JOHNS HOPKINS APL

TECHNICAL DIGEST

December 2025, Volume 38, Number 1
APL Research and Development

JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

The *Johns Hopkins APL Technical Digest* is an unclassified technical journal published by the Johns Hopkins Applied Physics Laboratory. Its purpose is to communicate recent Laboratory advances in science, technology, engineering, and mathematics, along with expository articles by APL staff members that accelerate education and understanding of new capabilities, results, and discoveries.

The *Johns Hopkins APL Technical Digest* (ISSN 1930-0530) is published under the auspices of the Johns Hopkins University Applied Physics Laboratory (APL), 11100 Johns Hopkins Road, Laurel, MD 20723-6099.

Requests to be added to the email distribution list announcing each issue should be submitted to TechnicalDigest@jhuapl.edu. Permission to reprint text and/or figures should be requested at http://bit.ly/apl-3sxU2bS.

The following abstracting services currently cover the *Johns Hopkins APL Technical Digest*: Chemical Abstracts; Current Contents; Science Citation Index; Engineering Village; and the following CSA abstracts: Aerospace and High Technology Database; Aquatic Sciences and Fisheries Abstracts; Computer and Information Systems Abstracts; Electronics and Communications Abstracts; Mechanical and Transportation Engineering Abstracts; Meteorological and Geoastrophysical Abstracts; and Oceanic Abstracts.

Distribution Statement A: Approved for public release; distribution is unlimited.

The *Johns Hopkins APL Technical Digest* is published electronically at https://www.jhuapl.edu/technical-digest.

JOHNS HOPKINS APL
TECHNICAL DIGEST

Volume 38, Number 1
APL Research and Development

The *Johns Hopkins APL Technical Digest* can be accessed electronically at **www.jhuapl.edu/technical-digest**.

Bayesian Statistics: An Introduction for the Practicing Reliability Engineer

Carsten H. Botts

ABSTRACT

This article introduces and reviews some of the principles and methods used in Bayesian reliability. It specifically discusses methods used in the analysis of success/no-success data and describes a simple Monte Carlo algorithm that can be used to calculate the posterior distribution of a system's reliability. This algorithm is especially useful when a system's reliability is modeled through the reliability of its subcomponents, yet only system-level data are available.

INTRODUCTION

A common way to measure the reliability of a system is to determine the probability that it will pass, or survive, a stress test. This typically requires several system tests, but it may be prohibitively expensive to conduct many tests of a sophisticated system, such as an air-to-air missile[1] or medical device.[2]

Bayesian statistical methods can help in such a situation, since they enable inclusion of other types of data (such as computer simulation experiments or subject-matter-expert opinions). Bayesian methods may also be necessary because many modern systems do not fail during testing. With no failures, it is difficult for classical statistics to accurately quantify the probability of failure.

This article begins by briefly reviewing Bayesian methods. It then discusses how these methods can be used to learn more about the probability of a system surviving a test, and it concludes by describing a straightforward algorithm for calculating a total system's reliability once it has been tested. This algorithm is simple, produces an exact answer, and is not mentioned in the Bayesian reliability literature. The efficiency of this algorithm is illustrated with an example.

BAYESIAN METHODS

This section introduces the basics of Bayesian principles and Bayesian statistical methodology. The most effective way to introduce this concept is to contrast it with the principles and methodology of classical statistics. The biggest difference between Bayesian and classical statistics is in how probability is defined. In classical statistics, probability is the long-run frequency of an event. So for a fixed (and unknown) parameter such as a population mean, μ,

$$\mathbb{P}(3.66 \leq \mu \leq 4.11) = \begin{cases} 1 & \text{if true} \\ 0 & \text{if not} \end{cases}. \qquad (1)$$

In words, Eq. 1 states that the fixed parameter μ is either in the stated interval or it is not.

