Piecewise Linear and Stochastic Models for the Analysis of Cyber Resilience

Michael J. Weisman, Alexander Kott
U.S. Army Combat Capabilities Development Command
Army Research Laboratory
Adelphi, MD
{michael.j.weisman2, alexander.kott1}.civ@army.mil

Joachim Vandekerckhove
University of California, Irvine
Department of Cognitive Sciences
Irvine, CA
joachim@uci.edu

***Abstract*— We model a vehicle equipped with an autonomous cyber-defense system in addition to its inherent physical resilience features. When attacked, this ensemble of cyber-physical features (i.e., "bonware") strives to resist and recover from the performance degradation caused by the malware's attack. We model the underlying differential equations governing such attacks for piecewise linear characterizations of malware and bonware, develop a discrete time stochastic model, and show that averages of instantiations of the stochastic model approximate solutions to the continuous differential equation. We develop a theory and methodology for approximating the parameters associated with these equations.**

I. Introduction

In this paper, we report progress we have made on a project called *Quantitative Measurement of Cyber Resilience* (QMoCR) whose goals include building a mathematical model to characterize cyber resilence, and finding objective quantitative measures of cyber resilience. In [1], we began to develop the tools to model cyber resilience mathematically. In a companion paper [2], we described an experimental testbed we have developed, and gave examples of the data we can produce. In the current paper, we continue to develop the mathematical model. In particular, we expand on the piecewise linear model we described in [1] and exhibit its stochastic counterpart.

The QMoCR program has focused on two research areas: (1) mathematical modeling of cyber resilience and (2) framing an infrastructure for experimentation and measurements.

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein. This work was partially funded by Cyber Technologies, Deputy CTO for Critical Technologies/Applied Technology, Office of the Under Secretary of Defense Research and Engineering. Dr. Vandekerckhove's research was sponsored by the Army Research Laboratory Cooperative Agreement Number W911NF-21-2-0284 and National Science Foundation grants #1850849 and #2051186. The authors would like to thank the anonymous reviewers for carefully reading the paper and for providing helpful suggestions.

In the mathematical modeling work, we model the impact on a surrogate vehicle of actions by malware as well as "bonware" – the ensemble of cyber-physical features that defend the vehicle's computer system and allow it to recover from attack. We develop a differential equation that models the effects of these competing forces on the system. We extend this to a stochastic differential equation model that captures the effects of uncertainty and randomness in the activity times of malware and bonware.

In parallel with the mathematical modeling, we have developed an inexpensive experimental environment to test the effects of malware and bonware on the Controller Area Network (CAN) bus of a generic military vehicle. Our testbed includes (a) a PASTA platform: a vehicle security testbed developed by Toyota which features a set of connected electrical control units or ECUs), (b) Unity: a popular game development platform, and (c) the Active Defense Framework (ADF): a government-developed framework used to quickly produce and test network-based cyber-defense techniques. When running this environment, we capture data that can be used to characterize cyber resilience metrics of the modeled vehicles.

Although we have made significant progress in both of these areas, we have also identified some areas that may benefit from collaboration with research partners. Thus far, we have modeled the impact on a single vehicle. By considering a network of vehicles, many interesting problems can be formulated. If one vehicle is under attack, what are the impacts on other vehicles that exchange information with this vehicle or that are in physical proximity? What are the probabilities that neighboring vehicles in the network are also under attack? How can our understanding of vehicle networks and robotics inform each other? What other techniques can we draw from to formulate models and compute metrics (for example game theory, neural networks, dueling network architectures, etc.)?

II. PRIOR WORK

In [1], we reviewed literature related to qualitative and quantitative assessments of a cyber system. We refer the reader to that paper for a more detailed summary. Here, we briefly review the quantitative work to date, and motivate the need for stochastic modeling of cyber resilience.

Most approaches to quantitative measurements of cyber resilience tend to involve the area under the curve (AUC) method [3, 4]. An experimental system engages in a collection of missions where it subject to cyber attack. Data containing the functionality of the system is collected, and a metric based on the area under the functionality curve is computed. Figure 1 illustrates the concept. One or more attacks compromise a system, causing the functionality to diminish. The ratio of the area under the curve and the area under the baseline curve (recorded during a mission where there is no attack) is computed.

