ESC-TR-2007-066

Technical Report
1134

Validating the Modeling and Simulation of a Generic Tracking Radar

MIT LINCOLN LABORATORY
0022
G18D268865A

H.C. Lambert
S.R. Vogl
A.S. Brewster
K-P. Dunn

28 July 2009

Lincoln Laboratory
MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LEXINGTON, MASSACHUSETTS

Prepared for the Missile Defense Agency under Air Force Contract FA8721-05 C-0002.

Approved for public release; distribution is unlimited.

20090803013

This report is based on studies performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. This work was sponsored by the Missile Defense Agency, SN, under Air Force Contract FA8721-05-C-0002.

This report may be reproduced to satisfy needs of U.S. Government agencies.

The ESC Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Gary Tutungian
Administrative Contracting Officer
Plans and Programs Directorate
Contracted Support Management

Non-Lincoln Recipients

PLEASE DO NOT RETURN

Permission has been given to destroy this document when it is no longer needed.

This page intentionally left blank.

Massachusetts Institute of Technology
Lincoln Laboratory

Validating the Modeling and Simulation of a Generic Tracking Radar

H.C. Lambert
S.R. Vogel
A.S. Brewster
Group 32

K-P. Dunn
Group 36

Technical Report 1134

28 July 2009

Approved for public release; distribution is unlimited.

Lexington Massachusetts

This report is based on studies performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. This work was sponsored by the Missile Defense Agency, SN, under Air Force Contract FA8721-05-C-0002.

This report may be reproduced to satisfy needs of U.S. Government agencies.

The ESC Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Gary Tutungian
Administrative Contracting Officer
Plans and Programs Directorate
Contracted Support Management

Non-Lincoln Recipients

PLEASE DO NOT RETURN

Permission has been given to destroy this document when it is no longer needed.

EXECUTIVE SUMMARY

This report proposes acceptability criteria for validating the modeling and simulation of a generic tracking radar. The validation process is limited to the comparison of a set of Monte Carlo realizations of the simulated time series of judiciously selected validation metrics with *single* discrete-event observations made by the actual system. Our approach is based on a statistical hypothesis test. The two hypotheses are (1) the hypothesis that the simulation is *consistent* with actual system performance—the null hypothesis, H_0, and (2) the hypothesis that the simulation is *inconsistent* with actual system performance—the alternative hypothesis, H_1. The proposed procedure is cognizant of the so-called model maker's risk, α, and the so-called model user's risk, β, corresponding to the probabilities of Type I and Type II errors, respectively. For each validation metric, we count the number of samples of the *observed* time series that fall outside of bounds prescribed by the Monte Carlo realizations of the simulated time series. Subsequently, if the number of observed samples that are outside of the simulation bounds are *above* a pre-computed rejection threshold, γ, computed based on a pre-specified model maker's risk, α, we declare the simulated time series of the particular validation metric under scrutiny as *inconsistent* with the observed time series. Any statistical dependence present in the time series of the validation metrics is accounted for in the computation of the rejection threshold, γ. The number of Monte Carlo realizations also impacts the computation of γ.

Results are summarized in a so-called scorecard. For each discrete-event observation, the scorecard contains a list of rejection indices for the different validation metrics, with each rejection index—expressed as a number between 0 and 100—denoting the ratio of the samples of the observed time series of the associated validation metric that are outside of the simulation bounds. Normalized rejection thresholds for the different validation metrics—also expressed as numbers between 0 and 100—are also included in the scorecard. The scorecard reveals any cross-correlation that exists among select validation metrics. Due to the unavailability of the probability density function of the observed behavior, which prevents us from computing the model user's risk, β, we require that a *family* of normalized rejection thresholds, corresponding to different values of the model maker's risk, α, be included in the scorecard. Using sound judgement and common sense, a validation agent may apply the scorecard to accept or reject a given modeling and simulation product. The scorecard has the added advantage of serving as a diagnostic tool—thus aiding in modeling and simulation *improvement*.

This page intentionally left blank.

ACKNOWLEDGMENTS

The authors wish to thank R. L. Bellaire, S. K. Simhal, and J. E. Dennis of Missile Defense Agency Sensors Directorate; K. T. Ryals of Johns Hopkins University Applied Physical Laboratory; S. S. Krigman and V. E. Burns of Raytheon Company; and J. A. Tabaczynski, L. J. Maciel, D. A. O'Connor, C-B. Chang, B. A. Telfer, J. T. Mayhan, E. V. Rossi, and S-H. Son of MIT Lincoln Laboratory for stimulating discussions and helpful suggestions.

This page intentionally left blank.

TABLE OF CONTENTS

		Page
	Executive Summary	iii
	Acknowledgments	v
	List of Illustrations	ix
	List of Tables	ix
1.	INTRODUCTION	1
2.	STATISTICAL HYPOTHESIS TESTING	3
	2.1 The Lazy Model Maker's Paradox	5
3.	CORRELATED TIME SERIES	7
	3.1 How Many Monte Carlo Realizations?	14
4.	ACCEPTABILITY CRITERIA	17
5.	CASE STUDY	21
6.	SUMMARY	33
	References	37

This page intentionally left blank.

LIST OF ILLUSTRATIONS

Figure No.		Page
1	First-Order Gauss–Markov Process	8
2	Monte Carlo Trials of a Generic Validation Metric	9
3	Variation of the Rejection Index	13
4	Variation of the Normalized Rejection Threshold	15
5	Variation of the Model Maker's Risk	15
6	Time Series for the Perfectly Matched Scenario	22
7	Scorecard for the Perfectly Matched Scenario	23
8	Mismatched Target SNR and RCS	25
9	Time Series for the Mismatched Target Scenario	26
10	Scorecard for the Mismatched Target Scenario	27
11	Mismatched Measurement Error	29
12	Time Series for the Mismatched Environment Scenario	30
13	Scorecard for the Mismatched Environment Scenario	31

LIST OF TABLES

Table No.		Page
1	Validation Metrics	18

This page intentionally left blank.

1. INTRODUCTION

The best way to test the performance of a sensor is to repeatedly conduct experiments using that sensor. For example, in the case of a tracking radar, we would collect measurements originating from a known target and, using the tracking filter implemented within the radar software, form a track based on those measurements. We would then compute the target state estimation error with reference to the target's true state—known *a priori*—and evaluate the performance of the tracking filter using the tried and true statistical methods discussed in classic textbooks such as [1]. While optimal, such experiments are, unfortunately, often not cost-effective. Worse, they are almost never repeatable. For example, we cannot expect the environment in which the sensor operates to remain constant—varying weather conditions being a favorite anecdote. Therefore, it is often more economical to operate within a *simulated* environment, wherein experiments can be tightly controlled and repeated ad nauseam. In order for the simulation to be trusted as a proxy for the observed behavior, we need to have a way of evaluating the accuracy of the models enabling the simulation. The discipline of modeling and simulation (M&S) verification, validation, and accreditation (VV&A) is as much an art form as science. A lucid and sobering account of many remaining M&S VV&A challenges can be found in [2].

In this report, we focus on validating the modeling and simulation of a generic tracking radar. The proposed validation criteria can be extended to other radar functions as well. For rigor's sake, we abide by the following definitions from [3]:

- **Verification:** "The process of determining that a model implementation and its associated data accurately represents the developer's conceptual description and specifications."
- **Validation:** "The process of determining the degree to which a model and its associated data are an accurate representation of the real world from the perspective of the intended uses of the model."
- **Accreditation:** "The official certification that a model, simulation, or federation of models and simulations and its [sic] associated data are acceptable for use for a specific purpose."
- **Acceptability Criteria:** "A set of standards that a particular model, simulation, or federation must meet to be accredited for a specific purpose."

The purpose of this report is to address specifically the design of acceptability criteria appropriate for validating the modeling and simulation of a generic tracking radar following the simulation validation guidelines provided in [4]. The techniques we propose would directly benefit the "validation agent," who, according to [3], is "[t]he person or organization designated to perform validation of a model, simulation, or federation of models and/or simulations and the associated data."

In an effort to devise effective acceptability criteria, we aim to satisfy three objectives. First, we note the crucial point that the modeling and simulation product must be able to replicate the sensor's behavior *irrespective of its performance*. In other words, if the sensor is expected to perform poorly under certain conditions, then we would like the modeling and simulation of the sensor to

replicate the same poor performance—otherwise, for testing purposes, we would not be able to rely on the simulation as a true surrogate for the sensor. Thus, the model maker must not confuse sensor performance with sensor performance *replication*. Unfortunately, in our experience, many a good model maker has fallen prey to an inability to make this important distinction, a phenomenon we refer to as the *model maker's fallacy*. Such a fallacy tends to occur more frequently when the model maker is also the equipment maker.

Our second objective is to ensure that the acceptability criteria for validating the modeling and simulation of a given sensor are "anchored" to the behavior observed by that sensor—such as observations of targets of opportunity in the case of tracking radars. Specifically, we aim at validating the repeated behavior exhibited by a given modeling and simulation product with a *single* discrete-event observation—say, of a single satellite pass in the case of a tracking radar—through the use of sound statistical techniques. Unfortunately, we almost never have access to the probability distribution functions of uncertainties affecting the observed behavior of the sensor. Nevertheless, we must devise criteria that minimize the risk to the model user—or more precisely the validation agent who is responsible for passing or failing a given modeling and simulation product.

Our third, and possibly most important, objective is that the requisite validation metrics should actually aid in *improving* the modeling and simulation of the sensor. In other words, we seek to devise a set of acceptability criteria that not only would allow us to pass or fail a given modeling and simulation product, but also, in case of failure, would serve as a *diagnostic tool* to help us identify the sources of failure. By satisfying this objective, the validation agent will be able to make a more informed decision about the overall performance of the modeling and simulation product. When acceptability criteria are tied to "physics," it becomes easier to identify statistical outliers, and their impact on the simulation validation process can thus be minimized.

Our treatise begins with an outline of a statistical decision theoretic method to modeling and simulation validation in Section 2. Here, we discuss risks and benefits from the point of views of the model maker and the model user. A decision theoretic approach to modeling and simulation validation is by no means original (see [5] for a summary of approaches). Of particular value is an extension of the statistical testing procedure for the equality of the power spectral densities of multiple short memory time series devised by [6] to modeling and simulation validation. The modeling and simulation validation approach proposed in this report is different and unique in that it combines results from a specific statistical hypothesis test with physical constraints imposed by judiciously selected validation metrics to allow for an informed and efficient decision making process with the added bonus of providing a road map for modeling and simulation improvement.

In Section 3, we elaborate on how to account for any statistical dependence that is present in the time series of the validation metrics relevant to the modeling and simulation of a tracking radar. In this section, we also answer the often-asked question: "How many Monte Carlo realizations are sufficient to validate a given modeling and simulation product using the method proposed in this report?" We list the validation metrics relevant to the modeling and simulation of a tracking radar in Section 4. Via a controlled numerical experiment, we examine the effectiveness of the proposed method in Section 5. A summary of our results is given in Section 6.

2. STATISTICAL HYPOTHESIS TESTING

We can formulate the simulation validation process as a statistical hypothesis test [5]. We consider two hypotheses. We define the *null hypothesis*, H_0, to be the hypothesis that the simulation is *consistent* with actual system performance, while we define the *alternative hypothesis*, H_1, to be the hypothesis that the simulation is *inconsistent* with actual system performance. Therefore, we would accept a valid simulation when H_0 is true, and we would reject an invalid simulation when H_1 is true. However, due to the statistical nature of the problem, two types of decision error can arise. The so-called Type I error would correspond to *rejecting a valid simulation*, while the so-called Type II error would correspond to *accepting an invalid simulation*. In the modeling and simulation literature, the probability, α, of Type I error is often referred to as the *model maker's risk*, and the probability, β, of Type II error is often referred to as the *model user's risk* [5].

The validation problem lies in "detecting" a simulation that is inconsistent with actual system performance. For an optimal solution, one could, in theory, invoke the Neyman-Pearson theorem to devise a "detector" that minimizes the model user's risk, β, for a given model maker's risk, α [7]. In other words, the model maker's risk, α, is treated as a parameter of the decision problem; it is used to compute a "rejection threshold," γ, for an appropriate "test statistic" of a chosen "validation metric." If the test statistic is observed to *exceed* the rejection threshold, then, for the particular validation metric under consideration, the simulation is deemed to be *inconsistent* with actual system performance. When there are more than a single metric to be considered, correlations among the metrics must be taken into account. For modeling and simulation of tracking radars, validation metrics come in the form of *time series*—as opposed to single scalars. Hence, temporal correlations present in the time series must also be taken into account. A list of the metrics proposed for the validation of a given tracking radar simulation is given in Table 1, Section 4.

