizes the cost $\|\mathbf{z}\|^2$ (R5). For the linear quadratic Gaussian (LQG) optimization problem in (R5), which is obtained as the limit of the H$^\infty$ problem as $\gamma \to \infty$, we use the expected value of $\int_{-\infty}^{\infty} \|\mathbf{z}\|^2 dt$ as the quadratic cost, which we denote by $\|\mathbf{z}\|^2$ by a slight abuse of notation.

These responses can conveniently be evaluated and compared based on the cost ($\|\mathbf{z}\|^2$) and cost ratio (L) defined

TABLE I

Cost Ratios (L) of Controllers Under Various Attacks ($b = 0.5$)

Attack	R1	R2	R3	R4	R5
A1	0.00	3.48	0.00	2.35	2.04
A2	8.36	3.00	8.02	4.45	5.42
A3	9.07	2.88	5.76	4.42	4.71
A4	9.42	2.90	5.31	4.49	5.15

TABLE II

Costs ($\|z\|^2$) of Controllers Under Various Attacks ($b = 0.5$) ($\times 10^3$)

Attack	R1	R2	R3	R4	R5
A1	0	1.172	0	0.788	0.682
A2	105.4	18.24	94.08	46.85	88.24
A3	22.68	5.579	16.77	12.50	10.34
A4	27.97	4.979	13.51	12.63	14.24

above. Simulations are run with three sets of costs ($\|\mathbf{z}\|^2$ and L) that differ in their coefficients. The ratio between the cost $H^T H$ on inbound malware packets $\mathbf{x}$ and the cost $G^T G$ on filtering packets $\mathbf{u}$ is set at 10:1, 100:1, and 1000:1.

The A matrix is set to be the identity matrix multiplied by -1. The b quantity is set to 0.5. This value is consistent with a detection rate (true-positive rate) of 0.7 and a very low (10^{-5}) false-positive rate [2]. The D matrix is set up such that sub-networks are more likely to transfer the worm within their group of three sub-networks. The C matrix is set to be 2 multiplied by the identity matrix, which is derived from values observed in the Ns-2 simulations (Sub-section III-B).

Essentially, we allow the system model to evolve according to (2), updating the $\mathbf{w}_a(t)$ term according to the attacks A1-A4 described above and updating the $\mathbf{u}(t)$ term according to the responses R1-R5 described above.

The H$^\infty$-optimal controller (R2) performs better than every other controller whenever malware is present, as seen in Table I. In this case, we choose a 100:1 malware packet to filtering action cost ratio. The resulting γ^* is 4.52.

Table II shows the actual costs incurred by the system in each scenario with the same cost structure. The significantly lower cost values for the H$^\infty$-optimal controller in the face of attacks highlight its ability to filter enough to prevent sub-networks from becoming infected.

The preventative ability of the H$^\infty$-optimal controller can also be observed in Fig. 2. As soon as the first network detects an increase in inbound malware packets shortly after 10 time units, the controller begins filtering significantly (see Fig. 2 “Filtering Rate”) all across the network. This prevents the second sub-network from becoming infected (note that it never sends malware in Fig. 2 “Malware Sending Rate”).

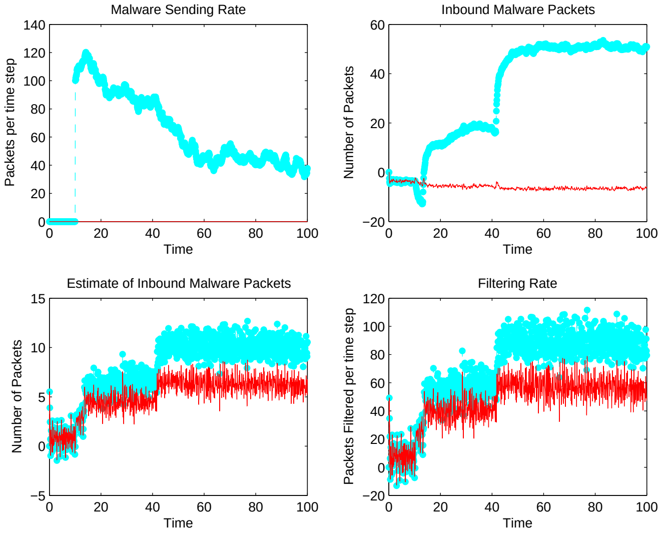

Fig. 2. Numerical analysis of slow worm attack with H$^\infty$ response applied on two sub-networks.