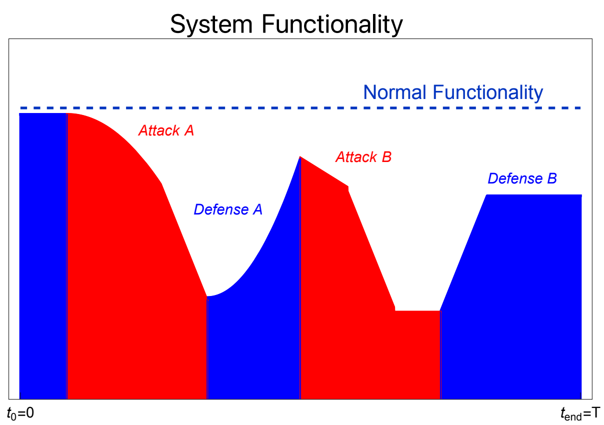

Figure 1. Resilience can be measured by subjecting a system to cyber attacks computing the ratio of the area under the compromised system functionality curve to that of the normal functionality curve.

This class of measures seems reasonable, however, AUC-based resilience measures are rather simple, and reveal little about the underlying processes. In [1] we began to explore ways to quantify the resilience impact of the bonware and quantify the impact of malware on a system. We also looked to understand how these values of impactfulness vary over time during an incident.

III. CONTINUOUS MODEL

We are interested in the functionality of our system, $F(t)$, which we defined in [1] to be the time derivative of mission accomplishment. We also proposed a baseline functionality (normal functionality), which in general could be time varying, but in our analysis, we take normal functionality, $F_{\mathrm{N}}(t) = F_{\mathrm{N}}$, to be a constant. We also assume that the system, prior to any attack or other deviation from normal operations, is operating normally: $F(t_0) = F_{\mathrm{N}}$. We assume that functionality is differentiable at least once and both malware impact and bonware impact are continuous functions of time: $F \in C^1$ and $\mathcal{M}, \mathcal{B} \in C^0$. The impact on functionality is the sum of the impacts of malware and bonware, and

$$\frac{dF}{dt} + \mathcal{Q}(t)F(t) = F_{\mathrm{N}}\mathcal{B}(t), \tag{1}$$

where $\mathcal{Q}(t) = \mathcal{M}(t) + \mathcal{B}(t)$. In [1], we reasoned that $\mathcal{B}(t), \mathcal{M}(t) \geq 0$, $F_{\mathrm{N}} > 0$, and $F_{\mathrm{N}} \geq F(t) \geq 0$ and found the general solution to this first-order linear differential equation with initial condition:

$$F(t) = e^{-\int_0^t \mathcal{Q}(p)\,dp}\left(F(0) + F_{\mathrm{N}}\int_0^t e^{\int_0^\tau \mathcal{Q}(p)\,dp}\mathcal{B}(\tau)\,d\tau\right).$$

In [1] we also exhibited solutions for a number of elementary examples. Figure 2 contains plots of piecewise constant and piecewise linear models. The models' differential equations and their solutions are summarized in Table I.

IV. STOCHASTIC DIFFERENTIAL EQUATION MODEL

In this section, we develop the *stochastic* differential equation (SDE) model associated with the piecewise linear model that we introduced in [1] (see Table I). The extension is motivated by the discontinuous nature of the notional data in Figure 1. Whereas the differential equation model assumed a smooth functionality curve, our stochastic version allows for a more punctuated attack-and-restoration pattern. In [1] we obtained

$$\frac{dF}{dt} = \left(F_{\mathrm{N}} - F(t)\right) A^b(t)E^b(t) - F(t)A^m(t)E^m(t).$$

which we approximate by the stochastic difference equation

$$F_k = F_{k-1} + A^b(k)E^b(k)(F_N - F_{k-1}) - A^m(k)E^m(k)F_{k-1} \tag{2}$$

with parameters

$$\begin{aligned}
A^m(t) &\sim \mathrm{Bern}(\theta^m(t))\,,\\
A^b(t) &\sim \mathrm{Bern}\big(\theta^b(t)\big)\,,\\
E^m(t) &\sim \mathrm{Unif}(0, \gamma^m(t))\,,\\
E^b(t) &\sim \mathrm{Unif}\big(0, \gamma^b(t)\big)\,,
\end{aligned}$$

where $\mathrm{Bern}(\theta)$ indicates the Bernoulli distribution with rate θ and $\mathrm{Unif}(0, \gamma)$ indicates a uniform distribution with lower bound 0 and upper bound γ.