The computation of the likelihood ratio needed for the design of a Neyman-Pearson detector demands an *a priori* knowledge of the probability distribution functions (PDFs) of both the simulation and the actual system results—at least to within a normalizing constant. Generally, we do not have access to an accurate representation of the PDF of actual system results. Due to the nonlinear nature of the models and the presence of a large number of random contributors, we often have no choice but to resort to Monte Carlo sampling techniques to derive PDFs numerically. While simulations are repeatable, experiments involving actual systems might not be. This is certainly the case for experiments involving tracking radars. Hence, we have no choice but to treat the PDF of the actual system results as *unknown*. Fortunately, we can still compute a rejection threshold, γ, based on a given model maker's risk, α, since the computation of γ depends only on the PDF of the simulation results [7], which can in general be estimated from a histogram of the Monte Carlo samples. However, since we do not have access to the PDF of the actual system results, we cannot guarantee the model *user*'s risk, β, to be a minimum.

This report presents a procedure, inspired by the aforementioned decision theoretic concepts, in which multiple Monte Carlo realizations of time series corresponding to key validation metrics—obtained by running the simulation software multiple times—are compared with results obtained by the actual system during a *single* discrete-event observation. In order to apply this procedure, we begin by counting, for each validation metric, the number of times that *independently sampled* values

of the corresponding time series observed by the actual system fall outside of bounds prescribed by the simulation. The simulation boundaries are set to the minimum and maximum values of the Monte Carlo realizations of the time series valid at each time index of the observed values. We could have set the simulation boundaries to n standard deviations about the mean of the Monte Carlo realizations. However, this approach would be accurate only for validation metrics that have a Gaussian probability density function. Unfortunately, many of the validation metrics, such as the total position error listed in Table 1, Section 4, have probability density functions that are significantly different from the Gaussian PDF. We thus opt for setting the simulation bounds to the minimum and maximum values of the Monte Carlo realizations in lieu of setting bounds based on explicitly derived PDFs.

The simulation is declared to be *inconsistent* with actual system performance if the number of times that *independent* samples of the observed time series fall outside of the simulation bounds *exceeds* a pre-computed rejection threshold, γ. Since the samples are chosen to be statistically independent, the outcome of this process can be modeled with a binomial random variable, with cumulative mass function:

$$\Pr\{x \leq n\} = \sum_{k=0}^{n} \binom{N_i}{k} p^k (1-p)^{N_i - k}. \tag{1}$$

The form of the cumulative mass function depends on the number, N_i, of independently sampled values of the time series associated with the validation metric under scrutiny and on the probability, p, that a single sample of the observed time series falls outside of the simulation bounds. As a result, the mapping of the model maker's risk, α, to the rejection threshold, γ, also depends on these parameters.

At the time of simulation validation, the number of Monte Carlo trials is invariant; that is, the validation agent is given a *fixed* set of Monte Carlo realizations of the time series of the validation metrics, along with a single time series observed by the actual system. From the number, N_{MC}, of Monte Carlo trials, we can easily show that the probability, p, that a *single* sample of the time series of a given validation metric observed by the actual system falls outside of bounds prescribed by the Monte Carlo realizations is given by the simple expression

$$p = \frac{2}{N_{\mathrm{MC}} + 1}. \tag{2}$$

In order to judge whether the entire *history* of the sampled values of the observed time series are outside of the simulation bounds, we need to know something about the *statistical dependence* of those samples. In other words, we must account for any temporal correlations present in the time series of the scrutinized metrics in order to perform a meaningful, fair, and robust test. In the following section, we give a detailed account of the impact of temporally correlated time series on the simulation validation procedure. However, before handling correlated time series, we must first address an apparent concern with regard to a simulation validation procedure that is based on bounds prescribed by the simulation itself.

2.1 THE LAZY MODEL MAKER'S PARADOX

One might be tempted to think that the passing or failing of a given modeling and simulation product based on bounds set by the simulation itself would allow the model maker to devise a model that could be guaranteed to be consistent with all observations at all times. Consider the following gedankenexperiment. Given the modeling and simulation validation procedure outlined above, a lazy model maker, in an attempt to guarantee success, may naïvely decide to broaden the probability density functions of model uncertainties impacting the simulation output. This way, the time series of the validation metrics observed by the actual system would always fall within the simulation bounds—or so the model maker hopes. For example, in the case of a tracking radar, in order to avoid the explicit modeling of unanticipated systematic errors, such as temporally correlated measurement errors induced by the random heaving and tilting motion experienced by a shipboard radar, the model maker may simply decide to reduce the signal-to-noise ratio driving the measurement error variances. This way, temporally varying biases would be buried in noise, and the model, in a way, is guaranteed to be accepted by the validation agent. However, by doing so, if the model maker is also the equipment maker, he or she would be admitting that his or her equipment's performance is at best subpar. After all, a radar advertised—through behavior demonstrated via modeling and simulation—as having a poor signal-to-noise ratio would not be a desirable item to own. It follows that such a strategy would prove unwise if the model maker is also the equipment maker who wishes to sell the equipment. Thus, the model maker has no choice but to properly account for *all* sources of errors, including time-varying biases.

This page intentionally left blank.

3. CORRELATED TIME SERIES

The number, N_i, of independent samples in the time series of a validation metric plays an important role in the decision algorithm presented in the previous section. It can be estimated approximately by dividing the total duration, T, of the time series by the correlation time, τ, of the time series:

$$N_i \simeq \frac{T}{\tau}. \tag{3}$$

In other words, if we resample the time series at a rate of approximately $1/\tau$, then the N_i resulting samples are statistically independent. The correlation time, τ, can be obtained by employing any of the classical techniques discussed in the vast literature on time series analysis. For example, we could estimate τ from the *autocorrelation function* of the time series. The correlation time would then correspond to the point in time when the autocorrelation function falls, say, to $1/e$ times its maximum value at zero delay. Alternatively, we could estimate the correlation time from the *power spectral density* (PSD) of the time series, which is defined as the Fourier transform of the autocorrelation function. In that case, the correlation time, τ, would correspond to the inverse of an appropriately defined "roll-off frequency," $\nu = 1/\tau$, of the PSD. What if there are more than a single correlation time? In that case, we would resample the time series at a rate equal to one over the *longest correlation time*. That way, the resulting samples are guaranteed to be statistically independent.

For a stochastic time series, ψ, it can be shown that the PSD can be obtained directly from the Fourier transform, Ψ, of the time series through the relation [8]:

$$\mathrm{PSD}(f) = \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[|\Psi(f)|^2\right], \tag{4}$$

where f is the frequency, and $\mathbb{E}(\cdot)$ denotes the expected value of $(\cdot)$. In the case of *uniform sampling*, the Fourier integral can be approximated by the discrete Fourier transform (DFT) [9]:

$$\Psi(f_n) \simeq \Delta \sum_{k=0}^{N-1} \psi(t_k) \exp\left(\frac{i2\pi kn}{N}\right), \quad n = -\frac{N}{2}, \ldots, \frac{N}{2} - 1, \tag{5}$$

where Δ denotes the *uniform* sampling interval, and N is the *total* number of samples. The DFT for a uniformly sampled time series can be implemented using the efficient fast Fourier transform (FFT) algorithm [9]. Hence, it follows that, for large T, we can approximate the PSD by

$$\mathrm{PSD}(f_n) \simeq \frac{1}{T}\left[\Delta \sum_{k=0}^{N-1} \psi(t_k) \exp\left(\frac{i2\pi kn}{N}\right)\right]^2, \quad n = -\frac{N}{2}, \ldots, \frac{N}{2} - 1. \tag{6}$$

For *non-uniformly sampled* time series, more sophisticated techniques, such as the Lomb periodogram method [10], must be considered for the estimation of the PSD.

For illustration, we consider a time series prescribed by a first-order Gauss–Markov process [11]. The theoretical expression for the PSD of a first-order Gauss–Markov process with

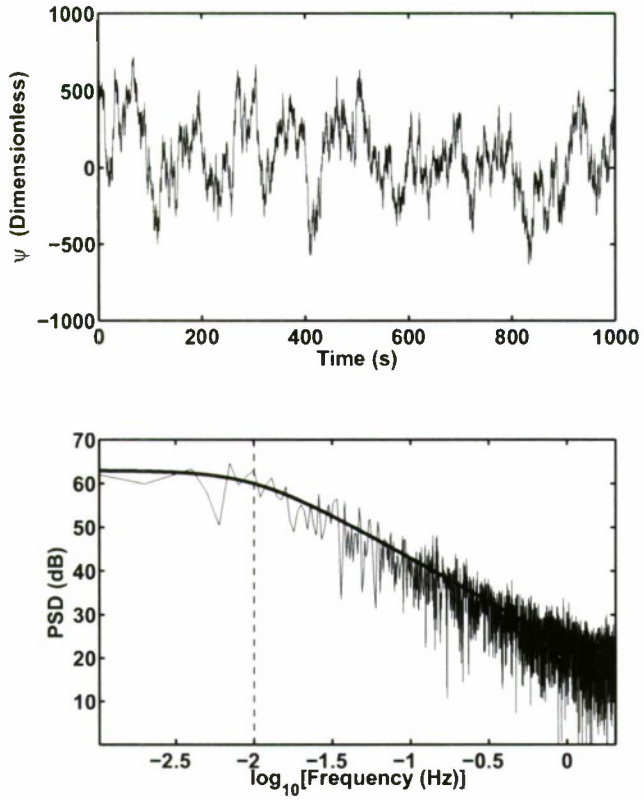

Figure 1. *A realization of a first-order Gauss–Markov time series, ψ, with standard deviation $\sigma_{GM} = 250$ units and correlation time $\tau_{GM} = 100$ s, and its power spectral density (PSD). The smooth curve in the bottom panel is obtained from the theoretical expression for the PSD of a first-order Gauss–Markov process with similar parameters. The dashed vertical line indicates the "roll-off frequency" of the PSD, corresponding to $1/\tau_{GM} = 0.01$ Hz.*

standard deviation σ_{GM} and correlation time $\tau_{\mathrm{GM}} = 1/\nu_{\mathrm{GM}}$ is given by [11]

$$\mathrm{PSD}_{\mathrm{GM}}(f) = \frac{\nu_{\mathrm{GM}}\sigma_{\mathrm{GM}}^2}{\pi\left(f^2 + \nu_{\mathrm{GM}}^2\right)}. \tag{7}$$

A realization, ψ, of the first-order Gauss–Markov time series with $\sigma_{\mathrm{GM}} = 250$ units and $\tau_{\mathrm{GM}} = 10$ s is shown in the top panel of Figure 1. The time series, ψ, can represent any validation metric. Thus, for convenience, we have chosen ψ to be dimensionless. The PSD computed directly from the time series, using Eq. (6), along with the theoretical PSD computed from Eq. (7) is shown in the bottom panel of Figure 1. The theoretical value of the inverse of the correlation time, $\nu_{\mathrm{GM}} = 0.01$ Hz, is shown with the dashed vertical line. It follows that if we resample the time series shown in the top panel of Figure 1 at a rate of 0.01 Hz, then the resulting sequence will correspond to a *white Gaussian noise sequence.*

To demonstrate how temporal correlations can be taken into account, we consider the hypothetical scenario depicted in Figure 2. Here, we are given a set of $N_{\mathrm{MC}} = 10$ Monte Carlo realizations, shown in black, of the time series of a generic validation metric, ψ, described by a

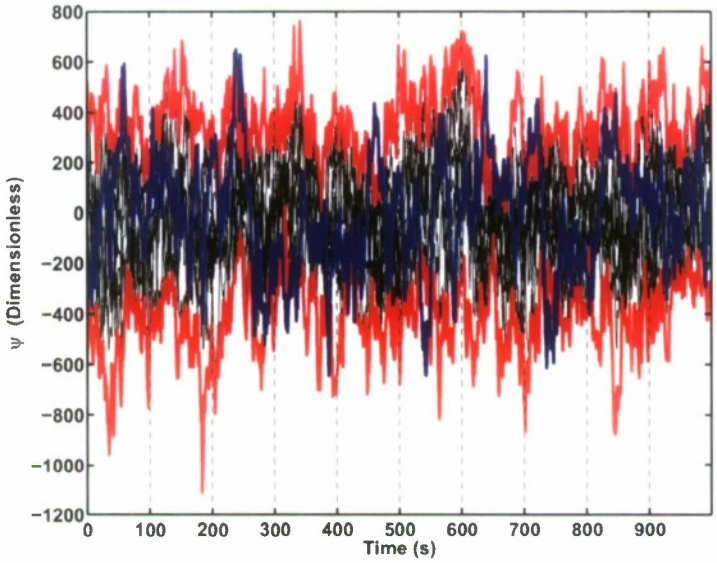

Figure 2. *Ten Monte Carlo realizations of the time series of a generic validation metric, ψ, shown in black, and a single time series of the validation metric observed by the actual system, shown in blue. The simulated and the observed time series are described by the same first-order Gauss–Markov process, with standard deviation $\sigma_{GM} = 250$ units and correlation time $\tau_{GM} = 100$ s. The simulation bounds, prescribed by the minimum and maximum values of the 10 Monte Carlo realizations, are shown in red. The dashed vertical lines separate the so-called correlation segments.*

first-order Gauss–Markov process with $\sigma_{\mathrm{GM}} = 250$ units and $\tau_{\mathrm{GM}} = 100$ s. Furthermore, we consider the case when the time series observed by the actual system, shown in blue in Figure 2, is described by the *same* first-order Gauss–Markov process. In other words, the null hypothesis, H_0, corresponding to the hypothesis that the simulation is consistent with actual system performance, is the true hypothesis. Thus, for our example, we have simply generated $N_{\mathrm{MC}} + 1 = 11$ Monte Carlo realizations of the same first-order Gauss–Markov process and have arbitrarily labeled one of them as the time series observed by the actual system. However, due to the random nature of the problem, there is always a chance that we may decide that the *alternative hypothesis*, H_1, corresponding to the hypothesis that the simulation is *inconsistent* with actual system performance, is the true hypothesis. In that case, we would reject a valid simulation. In binary hypothesis testing, this type of error is referred to as the Type I error, and its probability, referred to as the model maker's risk, α, in the modeling and simulation literature, serves as a parameter of the decision algorithm.