Bayesian statisticians think about probability in a different way. In Bayesian statistics, probability is the belief that a statement is true. So if one believes (based on their experience and/or the data that they have seen) that μ is within the stated interval with 95% probability, it would be fair to say that

$$\mathbb{P}(3.66 \leq \mu \leq 4.11) = .95. \quad (2)$$

The objective and point of Bayesian statistics is to calculate probabilities like the one in Eq. 2 and to ensure that this calculation is scientifically respected.

To calculate such a probability, a Bayesian statistician begins with a prior distribution. Assuming that the unknown parameter of interest is θ, this prior distribution is typically denoted as $\pi(\theta)$. The prior distribution indicates where the user believes the parameter θ to be before data are observed or collected. Assume, for example, that we purchased a coin at a magic shop. Upon the purchase, the shop owner tells us that the coin will more often turn up heads than tails. In this case, we will let $\theta = \mathbb{P}(H)$ and define the prior distribution $\pi(\theta)$ for all values of θ between 0 and 1. This prior will also be more heavily weighted toward values of 1 to indicate that, a priori, the coin is expected to turn up heads more frequently than tails.

Once the prior is formulated, data are collected. The distribution of the data conditioned on a value of θ is written as $p(\mathbf{x}|\theta)$, i.e.,

$$(X_1, X_2, \ldots, X_n) \sim p(x_1, x_2, x_3, \ldots, x_n|\theta) = p(\mathbf{x}|\theta),$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_n)$. The function $p(\mathbf{x}|\theta)$ is also referred to as the likelihood of θ.

With the prior and the likelihood, the posterior distribution, typically denoted as $\pi(\theta|\mathbf{x})$, can be calculated. The posterior is calculated using Bayes's rule.[3] This calculation is shown below:

$$\pi(\theta|\mathbf{x}) = \frac{p(\mathbf{x}, \theta)}{p(\mathbf{x})} = \frac{p(\mathbf{x}|\theta)\pi(\theta)}{\int_\Theta p(\mathbf{x}|\theta)\pi(\theta)d\theta} \propto p(\mathbf{x}|\theta)\pi(\theta), \quad (3)$$

where Θ is the set of all possible values of θ. The formula given in Eq. 3 makes sense: the posterior is proportional to the prior distribution of θ (where we thought θ was before collecting data) times the likelihood (where the data suggests θ to be).

This article focuses on how to use Bayesian methods to learn more about the probability that a system survives a test. The subsection titled The Prior and Posterior of One Subsystem specifically discusses how Bayesian methods are used to learn about the survival probability of one system, and the subsection following that one, The Prior and Posterior of the Entire System, discusses the methods necessary to learn about the survival probability of one system composed of multiple subsystems. The algorithm discussed in the latter subsection is elementary and uses no approximations when calculating its answer.

BAYESIAN METHODS FOR BERNOULLI EXPERIMENTS

The Prior and Posterior of One Subsystem

Let us first assume that we are interested in learning about the probability that a system passes an endurance test of some sort. We will denote this probability as θ, and we will conduct n trials/tests on this system and record X, the number of times (out of the n trials) that the system passes a test.

To do a Bayesian analysis on θ, we begin by specifying a prior distribution for it. The beta distribution is often used as the prior for the probability of success in a sequence of success/failure trials.[4,5] The beta distribution is specified by two parameters and is especially convenient in cases such as this since it is a conjugate prior (i.e., it produces a posterior distribution of the same form). The beta prior takes the form

$$\pi(\theta) = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1} \qquad 0 \leq \theta \leq 1,$$

where $\Gamma(\cdot)$ is the gamma function. This prior has mean

$$\text{Prior Mean} = \frac{\alpha}{\alpha+\beta}$$

and variance

$$\text{Prior Var} = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}.$$

The values of α and β ($\alpha, \beta > 0$) are selected to reflect the user's prior belief. This prior belief is often informed in a variety of ways, such as expert opinion, computer simulation, or prior experiments. A user who believes that θ is small (<0.5) would set $\alpha < \beta$ (making the prior mean < 0.5). A user who believes that θ is large (>0.5) would set $\alpha > \beta$. The confidence in these prior beliefs is, of course, reflected in the variance of the prior. If a user wanted to set the prior mean of θ to 0.4, they could set $\alpha = 2$ and $\beta = 3$, making the prior variance 0.04. If a user wanted to elevate the confidence in this statement (that the prior mean of θ is 0.4), they could adjust the values of α and β to $\alpha = 20$ and $\beta = 30$, making the prior variance 0.004. And if a user knew absolutely nothing about θ, they would set $\alpha = \beta = 1$, in which case the prior for θ is uniform over the interval (0, 1).