Hence, $\theta^m(t) \in [0, 1]$ is the probability that malware is successful at time t, $\theta^b(t) \in [0, 1]$ is the probability that bonware is successful at time t, $\gamma^m(t) \in (0, 1]$ is the maximum fraction of damage inflicted by malware, and $\gamma^b(t) \in (0, 1]$ is the maximum fraction of damage undone by bonware.

Like the ordinary differential equation (ODE) model, the SDE model allows for a number of interesting variants. In the remainder of this section, we introduce its extension to piecewise linear models.

Table I
MATHEMATICAL MODELS DEVELOPED IN [1]

	Differential Equation	Solution
Continuous Model	$\frac{dF}{dt} + \mathcal{Q}(t)F(t) = F_{\mathrm{N}}\mathcal{B}(t)$	$F(t) = e^{-\int_0^t \mathcal{Q}(p)\,dp}\left(F(0) + F_{\mathrm{N}}\int_0^t e^{\int_0^\tau \mathcal{Q}(p)\,dp}\mathcal{B}(\tau)\,d\tau\right)$
Constant	$\frac{dF}{dt} + \mathcal{Q}F(t) = F_{\mathrm{N}}\mathcal{B}$	$F(t) = \left[F(0) - \frac{F_{\mathrm{N}}\mathcal{B}}{\mathcal{Q}}\right]e^{-\mathcal{Q}t} + \frac{F_{\mathrm{N}}\mathcal{B}}{\mathcal{Q}}$
Piecewise Constant	$\frac{dF}{dt} = \sum_{j=0}^{N-1}(F_{\mathrm{N}} - F(t))\mathcal{B}_j - F(t)\mathcal{M}_j,$ $(t_j \le t < t_{j+1}), \quad (j = 0, \cdots, N-1)$	$F(t) = \left[F(t_{j-1}) - \frac{F_{\mathrm{N}}\mathcal{B}_j}{\mathcal{Q}_j}\right]e^{-\mathcal{Q}_j(t-t_{j-1})} + \frac{F_{\mathrm{N}}\mathcal{B}_j}{\mathcal{Q}_j}$
Linear	$\frac{dF}{dt} + (\lambda - \omega t)F(t) = F_{\mathrm{N}}(\alpha - \beta t)$	$\frac{F(t)}{F_{\mathrm{N}}} = \frac{1}{\Omega(t)}\left\{\frac{F(0)}{F_{\mathrm{N}}} - \frac{\beta}{\omega}(1 - \Omega(t)) + (\alpha\omega - \beta\lambda) \times \frac{\sqrt{\frac{\pi}{2}}e^{\Lambda^2}}{\omega^{3/2}}\left[\mathrm{erf}(\Lambda) + \mathrm{erf}\left(\frac{\omega t}{\sqrt{2\omega}} - \Lambda\right)\right]\right\}$ where $\Omega(t) = e^{\lambda t - \frac{1}{2}\omega t^2}$ and $\Lambda = \lambda/\sqrt{2\omega}$
Piecewise Linear	$\frac{dF}{dt} = \sum_{j=0}^{N-1}\left[(\lambda_j - \omega_j t)F(t) - F_{\mathrm{N}}(\alpha_j - \beta_j t)\right],$ $(t_j \le t < t_{j+1}), \quad (j = 0, \cdots, N-1)$	$\frac{F(t)}{F_{\mathrm{N}}} = \frac{1}{\Omega(t - t_{j-1})}\left\{\frac{F(t_{j-1})}{F_{\mathrm{N}}} - \frac{\beta}{\omega}(1 - \Omega(t - t_{j-1})) + (\alpha\omega - \beta\lambda) \times \frac{\sqrt{\frac{\pi}{2}}e^{\Lambda^2}}{\omega^{3/2}}\left[\mathrm{erf}(\Lambda) + \mathrm{erf}\left(\frac{\omega(t - t_{j-1})}{\sqrt{2\omega}} - \Lambda\right)\right]\right\},$ where $\Omega_j(t) = e^{\lambda_j(t-t_j) - \frac{1}{2}\omega_j(t-t_j)^2}$ and $\Lambda_j = \lambda_j/\sqrt{2\omega_j}$