In order to determine the number, N_i, of independent samples, we must first estimate the correlation time, τ, of the time series, ψ. Subsequently, the number of independent samples can be obtained from Eq. (3). Since the validation process is anchored to the simulation, the correlation time, τ, ought to be computed from the simulated time series, instead of from the time series observed by the actual system. We can estimate the correlation time, τ, by using any of the techniques discussed earlier. Since the simulated time series are drawn from the same probability

distribution function, we can reduce the error in estimating the correlation time by combining results obtained from the independent processing of the individual time series. For example, if we decided to estimate the correlation time, τ, from the PSD, then the PSDs computed separately for each of the $N_{\mathrm{MC}} = 10$ Monte Carlo realizations shown in Figure 2—say using Eq. (6)—can be averaged to result in a smoother estimate of the PSD, thereby reducing the error in estimating the roll-off frequency, $\nu = 1/\tau$.

Once we have estimated the correlation time, τ, from the simulated time series, we can divide the time interval including both the simulated and the observed time series into so-called correlation segments, each of duration τ. The correlation segments are shown separated with dashed vertical lines in Figure 2. For the simulated time series, the samples within each segment are correlated, whereas samples selected from different segments and separated by at least one correlation time, τ, are statistically independent. If the true hypothesis is the null hypothesis, H_0, corresponding to the hypothesis that the simulation is consistent with actual system performance, then the same behavior is obtained for the observed time series. In other words, in case of H_0 being the true hypothesis, if, for *any* time series (whether simulated or observed), we pick a sample from each correlation segment, then, as long as the selected samples are as least one correlation time apart, the resulting sequence will be a white Gaussian noise sequence.

The simulation bounds, prescribed by the minimum and maximum values of 10 Monte Carlo realizations, are shown in red in Figure 2. Given a sequence of τ-separated samples of the *observed* time series and simulation bounds valid at the times of the selected samples, we count the number of times the samples fall outside of those bounds. Next, we compare the number of samples that fall outside of the simulation bounds with a rejection threshold, γ. The rejection threshold is determined from the number, N_i, of independent samples; the probability, p, of an arbitrary sample falling outside of the simulation bounds; and the model maker's risk, α. If the number of τ-separated samples falling outside of the simulation bounds is smaller than the rejection threshold, γ, then we declare the simulation to be consistent with actual system performance. If the true hypothesis is the alternative hypothesis, H_1, then there is no reason to expect the observed time series to behave in a way predicted by the simulation. Specifically, resampling the observed time series at a rate of $1/\tau$, where the correlation time, τ, is estimated from the *simulated* time series, may not result in an uncorrelated sequence. However, this discrepancy can only add to the inconsistency between the simulation and actual system performance and would therefore not degrade the performance of the decision algorithm.

The choice of which sample to pick as the starting point of the resampling process is somewhat arbitrary. For instance, we could choose to always pick the *first* sample in each of the correlation segments shown in Figure 2. Alternatively, we could haven chosen to always pick the *second* sample in each of the correlation segments, or the *third* sample, and so on. Once we have committed to a particular starting point, we may begin to wonder whether the ignored samples might have afforded any further utility. Also, what if our chosen starting point happens to produce a sequence that corresponds to a statistical outlier, thereby skewing the validation process, whereas had we chosen another starting point, might we have obtained a more normative sequence? One way to remedy such quandaries would be to consider *all* possible starting points: resampling the observed time series by picking the first sample in each correlation segment, followed by resampling the observed

time series by picking the second sample in each correlation segment, etc., until we reached the end of the correlation segments. We could then report our result based on an *average* of all possible cases, along with an appropriate confidence interval. While such an approach might present a viable solution, we opt for a simpler procedure.

Instead of partitioning the data window into correlation segments and subsequently picking τ-separated samples of the observed time series from each segment, we consider *all* samples instead; that is, we choose to *ignore* any correlation that may exist between the samples. Specifically, we examine whether *any* of all samples fall inside or outside of the simulation bounds. Of course, by doing so, we would introduce an error, since the invocation of the binomial probability distribution function requires the samples to be statistically independent. However, if correlation effects are taken into account in the computation of the rejection threshold, γ, then, we argue, the effect of this error on the validation process will be innocuous. In other words, we would reach the same decision on the validity of a simulation had we incorporated the correlation effects in the averaging process discussed above. This method has the advantage of using all the available data without the need to resort to any complicated counting procedure or averaging. We therefore regard it as more practical.

We illustrate the concept by employing a first-order Gauss–Markov process modeling the time series of a generic validation metric. All time series contain $N = 1024$ samples and are sampled uniformly at a rate of 1 Hz—thus $T = 1024$ s. The correlation time is increased from $\tau_{\mathrm{GM}} = 1$ s to $\tau_{\mathrm{GM}} = 1024$ s in factors of 2. In other words, we start with a time series that can be regarded as a white Gaussian noise sequence, and we end with a time series that is more or less completely correlated, with correlation time equal to the duration of the time series, T. The standard deviation, $\sigma_{\mathrm{GM}} = 1$ unit, is the same for all time series. We consider the following numerical experiment. For each correlation time, τ_{GM}, we are given a set of N_{MC} Monte Carlo realizations of time series representing simulation results. We use these realizations to compute the simulation bounds. Also, from the number of Monte Carlo trials, N_{MC}, we compute the probability p given in Eq. (2). Next, we generate a set of 1000 first-order Gauss–Markov time series representing results observed by the actual system. The set of 1000 time series have the same σ_{GM} and τ_{GM} as the simulation; in other words, the true hypothesis is the null hypothesis, H_0. For each of the 1000 time series, we compute a so-called rejection index, ρ:

$$\rho \triangleq \frac{N_{\mathrm{out}}}{N} \times 100, \tag{8}$$

where N_{out} is the number of *observed* samples that fall outside of the simulation bounds. It follows that $0 \leq \rho \leq 100$. We note again that the rejection index, ρ, is computed using *all* available samples. We repeat this process for each of the different correlation times, so, for each correlation time, τ_{GM}, we compute 1000 rejection indices.

Results are summarized in Figure 3. The vertical lines indicate the range of values of ρ obtained over the course of 1000 observations. The horizontal dashes above and below the lines indicate the maximum and minimum values of ρ, respectively, obtained over the course of 1000 observations, while the dots represent their averages. Results shown in the top panel of Figure 3 correspond to the case when there are $N_{\mathrm{MC}} = 10$ Monte Carlo realizations of the simulated time series available, while results shown in the bottom panel correspond to the case of $N_{\mathrm{MC}} = 50$.

Values of $p \times 100$ for the two scenarios are represented by the blue horizontal lines. In the case of $N_{\mathrm{MC}} = 10$, $p \times 100 \simeq 18$, while in the case of $N_{\mathrm{MC}} = 50$, $p \times 100 \simeq 4$. We note that, as expected, the average values of ρ match the values of $p \times 100$. Since we have ignored any correlation effects in computing the rejection index, we also note that the *range* of values obtained for ρ over the course of 1000 observations increases with increasing correlation time, τ_{GM}. In the case of $\tau_{\mathrm{GM}} = 1$ s, the underlying time series are effectively uncorrelated, with the correlation time being equal to the sampling interval. Hence, for $\tau_{\mathrm{GM}} = 1$ s, ρ deviates only slightly around the mean value of $p \times 100$. However, as the correlation time increases, this deviation increases. As discussed earlier, this deviation will have no impact on the validation process if correlation effects are taken into account in the computation of the rejection threshold, γ.

Similar to the notion of a rejection index, ρ, defined in Eq. (8), we define a *normalized* rejection threshold, $\tilde{\gamma}$:

$$\tilde{\gamma}(p, N_i) \triangleq \frac{\gamma(p, N_i)}{N_i} \times 100. \tag{9}$$

The rejection threshold, γ is obtained from

$$\alpha = \int_{\gamma}^{\infty} f_{\mathrm{binomial}}(s; p, N_i)\, ds, \tag{10}$$

where f_{binomial} denotes the probability distribution function of a binomial random variable with parameters p and N_i:

$$f_{\mathrm{binomial}}(s; p, N_i) = \binom{N_i}{k} p^k (1-p)^{N_i - k} \delta(s-k), \tag{11}$$

where $\delta(\cdot)$ is the Dirac delta function. The number, N_i, of independent samples can be obtained from an appropriate estimate of the correlation time through Eq. (3). For our numerical experiment, the values of $\tilde{\gamma}$ computed from Eq. (9), corresponding to $\alpha = 0.01$, are shown in red in Figure 3. It is evident that the large deviations in ρ, due mainly to ignoring correlation effects, will have no impact on the simulation validation process as long as the correlation effect is taken into account in the computation of the rejection threshold.

From the results shown in Figure 3, we note that as the correlation time increases, the normalized rejection threshold, $\tilde{\gamma}$, increases in a way similar to the increase observed in the range of values covered by the rejection index, ρ, over the set of 1000 observations. Also, by comparing the results shown in the top and bottom panels of Figure 3, corresponding to $N_{\mathrm{MC}} = 10$ and $N_{\mathrm{MC}} = 50$, respectively, we note that the magnitude of the increase in both $\tilde{\gamma}$ and the range of values covered by ρ *decreases* with (1) increasing number of Monte Carlo realizations of the simulated time series and (2) decreasing correlation time. For example, in the case of the correlation time being equal to one quarter of the duration of the time series, corresponding to the case of $\tau_{\mathrm{GM}} = 256$ s in Figure 3, the value of the normalized threshold, $\tilde{\gamma}$, for $N_{\mathrm{MC}} = 50$ is roughly one third that for $N_{\mathrm{MC}} = 10$. Similarly, we would expect a lower value for $\tilde{\gamma}$ if the duration of the time series were longer.