There is no consistent selection of α and β in reliability studies. Leoni et al.[6] set α = 3 and β = 1 in one of their reliability studies, Burke and Harman[7] set α = 7.2 and β = 0.8 in one of their reliability studies, and Martz et al.[8] set α = 27.3 and β = 0.5. When priors are informed by simulation results or previous experiments, analysts sometimes set

$$\alpha = n_{\text{pr}} \cdot \hat{\theta}^{\text{pr}} + 1 \quad \text{and} \quad \beta = n_{\text{pr}} \cdot \left(1 - \hat{\theta}^{\text{pr}}\right) + 1,$$

where $\hat{\theta}^{\text{pr}}$ is a prior estimate of θ, and n_{pr} is some positive number that represents the confidence the analyst has in the simulation or experiment informing the prior;[9] think of n_{pr} as the effective sample size that informs the prior. The greater this effective sample size that informs the prior, the more peaked the prior distribution is near the prior estimate of θ. If no confidence exists in the simulation informing the prior, then $n_{\text{pr}} = 0$ and the prior would be flat.

Figures 1 and 2 illustrate what these prior distributions look like. The prior in Figure 1 puts large probability on low values of θ and does so by setting the value of α to be significantly less than the value of β. The prior in Figure 2 puts large probability on high values of θ and does so by setting the value of β to be smaller than the value of α. Also observe that the prior is much more peaked for small values of θ in Figure 1 than it is for large values of θ in Figure 2. This is a consequence of the difference in the values between α and β. The difference is larger for the prior in Figure 1 than it is for the prior in Figure 2.

Let us now assume that we observe x successes of the system out of n tests conducted. In this case, the likelihood is binomial,

$$p(x|\theta) = \binom{n}{x} \theta^x (1-\theta)^{n-x},$$

making the posterior distribution of θ

$$\pi(\theta|x) = \frac{p(x|\theta)\,\pi(\theta)}{\int_\Theta p(x|\theta)\pi(\theta)\,d\theta} = \frac{\binom{n}{x}\theta^x(1-\theta)^{n-x}\,\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}}{\int_0^1 \left[\binom{n}{x}\theta^x(1-\theta)^{n-x}\,\frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}\theta^{\alpha-1}(1-\theta)^{\beta-1}\right] d\theta}. \tag{4}$$

There is a simple way to calculate the posterior distribution in Eq. 4 without having to evaluate the integral in the denominator. First observe that the expression in the denominator is not a function of θ; it is a normalizing constant independent of θ, and for this reason we can write

$$\pi(\theta|x) \propto p(x|\theta)\,\pi(\theta).$$

Eliminating all multiplicative constants in $p(x|\theta)\ \pi(\theta)$ that do not depend on θ, we get that

$$\pi(\theta|x) = c \cdot \theta^{x+\alpha-1}(1-\theta)^{n-x+\beta-1},$$

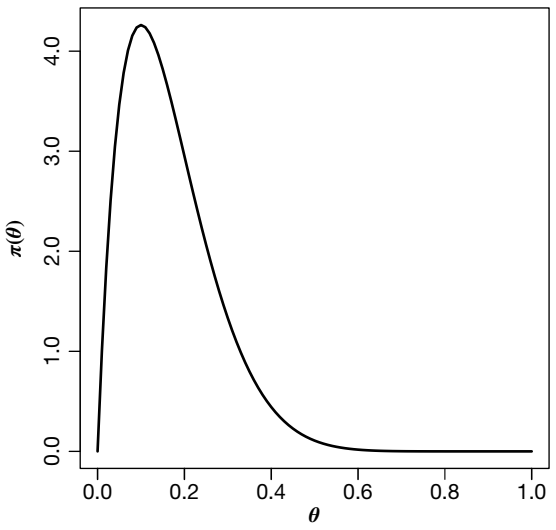

Figure 1. The prior $\pi(\theta)$ with α = 2 and β = 10. With this selection of α and β, the prior is peaked at low values of θ.