A. Piecewise linear parameters

In the plot on the left of Figure 2, malware is proportional to the difference of two step functions (malware is active starting from time $t = 0$ seconds and is turned off starting at time $t = 20$ seconds. Various levels of bonware are depicted. Each is expressed as a weighted step function, which turns on at time $t = 20$ seconds. In the plot on the right, malware impact is originally at $\mathcal{M}(0) = 0.5$ and decreases over time, but we enforce $\mathcal{M} \ge 0$. Various linear functions of bonware are shown. In both figures, curves with initial conditions of $\{0, 0.5, 1.0\}$ are illustrated.

In the example depicted in Figure 3, we set:

$$\theta^m(t) = 2\mathcal{M}(t) = \max(0, 0.5 - 0.135t),$$
$$\theta^b(t) = 2\mathcal{B}(t) = \min(1, 0.6 + 0.009t).$$

B. Relationship between continuous and SDE model

With the parameters of the stochastic model selected appropriately, we showed in [1] that as the number of stochastic realizations increases, the expectation of the solution to the stochastic differential equation model approaches that of the ODE model. In [1] we proved the following theorem:

Theorem. *Let* $y_k^m \sim \mathrm{Bern}(2\mathcal{M})$, $y_k^b \sim \mathrm{Bern}(2\mathcal{B})$, $z_k^m \sim \mathrm{Unif}(0, F_k)$, $z_k^b \sim \mathrm{Unif}(0, F_N - F_k)$, *and*

$$F_{k+1} = F_k - y_k^m z_k^m + y_k^b z_k^b, \quad (k = 1, \ldots, K).$$

Let $\mathcal{F}_{kn} = \frac{F_{kj}}{n}$, $(j = 1, \ldots, n)$, *then* $\mathcal{F}_k = \mathbb{E}(F_k) = \lim_{n\to\infty}\mathcal{F}_{kn}$ *and* $\mathcal{F}_k \approx F(k)$, *for large* k, *where* $F(t)$ *is the solution to the initial value problem given by Equation 1 with* $F(0) = \mathcal{F}_0$.

V. GENERATING STOCHASTIC REALIZATIONS

In [1] we developed a method to extract the parameters of the stochastic model associated with a continuous model. In the left pane of Figure 3, we plot a continuous model along with the average over 10,000 runs of its associated stochastic model. After the first two or three