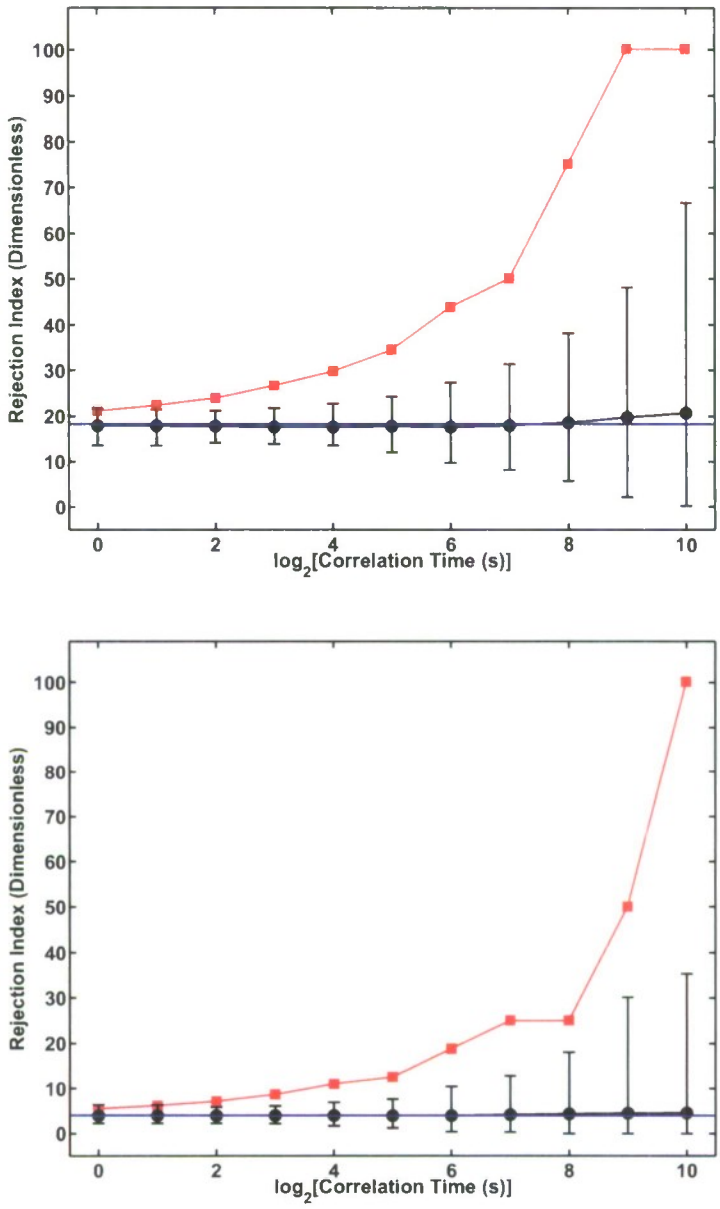

Figure 3. *Variation of the rejection index, ρ, and the normalized rejection threshold, $\tilde{\gamma}$, with correlation time, τ. The top panel shows results corresponding to the case when there are $N_{MC} = 10$ Monte Carlo realizations of the simulated time series, while the bottom panel shows results corresponding to the case of $N_{MC} = 50$. Both the simulated and observed time series are modeled as first-order Gauss–Markov processes with matching parameters; in other words, the true hypothesis is the null hypothesis, H_0, corresponding to the hypothesis that the simulation is consistent with actual system performance. The normalized rejection threshold, $\tilde{\gamma}$, corresponding to $\alpha = 0.01$, is shown in red. The vertical lines indicate the range of values of ρ obtained over the course of 1000 observations. The horizontal dashes above and below the lines indicate the maximum and minimum values of ρ, respectively, obtained over the course of 1000 observations, while the dots represent their average. The blue horizontal lines correspond to $p \times 100$, where p is given by Eq. (2).*

3.1 HOW MANY MONTE CARLO REALIZATIONS?

A question often asked is: "How many Monte Carlo realizations are sufficient to validate a given modeling and simulation product using the proposed method in this report?" The frank answer is: "It depends." In general, it is not possible to promulgate a single number, N_{MC}, of Monte Carlo realizations as a gold standard universally applicable to all simulations. As we saw in the discussion of correlated time series above, the duration, T, and the correlation time, τ, of the time series of the validation metrics play key roles in coming up with acceptable rejection thresholds, γ. As we saw in Figure 3, as the correlation time, τ, of a given time series becomes larger (or equivalently, as the duration, T, of the time series becomes smaller), the normalized rejection threshold, $\tilde{\gamma}$, may become excessively large. In other words, as τ becomes larger (or as T becomes smaller), we require more and more of the observed samples to fall within the simulation bounds. This may be overly conservative, thus reducing the fidelity of the validation process. We need more information—say by observing the time series of a given validation metric for a longer period of time—to make a more accurate assessment. Of course, we do not always possess the luxury of observing a time series as long as we desire. For example, in the case of tracking radars, the tracked target might exit the radar's field of view before sufficient information has been gathered.

By comparing the two plots in Figure 3, we note that as we increase the number, N_{MC}, of Monte Carlo realizations from 10 (top panel) to 50 (bottom panel), the normalized rejection threshold, $\tilde{\gamma}$, becomes smaller. Hence, for short time series (or time series with large correlation times), we may wish to obtain a larger number of Monte Carlo realizations. Of course, there is a limit to this strategy. For example, if the particular geometry of a radar tracking scenario happens to impose a fundamental limit to the amount of information extractable for the purposes of modeling and simulation validation, then the availability of a larger number of Monte Carlo realizations would not necessarily increase the fidelity of the validation process. Under these circumstances, we ought to instead reject the *observation* as a reliable anchoring point for the validation of a given modeling and simulation product.

The relationship between the number, N_{MC}, of Monte Carlo realization, the duration, T, and the correlation time, τ, the rejection threshold, γ, and the model maker's risk, α, is given by Eq. (10), which we write more explicitly as

$$\alpha = \int_{\gamma}^{\infty} f_{\mathrm{binomial}}\left(s; \frac{2}{N_{\mathrm{MC}}+1}, \frac{T}{\tau}\right) ds. \tag{12}$$

For $\alpha = 0.01$, plots of the normalized rejection threshold, $\tilde{\gamma}$, versus the number, $N_i = T/\tau$, independent samples for 10, 25, 50, 75, and 100 Monte Carlo realizations are shown in Figure 4. For $\tilde{\gamma} = 20\%$, plots of the model maker's risk, α, versus the number, N_{MC}, of Monte Carlo realizations for 1, 5, 10, 25, 50, 75, and 100 independent samples are shown in Figure 5. In both figures, the jaggedness in the curves is due to the discrete nature of the problem, which is revealed by the Dirac delta function in Eq. (11). The plots in Figures 4 and 5 reveal the monotonic relations that exist between α, γ, N_i, and N_{MC}. In general, such plots ought to be used to determine the appropriate number of Monte Carlo realizations and to assess the efficacy of a particular observation used as a modeling and simulation validation anchor. For example, as seen in Figure 5, the model maker's risk, α, decreases with increasing number, N_{MC}, of Monte Carlo realizations. The decrease in α

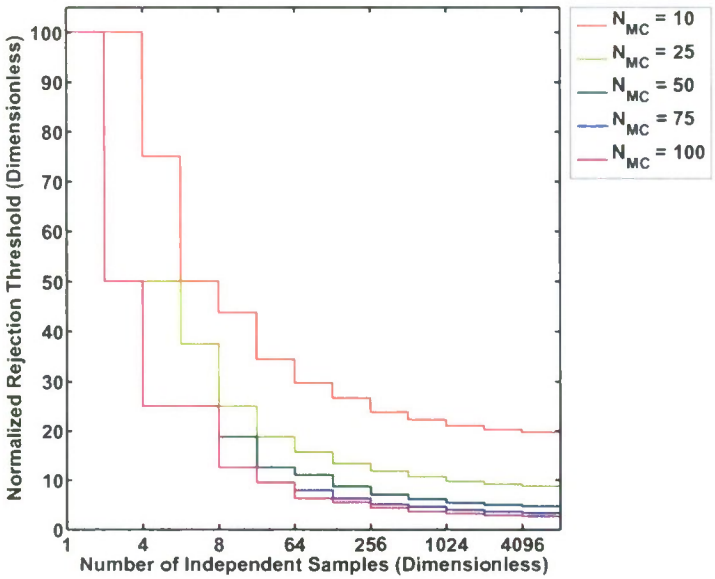

Figure 4. *Variation of the normalized rejection threshold, $\tilde{\gamma}$, with the number of independent samples for 10, 25, 50, 75, and 100 Monte Carlo realizations. For all plots, $\alpha = 0.01$. The jaggedness in the curves is due to the discrete nature of the problem, which is revealed by the Dirac delta function in Eq. (11).*

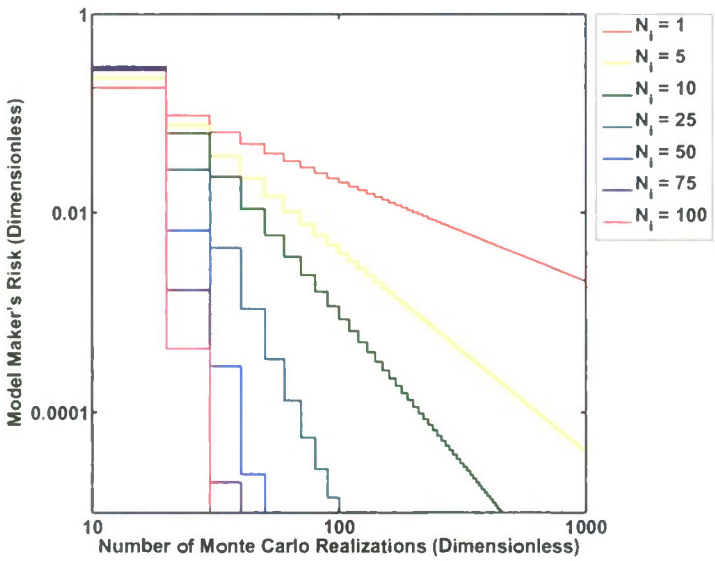

Figure 5. *Variation of the model maker's risk, α, with the number of Monte Carlo realizations for 1, 5, 10, 25, 50, 75, and 100 independent samples. For all plots, $\tilde{\gamma} = 20\%$. The jaggedness in the curves is due to the discrete nature of the problem, which is revealed by the Dirac delta function in Eq. (11).*

becomes more pronounced as the number of independent samples increases. As revealed in both figures, as the number of independent samples becomes smaller, an increase in the number, N_{MC}, of Monte Carlo realizations does not necessarily increase the efficacy of the validation process. Under such circumstances, validation results would be inconclusive.

4. ACCEPTABILITY CRITERIA

In this section, appropriate validation metrics necessary for validating the modeling and simulation of a generic tracking radar are identified. A list of 26 validation metrics is given in Table 1. All validation metrics are functions of time. For convenience, the time dependence of the validation metrics has been suppressed in the notation. We have assumed a phased-array radar, with the measurement space consisting of the range, r, to the target and the two orthogonal direction cosines u and v. The validation metrics can be easily modified to accommodate other types of radar—such as dish radars. The target state estimation problem has been limited to the case when the position and velocity vectors are sufficient to characterize the dynamics of the target. The list in Table 1 can be extended to accommodate for larger dimensional state vectors as needed.

The validation metrics listed in Table 1 can be divided into three broad categories:

1. **Macro tracker output validation metrics:** Items 1 through 6 in Table 1 are sufficient to characterize the general behavior of the tracker output. These validation metrics are particularly useful within a multiple sensor configuration where track data are shared among the sensors. The total position and velocity estimation errors valid at time index k are given by

$$\begin{aligned} \left\|\delta\mathbf{r}_{k|k}\right\| &= \hat{\mathbf{r}}_{k|k} - \mathbf{r}_k \quad \text{and} \\ \left\|\delta\mathbf{v}_{k|k}\right\| &= \hat{\mathbf{v}}_{k|k} - \mathbf{v}_k. \end{aligned} \tag{13}$$

where $\hat{\mathbf{r}}_{k|k}$ and $\hat{\mathbf{v}}_{k|k}$ correspond to the *updated* target position and velocity vector estimates valid at time index k, respectively, while $\mathbf{r}_k$ and $\mathbf{v}_k$ correspond to the true target position and velocity vectors valid at time index k, respectively. The square roots of the traces of the position and velocity quadrants of the error covariance matrix provide estimates of the size of the error hyper-ellipsoid. A characterization of the shape and orientation of the error hyper-ellipsoid, in turn, can be obtained from the so-called normalized estimation error squared (NEES) [1]:

$$\text{NEES}_k = \left[\hat{\mathbf{x}}_{k|k} - \mathbf{x}_k\right]^{\mathrm{T}} P_{k|k}^{-1} \left[\hat{\mathbf{x}}_{k|k} - \mathbf{x}_k\right]. \tag{14}$$

where $\hat{\mathbf{x}}_{k|k}$ is the *updated* target state estimate valid at time index k; $\mathbf{x}_k$ is the true target state valid at time index k; and $P_{k|k}$ is the *updated* state estimation error covariance valid at time index k. The normalized innovation squared (NIS) provides another useful validation metric [1]:

$$\text{NIS}_k = \left[\mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1})\right]^{\mathrm{T}} \left(H_k P_{k|k-1} H_k^{\mathrm{T}} + R_k\right)^{-1} \left[\mathbf{z}_k - \mathbf{h}_k(\hat{\mathbf{x}}_{k|k-1})\right]. \tag{15}$$

where $\mathbf{z}_k$ is the measurement vector valid at time index k; $\hat{\mathbf{x}}_{k|k-1}$ is the *predicted* target state estimate valid at time index k; $\mathbf{h}_k(\cdot)$ is the measurement function valid at time index k; R_k is the measurement error covariance valid at time index k; $P_{k|k-1}$ is the *predicted* state estimation error covariance valid at time index k; and H_k is the sensitivity matrix valid at time index k.

2. **Micro tracker output validation metrics:** Items 7 through 18 in Table 1 provide a more detailed characterization of the behavior of the tracker output. Similar to Eq. (13), the components of the target state estimation error vector, dx_r, dx_u, dx_v, $dx_{\dot{r}}$, $dx_{\dot{u}}$, and $dx_{\dot{v}}$, are defined as the difference between the estimated and true values. In addition to the diagonal elements of the state estimation error covariance matrix, the cross-diagonal elements can also be considered. These are of value particularly when the detailed shape and orientation of the error hyper-ellipsoid are of concern. An inspection of the micro tracker output validation metrics can potentially aid in limiting the possible sources of simulation inconsistency revealed by the macro tracker output validation metrics.