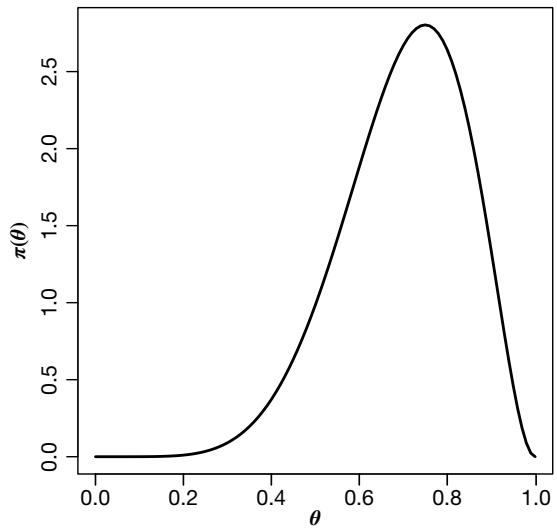

Figure 2. The prior $\pi(\theta)$ with α = 7 and β = 3. With this selection of α and β, the prior is peaked at high values of θ.

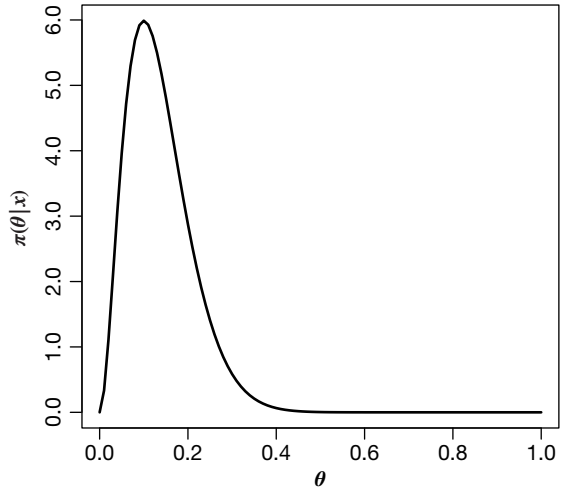

Figure 3. The posterior distribution with $n = 10$, $x = 1$, $\alpha = 2$, and $\beta = 10$.

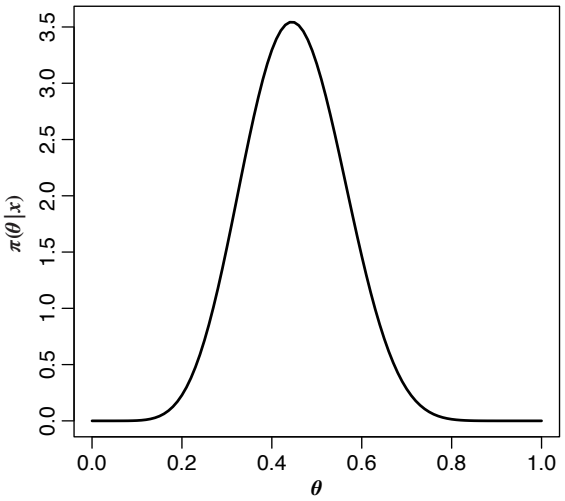

Figure 4. The posterior distribution with $n = 10$, $x = 2$, $\alpha = 7$, and $\beta = 3$.

where c is some constant such that

$$\int_0^1 c \cdot \theta^{x+\alpha-1} (1-\theta)^{n-x+\beta-1} \, d\theta = 1.$$