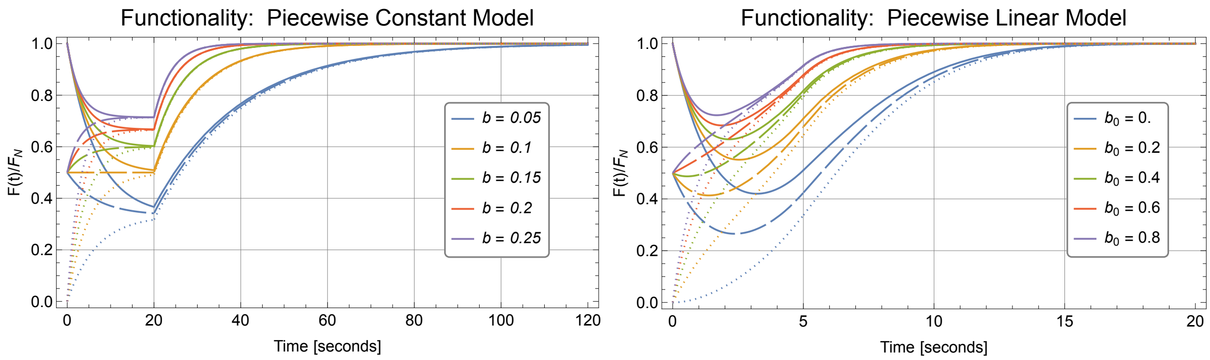

Figure 2. Left: Normalized functionality, $F(t)/F_{\mathrm{N}}$, is shown for piecewise constant models (Row 3 in Table 1). The malware impact is $\mathcal{M}(t) = [u(t) - u(t-20)]$ and the bonware impact is $\mathcal{B} = bu(t-20)$. Initially, malware attacks at mission time $t = 0$. At time $t = 20$ seconds, bonware becomes aware of the attack, counters malware, and brings the system back towards normal operations. Right: Normalized functionality, $F(t)/F_{\mathrm{N}}$, is shown for piecewise linear models (Row 5 in Table 1). The malware impact is $\mathcal{M}(t) = \max(0, 0.5 - 0.1t)$ and the bonware impact is $\mathcal{B} = b_0 + 0.04t$. Both malware and bonware impacts are initially linear functions of time. When malware impact reaches $\mathcal{M} = 0$, then malware impact is zero but bonware impact continues to increase. The function $u(t)$ is the unit step function: $u(t) = 0$ when $t < 0$ and $u(t) = 1$ when $t \geq 0$. The figure on the right is from [1], used by permission.

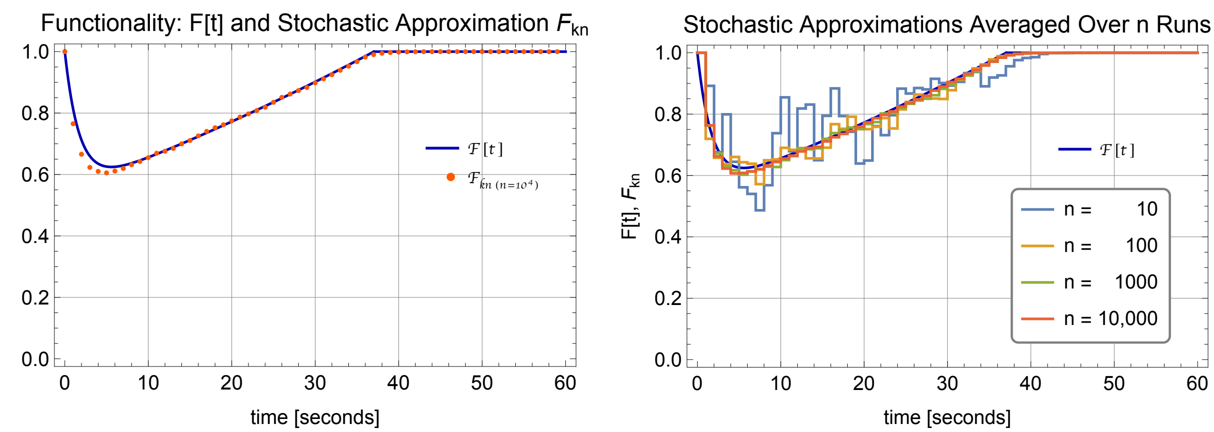

Figure 3. Left: The functionality for a piecewise linear model is shown in blue. $\mathcal{M}(t) = \max(0, 0.25 - 0.00675t)$ and $\mathcal{B}(t) = \min(0.5, 0.3 + 0.0045t)$. The average of 10,000 runs for the corresponding stochastic model is shown in orange. Right: The average of $n \in \{10, 100, 1000, 10, 000\}$ runs. As the number of runs averaged increases, the aggregate stochastic model approaches the solution of the continuous differential equation.

initial points, the agreement is excellent. In the right pane of Figure 3, we show averages of n instantiations of the stochastic model, for $n \in \{10, 100, 1000, 10, 000\}$. As n increases, we see better agreement with the solution to the continuous differential equation model. In Figure 4, we plot the absolute error between the two models. After the first five seconds of data, the absolute error between the continuous model solution and the average of the stochastic ensemble is less than 0.01.

VI. Parameter Estimation

Given the deterministic solutions to the stochastic differential equations presented above (see Eq. 2), we can now proceed to the estimation of model parameters from data.

The parameter estimation has two components: (a) a loss function that describes the distance between the observed data Y and the data implied by the model with parameter vector Θ, and (b) an optimization routine to find those values for Θ that minimize the loss.