3. **Macro radar front-end output validation metrics:** Items 19 through 26 characterize the behavior of the tracker *input*. Many of the inconsistencies observed in the tracker output metrics can be traced back to the tracker input. Thus, while not strictly necessary for validating the modeling and simulation of a given tracking radar, the macro radar front-end output validation metrics often provide invaluable diagnostics as to the cause of the observed inconsistencies.

TABLE 1: Validation Metrics

ITEM	VALIDATION METRIC	SYMBOL	CODE NAME
1	Total Position Estimation Error	$\|\delta\mathbf{r}\|$	`pe`
2	Total Velocity Estimation Error	$\|\delta\mathbf{v}\|$	`ve`
3	Square Root of the Trace of the Upper-Left 3×3 Quadrant of the State Estimation Error Covariance Matrix Corresponding to the Variance of the Total Position Estimation Error	$\sqrt{\mathrm{tr}\,[P(1:3,1:3)]}$	`tp`
4	Square Root of the Trace of the Lower-Right 3×3 Quadrant of the State Estimation Error Covariance Matrix Corresponding to the Variance of the Total Velocity Estimation Error	$\sqrt{\mathrm{tr}\,[P(4:6,4:6)]}$	`tv`
5	Normalized Estimation Error Squared (NEES)	NEES	`NEES`
6	Normalized Innovation Squared (NIS)	NIS	`NIS`
7	Range Estimation Error	δx_r	`dx1`
8	u Estimation Error	δx_u	`dx2`
9	v Estimation Error	δx_v	`dx3`
10	Range Rate Estimation Error	$\delta x_{\dot{r}}$	`dx4`
11	$\dot{u}$ Estimation Error	$\delta x_{\dot{u}}$	`dx5`

continued on next page

continued from previous page

Item	Validation Metric	Symbol	Code Name
12	$\dot{v}$ Estimation Error	$\delta x_{\dot{v}}$	dx6
13	Standard Deviation of the Range Estimation Error	$\sqrt{P_{rr}}$	sx1
14	Standard Deviation of the u Estimation Error	$\sqrt{P_{uu}}$	sx2
15	Standard Deviation of the v Estimation Error	$\sqrt{P_{vv}}$	sx3
16	Standard Deviation of the Range Rate Estimation Error	$\sqrt{P_{\dot{r}\dot{r}}}$	sx4
17	Standard Deviation of the $\dot{u}$ Estimation Error	$\sqrt{P_{\dot{u}\dot{u}}}$	sx5
18	Standard Deviation of the $\dot{v}$ Estimation Error	$\sqrt{P_{\dot{v}\dot{v}}}$	sx6
19	Range Measurement Error	δz_r	dz1
20	u Measurement Error	δz_u	dz2
21	v Measurement Error	δz_v	dz3
22	Standard Deviation of the Range Measurement Error	σ_r	sz1
23	Standard Deviation of the u Measurement Error	σ_u	sz2
24	Standard Deviation of the v Measurement Error	σ_v	sz3
25	Measured Signal-to-Noise Ratio	SNR	SNR
26	Measured Target Radar Cross-Section	RCS	RCS

Using the validation procedure discussed in Sections 2 and 3, results for a given modeling and simulation product are summarized in a so-called scorecard. The scorecard contains a listing of the rejection indices, ρ, and normalized rejection thresholds, $\tilde{\gamma}$, for the validation metrics listed in Table 1. It is evident that many of the validation metrics in Table 1 are dependent on one another. For example, all validation metrics corresponding to the measurement error and state estimation error covariances are dependent on the signal-to-noise ratio. By presenting the results in the form of a scorecard, correlations among the validation metrics become immediately apparent; thus, the scorecard can additionally serve as a diagnostic tool. By considering rejection thresholds corresponding to different values of the model maker's risk, α, and by noting the cross-correlation between select validation metrics, a validation agent can use a scorecard to declare a given modeling and simulation product as valid or invalid. Using numerical examples, we examine the utility of such scorecards in the next section.

This page intentionally left blank.

5. CASE STUDY

In this section, we devise a controlled numerical experiment to examine the effectiveness of the proposed validation algorithm. We choose a satellite as the target and use a phased-array radar to track the satellite. The measurement space consists of the range, r, to the target and the two orthogonal direction cosines u and v. We model the target radar cross-section (RCS) as an independent and identically log-normal distributed stochastic process with mean μ_{RCS} and variance σ^2_{RCS}. The signal-to-noise ratio (SNR) valid at time index k can be obtained from [12]

$$\text{SNR}_k = \left(\frac{R_0}{r_k}\right)^4 \cdot \frac{\text{RCS}_k}{1\ \text{m}^2}, \tag{16}$$

where R_0 denotes the distance to a perfectly conducting sphere with a cross-sectional area of 1 m^2 at which the SNR is 0 dB. Here, R_0 encompasses contributions from the radar receiver and transmitter functions, along with the relevant losses that appear in the radar range equation [13]. The measurement error variances induced by the receiver thermal noise are functions of SNR and can be expressed as [12]

$$\sigma^2_{i_k} = \frac{A_i^2}{2 \cdot \text{SNR}_k} + B_i^2, \quad i = r, u, v, \tag{17}$$

where A_i and B_i are pre-specified parameters.

In addition to zero-mean white Gaussian receiver thermal noise, we also include the possible effect of *colored noise* caused by unavoidable random effects present in the radar's operational environment, such as atmospheric propagation effects or random platform motion. For each component of the measurement vector (r, u, and v), we model the temporally correlated noise induced by the environment with a first-order Gauss-Markov process; in other words, each component of the measurement vector has a unique pair of standard deviation and correlation time, parametrizing the zero-mean colored Gaussian noise, associated with it. We can include the effect of a "constant" random bias by considering a first-order Gauss–Markov process with a very long correlation time.

We simulate a sequence of radar measurements by adding to the truth data a term accounting for the zero-mean white Gaussian receiver noise and a term accounting for the environmentally-induced zero-mean colored Gaussian noise. Given the sequence of simulated radar measurements, along with their associated measurement error variances, we form a track using a textbook extended Kalman filter [11]. We use the procedure outlined in Section 5.2.2 of [1] for track initialization. We produce results for both the simulation—a "meta-simulation"—and the actual observation. To examine the effectiveness of the proposed simulation validation procedure, we consider the following scenarios:

1. **Perfectly matched scenario:** In the case of a perfectly matched scenario, the simulation and the actual observation results are drawn from the same ensemble. In other words, the true hypothesis is the null hypothesis, H_0, corresponding to the hypothesis that the simulation is consistent with actual system performance. For this scenario, we only include the effect of uncorrelated receiver thermal noise on the measurements, while excluding the effect of any environmentally-induced correlated noise. Results of the simulation validation process are shown in Figures 6 and 7.

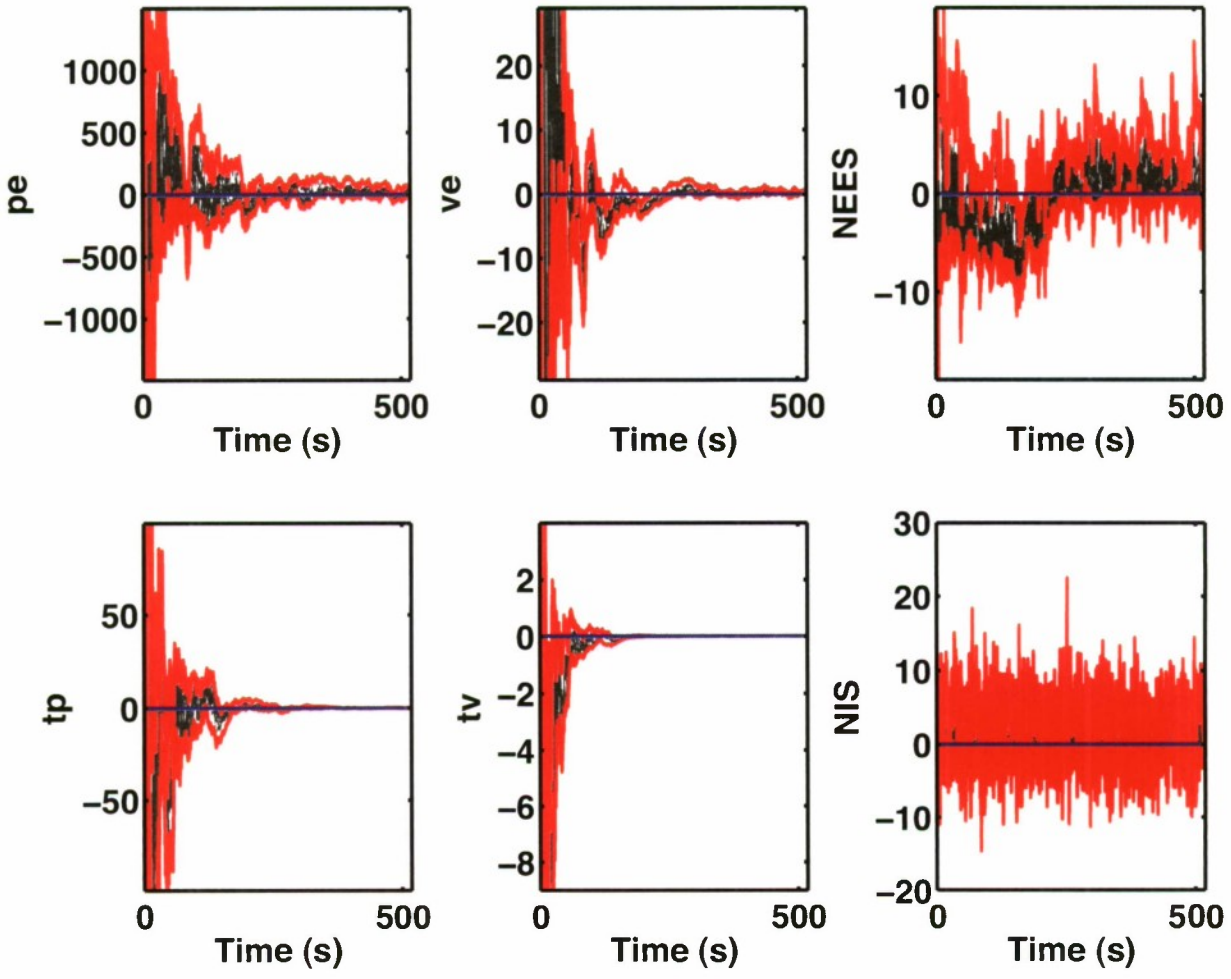

Figure 6. *Time series of the macro tracker output validation metrics for the perfectly matched scenario plotted relative to the observed time series (shown in blue). The 10 Monte Carlo realizations are shown in gray, while the simulation bounds are shown in red. See Table 1 for a definition of symbols. The total position error, "pe," and the square root of the trace of the position quadrant of the error covariance, "tp," are in meters; the total velocity error, "ve," and the square root of the trace of the velocity quadrant of the error covariance, "tv," are in meters per second; and the NEES and NIS are dimensionless. In the case of a perfectly matched scenario, the simulation is deemed to be consistent with actual system performance.*