The posterior $\pi(\theta|x)$ takes the form of a beta distribution, making

$$c = \frac{\Gamma(\alpha+\beta+n)}{\Gamma(\alpha+x)\Gamma(\beta+n-x)}.$$

The posterior of θ is thus a beta distribution with parameters α^{pst} and β^{pst} where

$$\begin{aligned} \alpha^{\text{pst}} &= \alpha + x, \quad \text{and} \\ \beta^{\text{pst}} &= \beta + n - x. \end{aligned}$$

The plots in Figures 3 and 4 show the posteriors corresponding to the priors shown in Figures 1 and 2, respectively. In Figure 3, one success was observed after 10 trials, emphasizing even more that the value of θ is small. Observe how the posterior in this case is more peaked at small values of θ than the prior was. In Figure 4, two successes were observed in 10 trials, indicating that the probability of success was much smaller than the prior anticipated. Observe how, in this case, the peak of the posterior has significantly shifted to smaller values of θ.

The Prior and Posterior of the Entire System

Let us now put this problem in the context of one large system that is composed of several subsystems. If all the subsystems have to work for the entire system to work, how do the posterior distributions of the subsystem reliabilities inform the distribution of the total system's reliability? And how would testing the entire system (as a whole) affect the posterior of the total system reliability? The next two subsections address these questions.

Subsystem Test Sizing

This subsection focuses on how the distributions of the subsystem survival probabilities affect the distribution of the total system's survival probability. If there are S subsystems, and the entire system fails if any one of the subsystems fail (i.e., the subsystems work in series), then the success probability of the total system, $\theta_{\text{Tot Sys}}$, is calculated as

$$\begin{aligned} \theta_{\text{Tot Sys}} &= \mathbb{P}(\text{Success of Subsys 1}) \times \mathbb{P}(\text{Success of Subsys 2}) \times \cdots \\ &\quad \times \mathbb{P}(\text{Success of Subsys } S) \\ &= \prod_{j=1}^{S} \theta_j, \end{aligned} \tag{5}$$

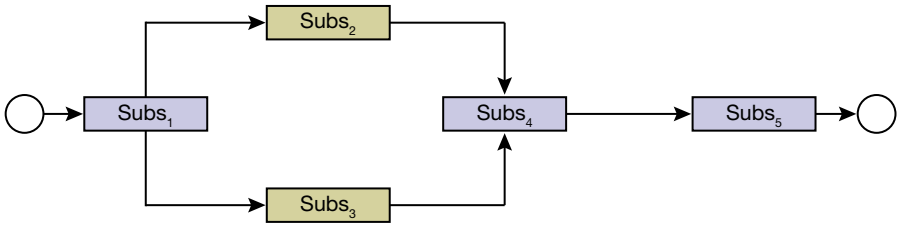

Figure 5. Flowchart of system composed of five subsystems (Subs), two of which work in parallel.

where θ_j is the success probability of the jth subsystem.

If $S = 5$ and the block diagram of the subsystems is as shown in Figure 5, then the system fails if subsystem 1, 4, 5, or both 2 and 3 fail. In this case, the success probability of the entire system would be calculated as

$$\begin{aligned}\theta_{\text{Tot Sys}} &= \mathbb{P}(\text{Success of Subsys 1}) \times \mathbb{P}(\text{Success of Subsys 2 or 3}) \times \mathbb{P}(\text{Success of Subsys 4}) \times \mathbb{P}(\text{Success of Subsys 5}) \\ &= \theta_1 (\theta_2 + \theta_3 - \theta_2\theta_3)\, \theta_4\theta_5.\end{aligned}$$

The value of $\theta_{\text{Tot Sys}}$ is thus the product and/or sum of beta random variables. The resulting distribution of a random variable such as $\theta_{\text{Tot Sys}}$ has been derived in a number of publications, but this distribution is very complicated and thus difficult to work with analytically.[1,10,11,12,13] The distribution of $\theta_{\text{Tot Sys}}$ is easy to work with and understand, however, using Monte Carlo methods. Since the posterior distribution of all the components of the system take the form of a beta distribution with known parameters, assuming independence of the subsystems, we can easily simulate n_{Sim} values from the prior distribution of $\theta_{\text{Tot Sys}}$. This requires simulating n_{Sim} S-tuples of (θ_1, θ_2, θ_3,..., θ_S). With each simulated S-tuple, we can calculate a value of $\theta_{\text{Tot Sys}}$. The algorithm for generating n_{Sim} values of $\theta_{\text{Tot Sys}}$ for a system in series is given in Procedure 1; that for a system as shown in Figure 5 is given in Procedure 2.