We considered two loss functions, depending on whether the performance data are constrained to a fixed domain or not. First, we considered the case where our performance data is expressed as a fraction of optimal performance and constrained to the open $(0, 1)$ interval. For such cases, a convenient option is the *beta loss* function [5]:

$$L_\beta(\Theta) = -\sum_k F_k(\Theta) \log(Y_k) - \sum_k (1 - F_k(\Theta)) \log(1 - Y_k),$$

where $F_k(\Theta)$ is the model-predicted performance at the k^{th} observation, given parameter vector Θ; and Y_k is the k^{th} observed performance value. $F_k(\Theta)$ can be calculated as the deterministic solution to the SDE as in

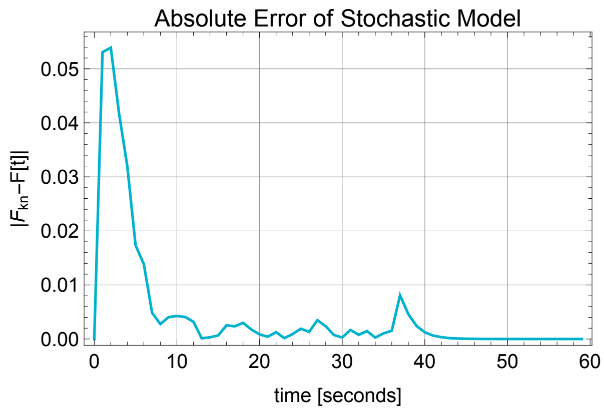

Figure 4. The absolute error of the stochastic model calculated as the absolute value of the difference between the solution to the differential equation and the average of 10,000 runs.

Equation 2, if that solution is available. Alternatively, if such a solution is unavailable or cumbersome to compute, $F_k(\Theta)$ could be numerically approximated using an iterative method such as the forward Euler method or other Runge-Kutta methods [6].

However, in our applications, the data are not usually restricted to this limited domain – nominal performance levels are often unknown or variable, and when they are known, performance is often at 100%. Therefore, in practice, we will use a *squared error loss* function [5]:

$$L_s(\Theta) = \sum_k (Y_k - F_k(\Theta))^2 .$$

The parameter vector Θ itself usually contains at least some elements that are restricted to a limited domain (e.g., the bonware and malware effectiveness parameters are strictly positive). This leaves us with a complex constrained optimization problem. Fortunately, there exist several well-known algorithms that can quickly find optima of loss functions under constraint, especially if the dimensionality of Θ is low, as it is here. We opted for a Nelder-Mead simplex optimization algorithm [7], which is fast, robust, and easy to implement. The Nelder-Mead simplex procedure is implemented in MATLAB's `fminsearch`, R's `optim`, and Python's `scipy.minimize`. The optimization procedure will yield the estimated parameter vector $\hat{\Theta} = \arg\min_{\Theta} L_s(\Theta)$.

VII. Next Steps

The next steps in the development of our technology for quantifying cyber resilience will be to test the efficiency and precision of these estimation models through numerical experiments (i.e., simulation studies). While the methods we use are well established, their application to the piecewise linear SDE models is not.

One specific issue to examine is that of *mimicry* – a phenomenon where multiple distinct combinations of parameters yield predictions that are impossible to distinguish with the available amounts of data. Figure 5 illustrates this issue. Here, we generated 10 runs of 60 observations using the same parameters as those used to generate Figure 3: $\mathcal{M}(t) = \max\{0, m_0 - m_1 t\}$ and $\mathcal{B}(t) = \min\{0.5, b_0 + b_1 t\}$ with $m_0 = 0.25$, $m_1 = 0.00675$, $b_0 = 0.3$, and $b_1 = 0.0045$. However, the estimated parameters were $\hat{m}_0 \approx 0.3146$, $\hat{m}_1 \approx 0.5227$, $\hat{b}_0 \approx 0.5447$, and $\hat{b}_1 \approx 0.5568$. Despite these clear differences, the model captures the fast-downward-then-slow-upward pattern in the data well. This indicates that the parameters of this model may be difficult to identify given the other qualities of the data (such as the sample size and magnitude of the random variability) or certain specific features of the generating parameter sets (e.g., the relative timing of the effective onset of malware and bonware).