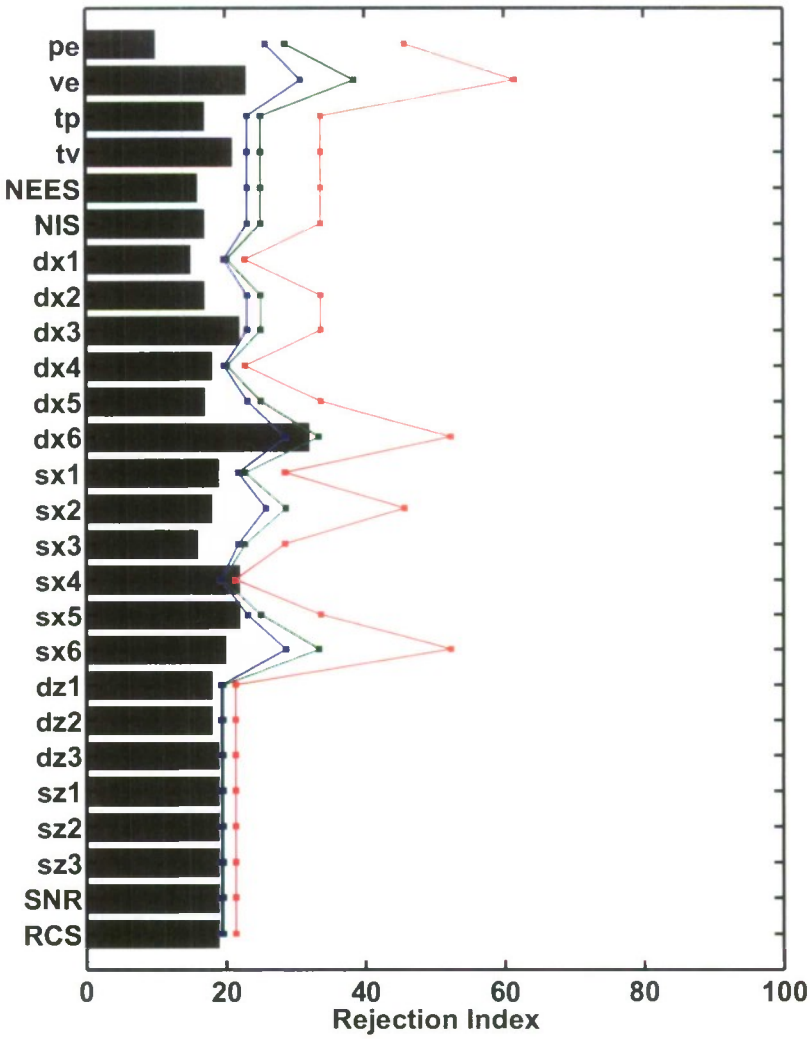

Figure 7. *"Scorecard" for the perfectly matched scenario. The gray bars indicate the rejection indices, ρ, for the 26 validation metrics listed in Table 1. The lines indicate the corresponding normalized rejection thresholds, $\tilde{\gamma}$, for $\alpha = 0.01$ (red), $\alpha = 0.05$ (green), and $\alpha = 0.1$ (blue). See Table 1 for a definition of symbols. In the case of a perfectly matched scenario, the simulation is deemed to be consistent with actual system performance.*

Plots of the time series for the first six validation metrics listed in Table 1 are shown in Figure 6. The time series are plotted relative to the time series of the observed validation metric—shown in blue in Figure 6. When simulation results are consistent with actual system performance, we expect the simulated time series of the validation metrics to cluster symmetrically around the observed time series. As seen in Figure 6, this is indeed the case for the matched scenario.

The scorecard summarizing the consistency of the simulation results with the observation is shown in Figure 7. The gray bars indicate the rejection indices, ρ, corresponding to the 26 validation metrics listed in Table 1. The lines indicate the normalized rejection thresholds, $\tilde{\gamma}$, corresponding to $\alpha = 0.01$ (red), $\alpha = 0.05$ (green), and $\alpha = 0.1$ (blue). The variation of the normalized rejection ratios from validation metric to validation metric is due to the fact that the time series of each validation metric has a unique correlation timescale; hence, the number of independent samples is not necessarily the same for all the validation metrics, even though the total number of samples might be the same. For $\alpha = 0.01$, the rejection indices for *all* validation metrics remain below the normalized rejection thresholds. Therefore, for $\alpha = 0.01$, the validation agent may safely declare the simulation results to be consistent with the observation. For larger values of α, the validation metrics $dx_{\dot{v}}$ and $P_{\dot{r}\dot{r}}$ ("`dx6`" and "`sx4`" in Figure 7, respectively) fall above the normalized rejection thresholds, albeit not too far above. Acceptance or rejection of the simulation based on these two metrics will depend on the validation agent's common sense and judgement. For example, the validation agent may have reason to believe that numerical errors might have caused the rejection indices of these validation metrics to have fallen below the normalized rejection thresholds corresponding to the larger values of α and thus pass the simulation. For this scenario, we would declare the simulation as consistent with actual system performance, despite the small transgression of the $dx_{\dot{v}}$ and $P_{\dot{r}\dot{r}}$ validation metrics for large values of the model maker's risk, α.

2. **Mismatched target scenario:** In the case of a mismatched target scenario, the simulation and the observation results are statistically matched except for the target model. In other words, the true hypothesis is the alternative hypothesis, H_1, corresponding to the hypothesis that the simulation is inconsistent with the actual system performance. To illustrate, we consider the case when the mean value of the observed target RCS is a factor of 2 (3 dB) larger than the simulated value. Plots of the SNR and RCS versus time are shown in Figure 8. As seen in Figure 8 and as is evident from Eq. (16), the mean value of the observed target SNR is also a factor of 2 larger than the simulated value. For this scenario, we also include only the effect of uncorrelated receiver thermal noise on the measurements, while excluding the effect of any environmentally induced correlated noise. Results of the simulation validation process are shown in Figures 9 and 10.

   Plots of the time series for the first six validation metrics listed in Table 1 are shown in Figure 9. The simulated absolute position and velocity estimation errors ("`pe`" and "`ve`," respectively) appear to be consistent with the observation, while the simulated square roots of the traces of the position and velocity quadrants of the error covariance matrix ("`tp`" and "`tv`," respectively) are inconsistent. The NEES and NIS also appear to be consistent. The inconsistency in the error covariance is attributable to the mismatch in the simulated and observed target SNR. Since the simulated SNR is on average a factor of 2 smaller than the

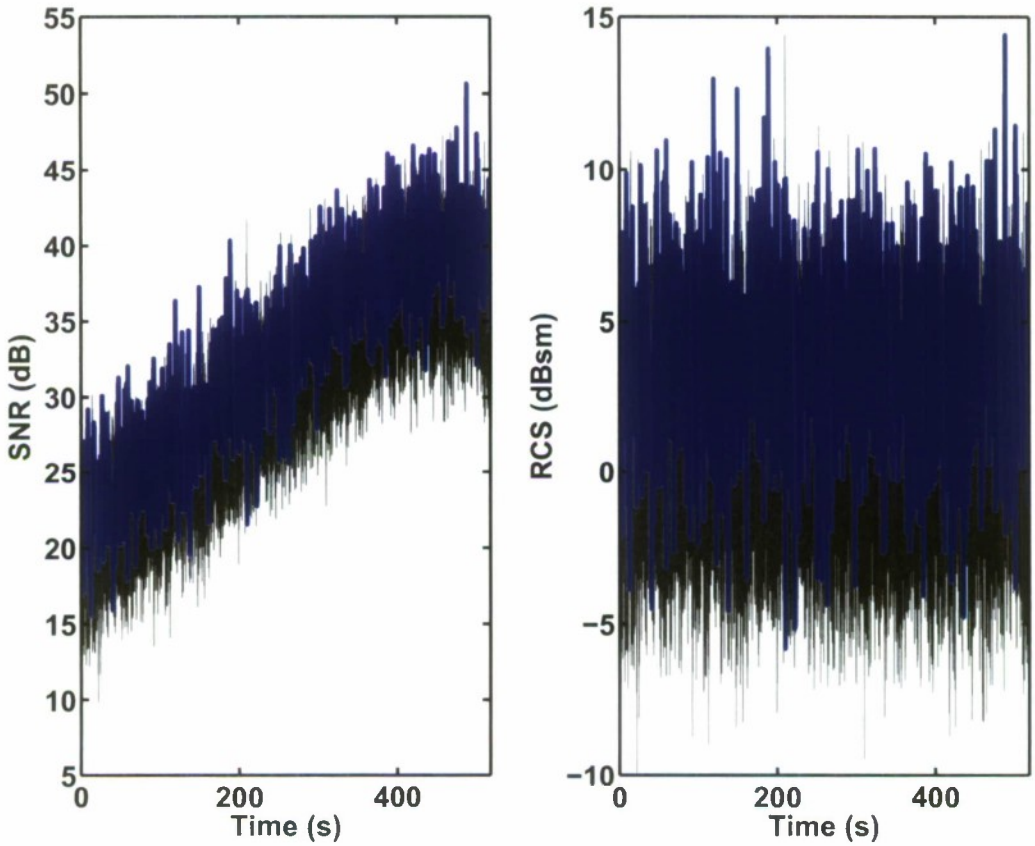

Figure 8. *Mismatched target SNR and RCS. The 10 Monte Carlo realizations are shown in gray, while the observed time series are shown in blue.*

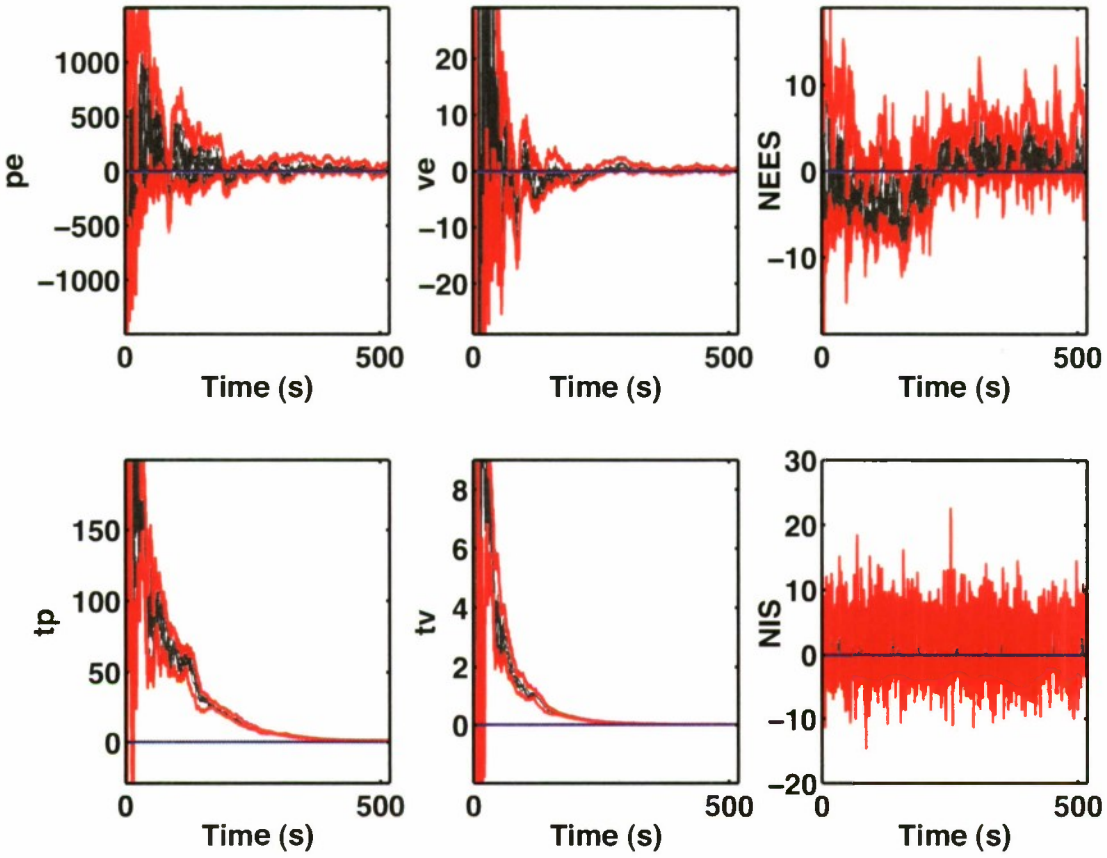

Figure 9. *Time series of the macro tracker output validation metrics for the mismatched target scenario plotted relative to the observed time series (shown in blue). The 10 Monte Carlo realizations are shown in gray, while the simulation bounds are shown in red. See Table 1 for a definition of symbols. The total position error, "*`pe`*," and the square root of the trace of the position quadrant of the error covariance, "*`tp`*," are in meters; the total velocity error, "*`ve`*," and the square root of the trace of the velocity quadrant of the error covariance, "*`tv`*," are in meters per second; and the NEES and NIS are dimensionless. In the case of a mismatched target scenario, the simulation is deemed to be inconsistent with actual system performance.*