Figures 6–12 illustrate how the posterior distributions of subsystem reliability affect the distribution of $\theta_{\text{Tot Sys}}$. In the simulations performed, we assumed that the entire system was composed of three subsystems ($S = 3$) and

---

Procedure 1: Simulating n_{Sim} values of $\theta_{\text{Tot Sys}}$ when the subsystems work in series

input : $\pi(\theta_1|x_1), \pi(\theta_2|x_2), \ldots, \pi(\theta_S|x_S)$, and n_{Sim}, where x_i is the number of successes of subsystem i
output: $\theta^{(1)}_{\text{Tot Sys}}, \theta^{(2)}_{\text{Tot Sys}}, \ldots, \theta^{(n_{\text{Sim}})}_{\text{Tot Sys}}$
for $i \leftarrow 1$ **to** n_{Sim} **do**
  $\theta^{(i)}_{\text{Tot Sys}} \leftarrow 1$
  for $j \leftarrow 1$ **to** S **do**
    Generate $\theta^{(i)}_j \sim \pi(\theta_j|x_j)$
    $\theta^{(i)}_{\text{Tot Sys}} \leftarrow \theta^{(i)}_{\text{Tot Sys}} \cdot \theta^{(i)}_j$.

Procedure 2: Simulating n_{Sim} values of $\theta_{\text{Tot Sys}}$ when the five subsystems work as shown in Figure 14 (series and parallel)

input : $\pi(\theta_1|x_1), \pi(\theta_2|x_2), \ldots, \pi(\theta_S|x_S)$, and n_{Sim}, where x_i is the number of successes of subsystem i
output: $\theta^{(1)}_{\text{Tot Sys}}, \theta^{(2)}_{\text{Tot Sys}}, \ldots, \theta^{(n_{\text{Sim}})}_{\text{Tot Sys}}$
for $i \leftarrow 1$ **to** n_{Sim} **do**
  $\theta^{(i)}_2 \sim \pi(\theta_2|x_2)$
  $\theta^{(i)}_3 \sim \pi(\theta_3|x_3)$
  $\theta^{(i)}_{2|3} \leftarrow \theta^{(i)}_2 + \theta^{(i)}_3 - \theta^{(i)}_2 \cdot \theta^{(i)}_3$
  $\theta^{(i)}_{\text{Tot Sys}} \leftarrow \theta^{(i)}_{2|3}$
  for $j \in \{1,4,5\}$ **do**
    Generate $\theta^{(i)}_j \sim \pi(\theta_j|x_j)$
    $\theta^{(i)}_{\text{Tot Sys}} \leftarrow \theta^{(i)}_{\text{Tot Sys}} \cdot \theta^{(i)}_j$.

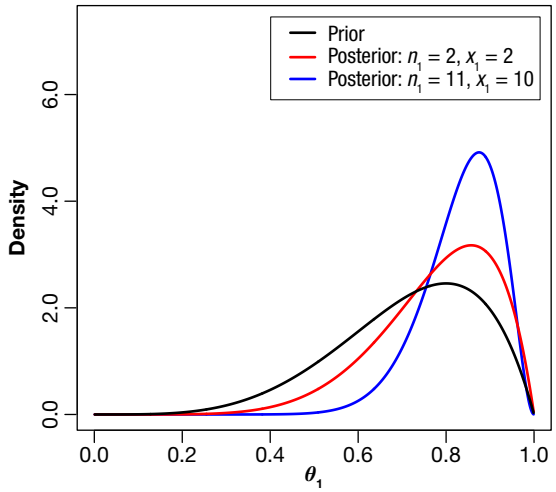

Figure 6. The prior and posterior of the first subsystem with $\alpha = 5$ and $\beta = 2$.