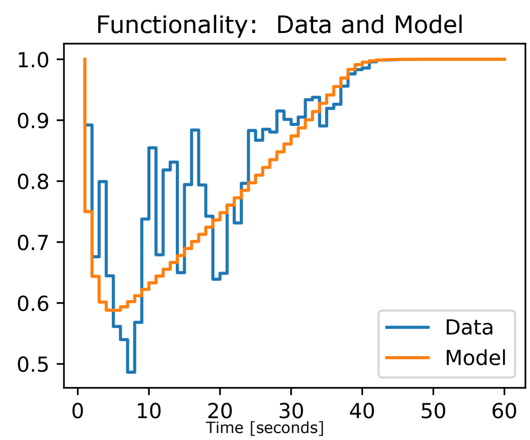

Figure 5. Model fit to the average of 10 runs. The model captures the general trend of the data well despite the fact that estimated parameters are quite different from the generating parameters.

VIII. Linear Filtering

Obtaining an initial approximation of the parameters of the stochastic difference equation (2), and positing an estimate of their uncertains, we can apply linear filtering theory, and frame our estimation problem as a Kalman-Bucy filter [8]. Our piecewise linear model includes abrupt changes in the trajectories of both malware and bonware. These changes can be accounted for by allowing random maneuvers in the "target dynamics" [9] or by employing interacting multiple models (IMM) [10] [11] where multiple filters track the parameters and a mechanism is established to choose which filter is most appropriate at each time increment [12]. For each of a finite set of parameters governing both malware and bonware impact, a filter may be established, and transitions can be tracked by evaluating the relative performance of these filters. A related approach [13] is to view our system as a linear stochastic system with unknown jumps (changes in the linear models governing the malware and bonware impacts). Since the changes occur infrequently, a monitoring system is set up to monitor filter residuals. When these become large, an adjustment is made to the filter.

IX. Discussion and conclusion

In [1], we have presented a broadly applicable framework for the analysis of the cyber resilience of military artifacts. Our framework relies on the construction of a custom differential equation time series model that shows good qualitative correspondence to the functionality of vehicles performing missions.

Both types of models can be extended to a large variety of custom circumstances, including the case where model parameters change gradually, abruptly, or predictably as a result of experimental manipulation. In this paper, we have extended the stochastic model to include piecewise linear malware and bonware activities. The piecewise linear model is mathematically tractable and can be used to generate data that qualitatively correspond to performance data in our lab tests. However, further analysis is needed to establish the conditions under which parameters of the model can be reliably estimated.

References

[1] A. Kott, M. Weisman, and J. Vandekerckhove, "Mathematical modeling of cyber resilience," *Proceedings of IEEE Military Communications Conference*, pp. 835–840, Dec. 2022.

[2] J. Ellis, T. Parker, J. Vandekerckhove, B. Murphy, S. Smith, A. Kott, and M. Weisman, "Experimental infrastructure for study of measurements of resilience," *Proceedings of IEEE Military Communications Conference*, pp. 841–846, Dec. 2022.

[3] S. Hosseini, K. Barker, and J. E. Ramirez-Marquez, "A review of definitions and measures of system resilience," *Reliability Engineering & System Safety*, vol. 145, pp. 47–61, 2016.

[4] A. Kott and I. Linkov, "To improve cyber resilience, measure it," *Computer*, vol. 54, no. 2, pp. 80–85, Feb. 2021.

[5] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd ed. New York, NY: Springer International Publishing, 2009.

[6] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical recipes 3rd edition: The art of scientific computing*. Cambridge university press, 2007.

[7] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. Vol. 7, pp. 308–313, 1965.

[8] A. H. Jazwinski, *Stochastic Processes and Filtering Theory*. Academic Press, 1970.

[9] E. Brookner, *Tracking and Kalman Filtering Made Easy*. Wiley-Interscience, 1998.

[10] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.

[11] H. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, Sept. 1988.

[12] A. S. Willsky, "Detection of abrupt changes in dynamic systems," in *Detection of Abrupt Changes in Signals and Dynamical Systems*, M. Basseville and A. Benveniste, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 1986, pp. 27–49.

[13] A. S. Willsky and H. L. Jones, "A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems," *IEEE Transactions on Automatic Control*, vol. AC-21, pp. 108–112, Feb. 1976.