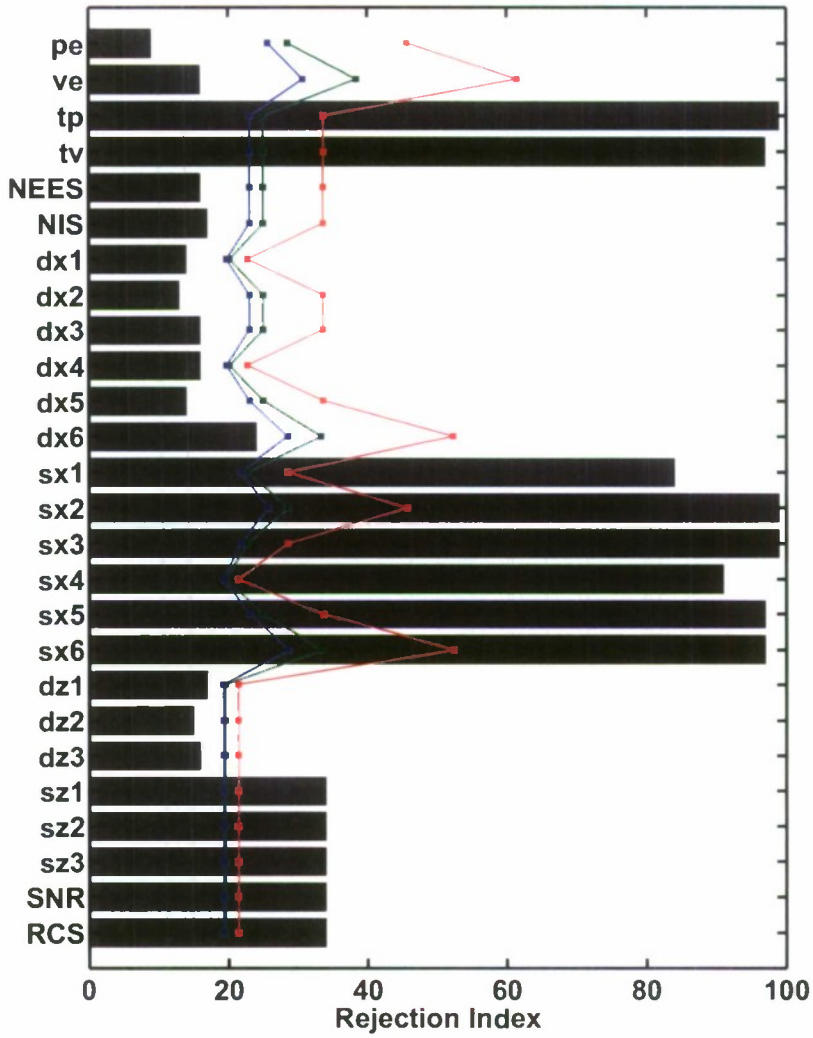

Figure 10. *"Scorecard" for the mismatched target scenario. The gray bars indicate the rejection indices, ρ, for the 26 validation metrics listed in Table 1. The lines indicate the corresponding normalized rejection thresholds, $\tilde{\gamma}$, for $\alpha = 0.01$ (red), $\alpha = 0.05$ (green), and $\alpha = 0.1$ (blue). See Table 1 for a definition of symbols. In the case of a mismatched target scenario, the simulation is deemed to be inconsistent with actual system performance.*

observed SNR, the simulated values of the error covariance are more "pessimistic." This explains why the simulated square roots of the traces of the position and velocity quadrants of the error covariance matrix are *above* the observation values. As seen from Figure 9, even though the simulated covariance matrix is inconsistent with the observed value, the covariance mismatch was not sufficient to cause an inconsistency in the NEES and NIS—at least not for this case.

The scorecard summarizing the consistency of the simulation results with the observation is shown in Figure 10. Again, we notice that the simulation fails for all the validation metrics that are dependent on SNR—specifically, the validation metrics corresponding to the measurement error and the state estimation error covariances. We also note that since the modeling and simulation remain exactly the same as in the perfectly matched scenario, the normalized rejection thresholds, $\tilde{\gamma}$, which are computed based on the simulation results, also remain the same.

3. **Mismatched environment scenario:** In the case of a mismatched environment scenario, the simulation and observation results are statistically matched except for the environmental impact. In other words, the true hypothesis is the alternative hypothesis, H_1. To illustrate, we consider the case when the observed time series of the v component of the measurement vector is corrupted by colored noise. The colored noise is modeled with a first-order Gauss–Markov process with a standard deviation of 1 msin and a correlation time of 130 s. Plots of the measurement errors for this scenario are shown in Figure 11. Results of the simulation validation process are shown in Figures 12 and 13.

   Plots of the time series for the first six validation metrics listed in Table 1 are shown in Figure 12. As seen in the figure, for this scenario, the simulated total position and velocity estimation errors ("`pe`" and "`ve`," respectively) appear to be inconsistent with the observation, while the square roots of the traces of the position and velocity quadrants of the error covariance matrix ("`tp`" and "`tv`," respectively) are consistent. The inconsistency in the state estimation error is obviously caused by the bias in the v component of the measurement error, which is not accounted for by the simulation. Since both the measurement and the state estimation error covariances depend mainly on SNR, they are not affected by the time-varying bias shown in Figure 11. However, for more severe biases, the state estimation error covariance can also be significantly impacted. This is due to the fact that for nonlinear problems (such as tracking a ballistic target), the Jacobians in the expressions for the error covariance in the prediction and update steps of the extended Kalman filter depend on the target state estimate [11], which, in turn, is directly impacted by measurement biases in the update step of the Kalman filter. Since the simulated state estimation errors are inconsistent with the observation, NEES is also seen to be inconsistent in Figure 12.

   The scorecard summarizing the consistency of the simulation results with the observation is shown in Figure 13. Again, we notice that the simulation fails for all validation metrics that are impacted by the v measurement error bias—specifically, the validation metrics corresponding to the state estimation error. Also, we again note that since the modeling and simulation remain exactly the same as in the previous two scenarios, the normalized rejection thresholds, $\tilde{\gamma}$, which are computed based on the simulation results, also remain the same.

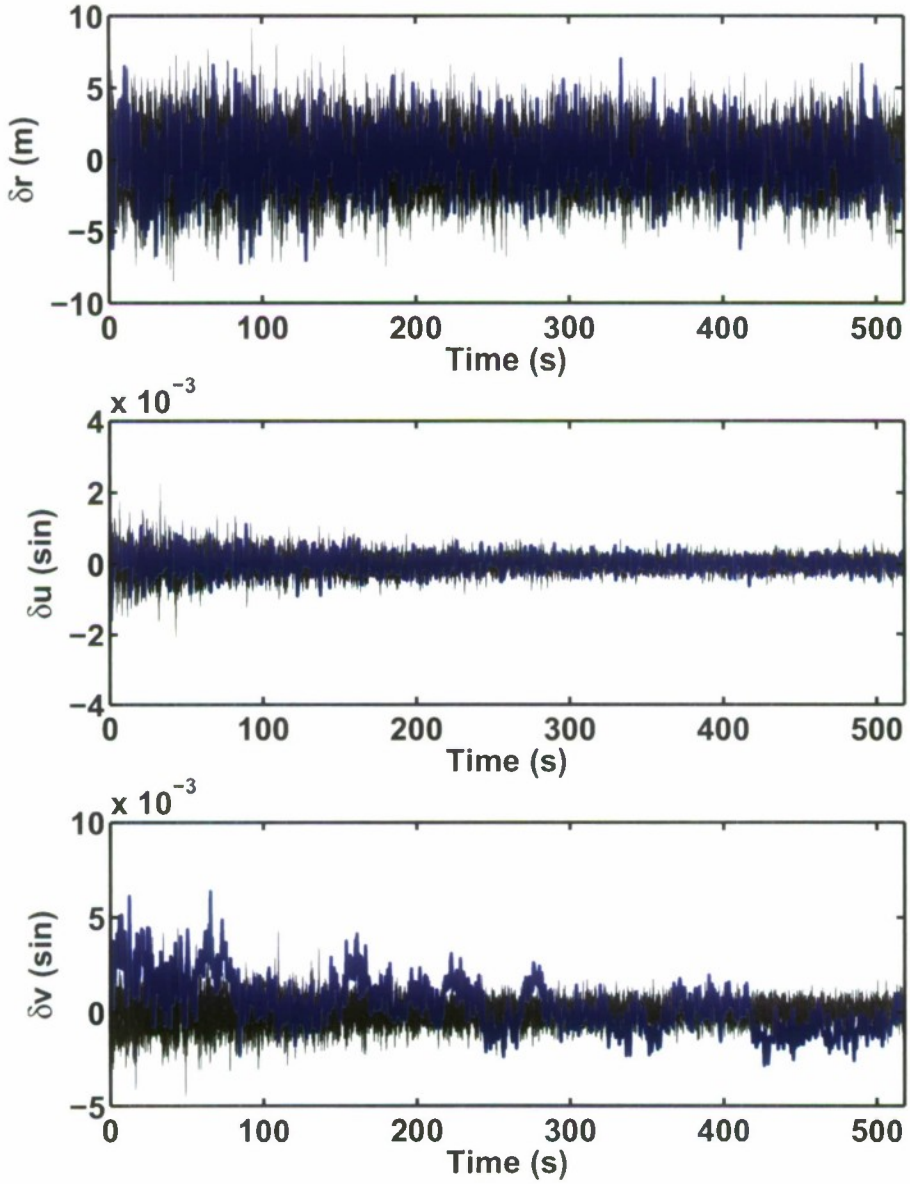

Figure 11. *Mismatched v measurement error. The 10 Monte Carlo realizations are shown in gray, while the observed time series are shown in blue.*

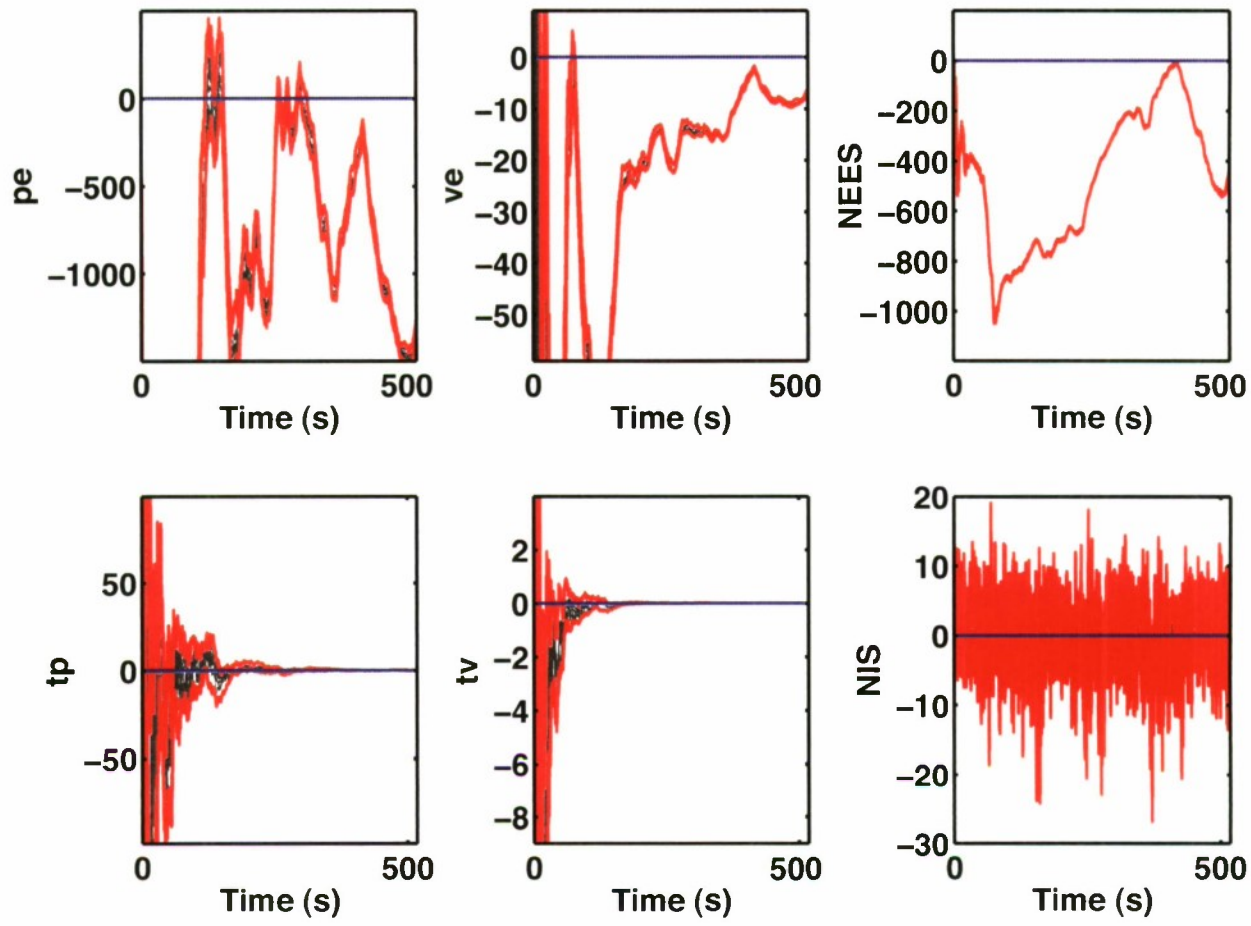

Figure 12. *Time series of the macro tracker output validation metrics for the mismatched environment scenario plotted relative to the observed time series (shown in blue). The 10 Monte Carlo realizations are shown in gray, while the simulation bounds are shown in red. See Table 1 for a definition of symbols. The total position error, "pe," and the square root of the trace of the position quadrant of the error covariance, "tp," are in meters; the total velocity error, "ve," and the square root of the trace of the velocity quadrant of the error covariance, "tv," are in meters per second; and the NEES and NIS are dimensionless. In the case of a mismatched environment scenario, the simulation is deemed to be inconsistent with actual system performance.*