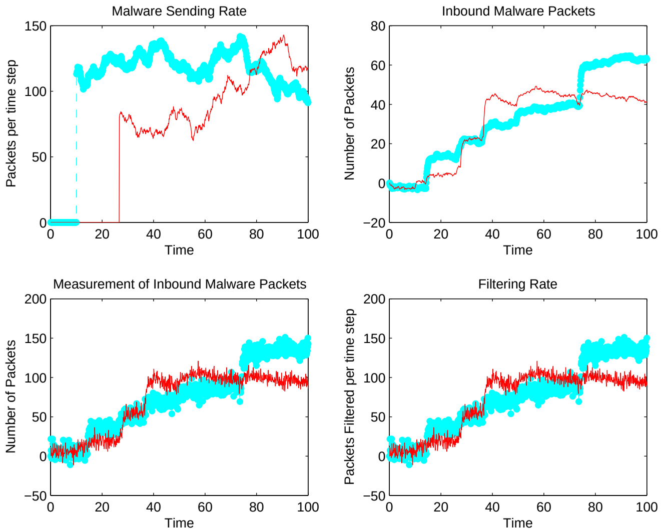

Fig. 3. Numerical analysis of slow worm attack with the heuristic controller that removes as many malware packets as it measures on two sub-networks.

For comparison, Fig. 3 shows the performance of the controller that removes all the estimated malware packets, thereby disregarding measurement errors and network-wide conditions. While it does over-filter, it does *not* filter network-wide when a single sub-network detects significant numbers of malware packets. Thus the uninfected sub-network eventually becomes infected at around time step 25, which causes it to send malware (Fig. 3). The LQR optimal controller (R5), on the other hand, does filter network-wide upon detection of inbound malware packets anywhere in the network. It does not, however, filter as much as the H$^\infty$-optimal controller. Moreover, it is hindered in that it assumes a zero-mean disturbance, an assumption that becomes more inaccurate as more sub-networks become infected. The H$^\infty$-optimal controller, on the other hand, incurs relatively high costs and cost ratios when there are no infected sub-networks due to its network-wide over-response (Tables I and II).

Simulations were also run for other cost functions. The H$^\infty$-optimal controller performed relatively better when there

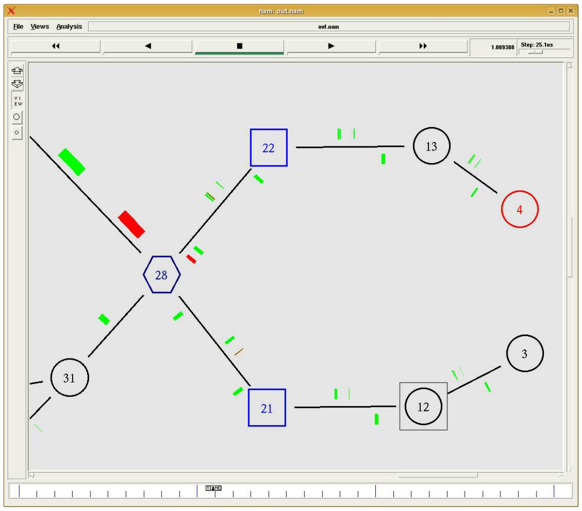

Fig. 4. Screenshot of the Ns-2 simulator output. Green packets are legitimate and red packets are malware.

was a greater cost put on the inbound malware packets and vice versa. This is to be expected, as this controller is rewarded more for being cautious when the inbound malware packets increase in cost. When the b value was decreased from 0.5 to 0.3, the H^∞-optimal controller also performs relatively better. This decrease in b means that when filtering does occur, we are less likely to actually filter a malicious packet, and thus controllers that filter more are rewarded.

B. Ns-2 Implementation

We simulate the traffic control algorithm developed at the packet level using the *Ns-2* network simulator. Our goal is to further investigate the characteristics of the designed H^∞-optimal controller and demonstrate its capabilities in a realistic setting. We define in Ns-2 the same network topology as in Sub-section III-A, which is depicted in Fig. 1.

In order to simulate the filtering algorithm, we consider here a specific implementation consisting of *monitoring* and *filtering* elements. The monitoring nodes, depicted as hexagons in Fig. 4, associate a malware score $s \in [0, 99]$ to each individual packet passing through the link from the outside. We generate the scores randomly according to different probability distributions for legitimate and malicious packets. The monitoring elements use this score s and a specific constant threshold to make an initial estimate on the nature of the packet and label it as malware or not. A count of these observed malware packets gives $\mathbf{y}(t)$. The monitoring node may utilize any algorithms or approaches to determine this quantity.