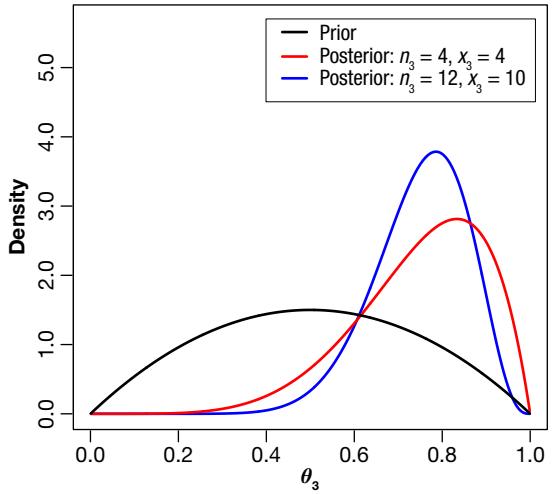

Figure 8. The prior and posterior of the third subsystem with $\alpha = 2$ and $\beta = 2$.

that these subsystems worked in series. The priors of the three subsystems are shown in black in Figures 6–8. The subsystems are then tested with $n_1 = 2$, $n_2 = 5$, and $n_3 = 4$, where n_j is the number of times the jth subsystem is tested. The resulting posteriors are shown in red in Figures 6–8, and 10,000 draws from the resulting distribution of $\theta_{\text{Tot Sys}}$ are shown in Figure 9. The subsystems were also tested at $n_1 = 11$, $n_2 = 14$, and $n_3 = 12$, and the corresponding posteriors are shown in blue in Figures 6–8. Observe that these posteriors are more peaked (more informed) than the others since the sample sizes are larger. The distribution of $\theta_{\text{Tot Sys}}$ corresponding to these larger sample sizes is shown in Figure 10. Observe how the variance of this posterior is smaller than that shown in Figure 9; this is because the subsystem sample sizes are larger.

Total System Test Sizing

Let us now investigate how testing the entire system (and not just its individual components) affects the posterior distribution of $\theta_{\text{Tot Sys}}$. Updating the prior distribution of $\theta_{\text{Tot Sys}}$ given test results on the total system is more challenging than updating the subcomponent values of θ because, in this case, the original distribution of $\theta_{\text{Tot Sys}}$ is not a beta distribution. Recall that the prior distribution of $\theta_{\text{Tot Sys}}$ was analytically challenging to work with and, as a result, was obtained using Monte Carlo methods. It is not uncommon for practitioners to approximate this prior with another (perhaps beta) distribution[1,11,14,15,16,17] to make the posterior analysis simpler and more convenient. Others redefine the priors of the independent components entirely just so the prior of the total system's reliability is analytically tractable.

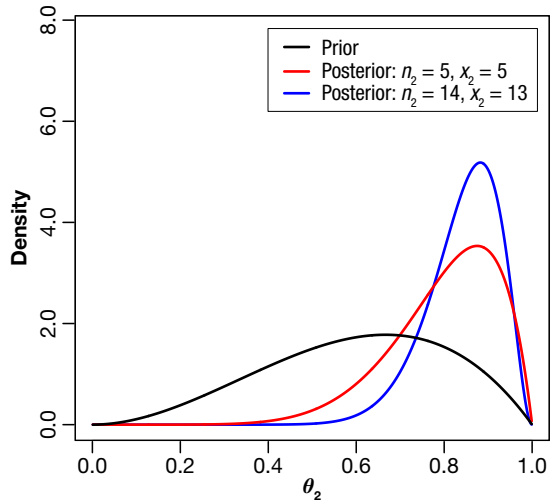

Figure 7. The prior and posterior of the second subsystem with $\alpha = 3$ and $\beta = 2$.