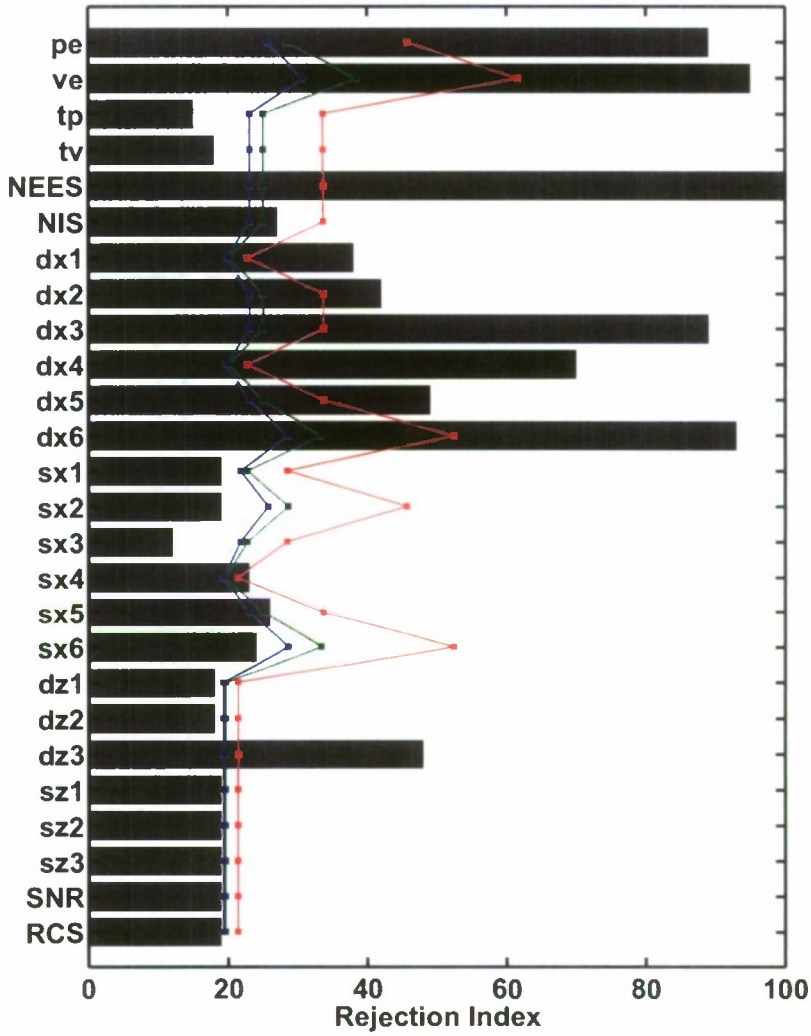

Figure 13. "Scorecard" for the mismatched environment scenario. The gray bars indicate the rejection indices, ρ, for the 26 validation metrics listed in Table 1. The lines indicate the corresponding normalized rejection thresholds, $\tilde{\gamma}$, for $\alpha = 0.01$ (red), $\alpha = 0.05$ (green), and $\alpha = 0.1$ (blue). See Table 1 for a definition of symbols. In the case of a mismatched environment scenario, the simulation is deemed to be inconsistent with actual system performance.

This page intentionally left blank.

6. SUMMARY

The procedure proposed in this report for validating the modeling and simulation of a generic tracking radar is based on a statistical hypothesis test. The two hypotheses are (1) the hypothesis that the simulation is *consistent* with actual system performance—the null hypothesis, H_0—and (2) the hypothesis that the simulation is *inconsistent* with actual system performance—the alternative hypothesis, H_1. The procedure is cognizant of the model maker's risk, α, and the the model user's risk, β, which correspond to the probabilities of Type I and Type II errors, respectively.

The proposed acceptability criteria are anchored to *single* discrete-event observations. Since, in general, the observed behavior is not repeatable, the probability density function necessary for the computation of the model user's risk, β, is not accessible. However, it is always possible to derive any desirable statistics for the *simulation* results through multiple Monte Carlo realizations; thus, it is always possible to derive appropriate rejection thresholds, γ, based on pre-specified values of the model maker's risk, α. The model maker's risk is used as an adjustable parameter for the validation procedure providing different rejection thresholds. Even though a "receiver operating characteristic" or "ROC" curve, providing the trade-off between the model maker's risk, α, and the model user's risk, β, cannot be computed explicitly (due to the unavailability of the probability density function of the observed behavior), it is nevertheless understood that smaller values of the model maker's risk, α, would give rise to larger values of the model user's risk, β. Thus, a *family* of rejection thresholds corresponding to different values of the model maker's risk, α, ought to be considered in an effort to minimize the model user's risk, β.

The modeling and simulation validation procedure proposed in this report is performed independently on a set of validation metrics. A list of 26 validation metrics sufficient for validating the modeling and simulation of a phased-array radar tracking a ballistic target is given in Table 1. The list can be readily expanded to account for models of tracking nonballistic targets with other types of sensors. Conversely, for many applications, not all of the validation metrics listed in Table 1 need be considered. For example, in a multiple sensor configuration, where only the total position and velocity errors in shared track data are of concern, it may be sufficient to examine only the first six items listed in Table 1.

Many of the validation metrics listed in Table 1 are not statistically independent. For example, the validation metrics corresponding to the measurement error and state estimation error covariances are dependent on the signal-to-noise ratio. The correlation among select validation metrics can serve as a diagnostic tool helping to identify the root cause of the failure of a given modeling and simulation product. The numerical experiment conducted in Section 5 demonstrates the utility of the correlation among select validation metrics. All validation metrics listed in Table 1 come in the form of time series; hence, any temporal correlation present in the time series must also be taken into account.

The steps taken in the proposed validation procedure are summarized as follows. For each validation metric, we count the number of samples of the *observed* time series that fall outside of bounds prescribed by N_{MC} Monte Carlo realizations of the simulated time series. The bounds at each time index correspond to the minimum and maximum values of the Monte Carlo realizations.

Subsequently, if the number of observed samples that are outside of the simulation bounds are *above* a pre-computed rejection threshold, we declare the simulated time series of the particular validation metric under scrutiny as *inconsistent* with the observed time series. For each validation metric, the rejection threshold, γ, is computed using Eq. (10). The rejection threshold depends on (1) the model maker's risk, α, (2) the number, N_i, of independent samples in the simulated time series, and (3) the number, N_{MC}, of Monte Carlo realizations—through the use of Eq. (2). The interrelationship between γ, N_i, and N_{MC} for a given α is explored in Figure 4.

The number, N_i, of independent samples can be obtained using any of the techniques discussed in Section 3. In order to model the outcome of the aforementioned counting process as a binomial random variable, the samples must be statistically independent. Statistical independence can be ensured by devising a counting procedure that repeatedly divides the time series into uncorrelated segments and then picks independent samples from each segment. We argued that the statistical dependence that would exist in counting *all* samples can be accounted for in the computation of the rejection threshold. Based on a Monte Carlo analysis, we showed in Section 3 that rejection indices, ρ, computed based on counting all samples, rarely exceed the compensated normalized rejection thresholds, $\tilde{\gamma}$. We have thus opted for the simpler approach of accounting for the statistical dependence of the time series in the computation of the rejection threshold.

The last step of the proposed validation procedure consists of summarizing the results of the above computations for each of the validation metrics listed in Table 1 in a scorecard. The scorecard reveals any cross-correlation that exists among select validation metrics—thus serving as a diagnostic tool. For each discrete-event observation, the scorecard contains a list of rejection indices for the different validation metrics, with each rejection index—expressed as a number between 0 and 100—denoting the ratio of the samples of the observed time series of the associated validation metric that are outside of the simulation bounds. Normalized rejection thresholds for the different validation metrics—also expressed as numbers between 0 and 100—are also included in the scorecard. Furthermore, we require a *family* of normalized rejection thresholds, corresponding to different values of the model maker's risk, α, be included in the scorecard. Examples of scorecards are presented in Section 5. Specifically, Figures 7, 10, and 13 illustrate useful graphical representations of scorecards obtained for the different scenarios we studied in Section 5. Use of such graphical displays of the simulation results is encouraged as they readily reveal the cross-correlation among select validation metrics. Thus, using sound judgement and common sense, a validation agent may use such scorecards—obtained for various discrete-event observations—to accept or reject a given modeling and simulation product. More importantly, the scorecards also serve as a first step toward identifying problems in a given product and thus pave the road to modeling and simulation *improvement*.

The modeling and simulation acceptability criteria proposed in this report focused mainly on validating the modeling and simulation of the tracking capability of a generic radar. However, these criteria can be generalized to include other radar functions—or other types of sensors, such as optical or IR sensors. Furthermore, the proposed approach can be readily extended to validating the modeling and simulation of the *targets* themselves. Also, the proposed approach can be extended to validating the modeling and simulation of the *operational environment* of a given sensor, which directly impacts the performance of all sensor functions. All such problems involve the modeling

and simulation of time series, and we believe the techniques proposed in this report are well suited to the validation of the modeling and simulation of any simulated time series based on single discrete-event observations.

This page intentionally left blank.

REFERENCES

[1] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Wiley & Sons, 2001.

[2] D. K. Pace, "Modeling and simulation verification and validation challenges," *Johns Hopkins APL Technical Digest*, vol. 25, no. 2, 2004.

[3] DoD Instruction 5000.61, "DoD modeling and simulation (M&S) verification, validation, and accreditation (VV&A)," May 2003.

[4] O. Balci, "Principles and techniques of simulation validation, verification, and testing," in *Proceedings of the 1995 Winter Simulation Conference*, C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldman, Eds., 1995.

[5] R. G. Sargent, "Validation and verification of simulation models," in *Proceedings of the 2004 Winter Simulation Conference*, R. G. Ingalls, M. D. Rosetti, J. S. Smith, and B. A. Peters, Eds., 2004.

[6] A. Roy, R. K. Siddani, and P. K. Kundu, "Testing equality of spectral densities across spatial locations for rain gauge data," *Submitted to the Annals of Applied Statistics*, December 2008.

[7] S. M. Kay, *Fundamentals of Statistical Signal Processing. Volume II: Detection Theory*. Prentice Hall PTR, 1998.

[8] K. Rohlfs and T. L. Wilson, *Tools of Radio Astronomy*, 2nd ed. Berlin: Springer-Verlag, 1996.

[9] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in C++: The Art of Scientific Computing*, 2nd ed. Cambridge, UK: Cambridge University Press, 2002.

[10] N. R. Lomb, "Least-squares frequency analysis of unequally spaced data," *Astrophysics and Space Science*, vol. 39, pp. 447–462, 1976.

[11] A. Gelb, *Applied Optimal Estimation*. Cambridge, Massachusetts: The MIT Press, 1974.

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

[13] L. V. Blake, *Radar Handbook*, 2nd ed. Boston: McGraw-Hill, 1990, ch. 2. Prediction of Radar Range.

This page intentionally left blank.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

1. REPORT DATE
28 July 2009

2. REPORT TYPE
Technical Report

3. DATES COVERED *(From - To)*

4. TITLE AND SUBTITLE

Validating the Modeling and Simulation of a Generic Tracking Radar

5a. CONTRACT NUMBER
FA8721-05-C-0002

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

H.C. Lambert, S.R. Vogl, A.S. Brewster, and K-P. Dunn

5d. PROJECT NUMBER
1282

5e. TASK NUMBER
1

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) AND ADDRESS(ES)

MIT Lincoln Laboratory
244 Wood Street
Lexington, MA 02420-9108

8. PERFORMING ORGANIZATION REPORT NUMBER
TR-1134

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Missile Defense Agency/SN
7100 Defense Pentagon
Washington, DC 20301-7100

10. SPONSOR/MONITOR'S ACRONYM(S)

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
ESC-TR-2007-066

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report proposes acceptability criteria for validating the modeling and simulation of a generic tracking radar. The approach is based on a statistical hypothesis test that strives to minimize risk to both the model user and the model maker. The validation process is limited to the comparison of a set of Monte Carlo realizations of judiciously selected validation metrics with *single* discrete-event observations made by the actual sensor. The effectiveness of the criteria is examined with controlled numerical experiments whereby the impact of poor models for target signature and signal propagation effects on the simulation of the sensor's tracking function is explored. Results are summarized in a scorecard containing a list of rejection indices and rejection thresholds for the different validation metrics. The rejection thresholds take into account the effect of any statistical correlations present in individual validation metrics. Due to the unavailability of the probability density function of the observed behavior, which prevents the computation of the model user's risk directly, a family of normalized rejection thresholds, corresponding to different values of the model maker's risk, are included. Scorecards also reveal any cross-correlations that exist among select validation metrics. This feature of the scorecard can serve as a diagnostic tool—thus, aiding in modeling and simulation improvement.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE	Same as report	48	19b. TELEPHONE NUMBER *(include area code)*
Unclassified	Unclassified	Unclassified			

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39.18