The filtering elements depicted as boxes in Fig. 4 fetch the malware score s and the flag from the headers of inbound packets and use either a heuristic or a H^∞ controller-based algorithm to make filtering decisions. The algorithms decide on a time-varying threshold value (different than the previous constant measurement threshold), resulting in a dynamic filtering scheme. Packets with a score higher than the threshold are filtered. For comparison purposes, we simulate the R4 algorithm in Section III-A, which we denote as *heuristic*, in addition to the H^∞ algorithm.

We calculate the H^∞-optimal controller offline in Matlab and transfer the results to the Ns-2 simulator. In accordance with the model in Section II, the resulting controller decides on the number of malware packets to be filtered at a given time interval (i.e. packet filtering). We translate this number into a threshold value by periodically observing the distribution of scores generated by the monitoring element. Hence, the threshold is chosen such that the number of packets with a score higher than the threshold (i.e., to be filtered) matches the number dictated by the H^∞-optimal controller.

Remark 2. *The example Ns-2 implementation we choose here does not play a significant role for the analysis and demonstration of our algorithm. In fact, depending on the specific application at hand, one can choose a variety of equivalent implementations without loss of any generality. For example, the monitoring and filtering elements can be parts of larger units each or combined within a dedicated physical device. The possible combinations are numerous.*

We simulate, compare, and contrast the H^∞ and detection-based heuristic filtering schemes in a variety of scenarios under different cost structures, detection capabilities, and traffic levels. The hypothetical scenarios we consider are summarized as follows:

1) A cost on malware packets ($\mathbf{x}$) to cost on filtering ($\mathbf{u}$) ratio of 100:1 in $\|\mathbf{z}\|^2$ and L. We assume that the monitoring devices are capable of scoring and labeling only half of the malware packets correctly (S1).
2) The cost is the same as in Scenario 1, but we consider a more pessimistic case where the monitoring device only detects a quarter of the total malware packets (S2).
3) This scenario is the same as Scenario 1 except for an increase in the cost coefficient ratio to 200:1 (S3).
4) Likewise this scenario is the same as Scenario 2 with a cost coefficient ratio of 200:1 (S4).
5) The final scenario matches Scenario 1 but has a cost coefficient ratio of 0.1:1 (S5).

In all of the above cases, each end-node (sub-network) sends randomly fluctuating 1000KB legitimate traffic to all sub-networks. In addition we consider an "infection" or worm-like malware propagation scheme, where each sub-network becomes "infected" with some probability if it receives sufficiently many malware packets and afterwards generates malware traffic of 200KB to other sub-networks.

The numerical results for both of the algorithms under each scenario described above are summarized in Table III. We observe here several expected characteristics of the H^∞ controller such as optimality with respect to the cost functions and robustness. In almost all of the cases and over a wide range of cost coefficient ratios it outperforms the detection-based heuristic scheme. More importantly, it exhibits robustness with respect to variations in detection quality (see case 1 versus 2) and guarantees an upper bound on the cost L. It is observed that the L value is always near

TABLE III
COST RESULTS OF NS-2 SIMULATIONS

	H^∞-Optimal			Detection-based	
Scen.	L	$\|\mathbf{z}\|^2 \quad (\times 10^6)$	γ^*	L	$\|\mathbf{z}\|^2 \quad (\times 10^6)$
S1	3.9	77	3.2	4.9	147
S2	3.7	89	3.2	6.6	369
S3	4.2	87	4.2	6.9	287
S4	4.9	155	4.2	9.3	736
S5	0.31	0.68	0.3	1.05	6.78

the theoretically calculated bound γ^*. Another indication of the H$^\infty$-optimal controller's robustness is the satisfactory performance of the controller even though it is calculated offline with estimated system characteristics. This, along with the assumptions inherent in the model, explains the occasional discrepancies observed between L values and the theoretical upper-bounds γ^*.

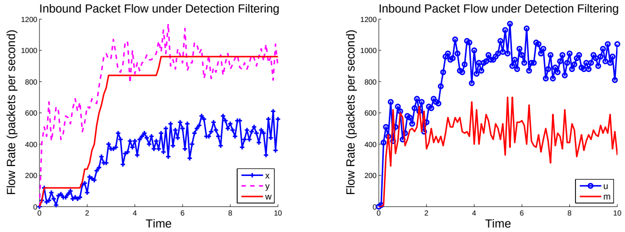

Fig. 5. Various inbound packet flow rates to sub-network 1 under the detection-based filtering.

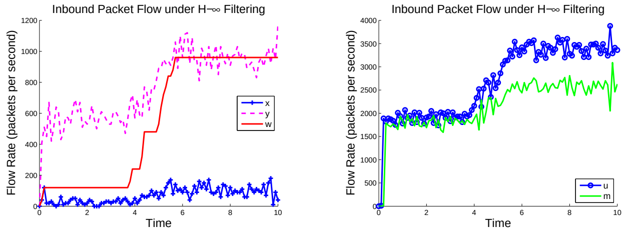

Fig. 6. Various inbound packet flow rates to sub-network 1 under H$^\infty$ controller.

We next analyze the time-series data collected for a representative sub-network. We depict quantities of interest x (malware packets that pass through the filter), y (packets labeled as malware by monitor), and u (filtering rate) as in Sub-section III-A. In addition, we plot the the rate of falsely positive labeled packets m and the rate of real malware flow, w. Fig. 5 shows the evolution of these quantities over time in Scenario 1 under the detection-based scheme, whereas Fig. 6 depicts the counterpart for the H$^\infty$ controller. We observe that the H$^\infty$ controller performs better than the detection-based scheme in terms of removing the malware packets through aggressive filtering in line with the preferences expressed in the cost function. Concurrently, this leads to a slower infection rate as can be inferred from the evolution of real malware flow rate (w) in Fig. 6.

IV. CONCLUSION

We have studied an application of optimal control theory to network security by investigating an optimal control formulation of the network filtering problem that captures its inherent challenges such as the base-rate fallacy and takes into account relevant costs. The corresponding H$^\infty$-optimal controller has been derived and analyzed numerically in Matlab as well as simulated in Ns-2. It has been observed to perform better than alternative controllers when there is a significant amount of malware traffic present. In addition, it provides a certain performance guarantee for a wide range of conditions.

There exist several possible extensions to this work. Obtaining a distributed version of this controller for a larger system could be one future direction. Another research direction is the application of similar H$^\infty$-optimal controllers to other network security problems, such as spam filtering and DDoS attacks. Finally, this theoretical work should be applied and tested in actual networks.

Overall, the promising results in this paper demonstrate that optimal control theory is applicable and provides a powerful approach to network security problems.

REFERENCES

[1] Reuters, "The cost of 'code red': $1.2 billion," USA Today, Aug. 1, 2001, http://usatoday.com/tech/news/2001-08-01-code-red-costs.htm.

[2] S. Axelsson, "The base-rate fallacy and its implications for the difficulty of intrusion detection," in *Proc. of 6th ACM Conference on Computer and Communications Security*, Kent Ridge Digital Labs, Singapore, 1999, pp. 1–7.

[3] Cisco, "NAT and stateful inspection in Cisco IOS firewall," white paper, 2006. [Online]. Available: http://www.cisco.com/en/US/tech/tk648/tk361/technologies_white_paper09186a0080194af8.shtml

[4] T. Başar and P. Bernhard, *H^∞-Optimal Control and Related Minimax Design Problems: A Dynamic Game Approach*, 2nd ed. Boston, MA: Birkhäuser, 1995.

[5] M. Tulloch, *Microsoft Encyclopedia of Security*. Redmond, WA: Microsoft Press, 2003.

[6] S. Hazelhurst, "A proposal for dynamic access lists for TCP/IP packet filtering," in *Proc. of South African Instutute of Computer Scientists and Information Technologists Annual Conference*, Pretoria, South Africa, September 2001, http://arxiv.org/abs/cs/0110013.

[7] C. Zou, D. Towsley, and W. Gong, "A firewall network system for worm defense in enterprise networks," University of Massachusetts, Amherst, MA, Technical Report TR-04-CSE-01, Feb. 2004.

[8] T. M. Chen and N. Jamil, "Effectiveness of quarantine in worm epidemics," in *Proc. of IEEE ICC 2006*, Istanbul, Turkey, June 2006, pp. 2142–2147.

[9] T. Alpcan and T. Başar, "A game theoretic analysis of intrusion detection in access control systems," in *Proc. 43rd IEEE Conf. Decision and Control*, Paradise Island, Bahamas, December 2004, pp. 1568–1573.

[10] ——, "An intrusion detection game with limited observations," in *Proc. of 12th International Symposium on Dynamic Games and Applications*, Sophia Antipolis, France, July 2006.

[11] R. Dantu, J. Cangussu, and A. Yelimeli, "Dynamic control of worm propagation," in *Proc. of IEEE Conference on Information Technology: Coding and Computing*, vol. 1, April 2004, pp. 419–423.

[12] M. Bloem, T. Alpcan, and T. Başar, "Intrusion response as a resource allocation problem," in *Proc. 45th IEEE Conf. Decision and Control*, San Diego, CA, USA, December 2006.