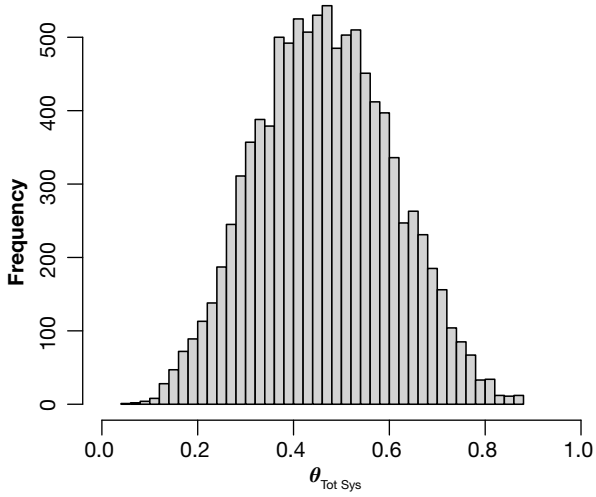

Figure 9. The resulting distribution of $\theta_{\text{Tot Sys}}$ when $n_1 = 2$, $n_2 = 5$, and $n_3 = 4$.

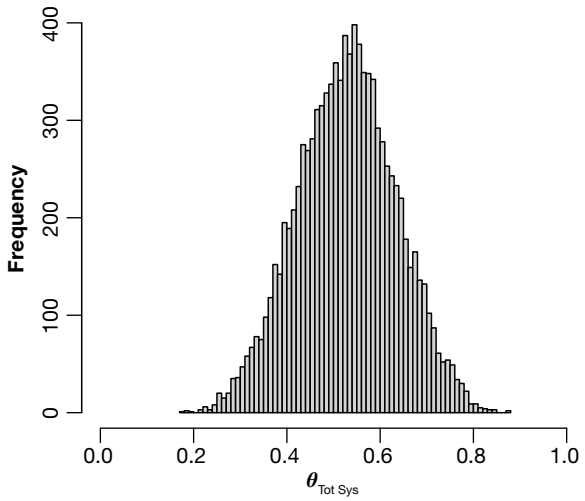

Figure 10. The resulting distribution of $\theta_{\text{Tot Sys}}$ when $n_1 = 11$, $n_2 = 14$, and $n_3 = 12$.

Zoh et al.,[18] for example, set the priors of the components to negative log-gamma distributions, yet they still had to execute complicated Markov chain Monte Carlo methods to do posterior analysis of the total system.

These workarounds and approximations to make posterior analysis easier are not necessary. This article describes a simple and quick Monte Carlo method that updates the prior distribution of $\theta_{\text{Tot Sys}}$ after testing the entire system. This method is a simple application of the algorithm proposed by Rubin.[19] Rubin observed that a sample from the posterior distribution of a parameter can be obtained by first generating values from its prior and then generating data conditioned on these sampled values. Those values of the parameter for which the generated data match the observed data follow the posterior distribution.

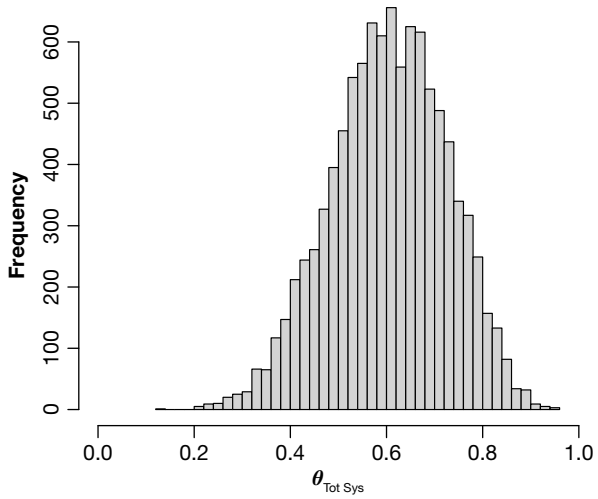

Figure 11. The resulting distribution of $\theta_{\text{Tot Sys}}$ when $n_1 = 2$, $n_2 = 5$, $n_3 = 4$, and $n_{\text{TS}} = 4$.

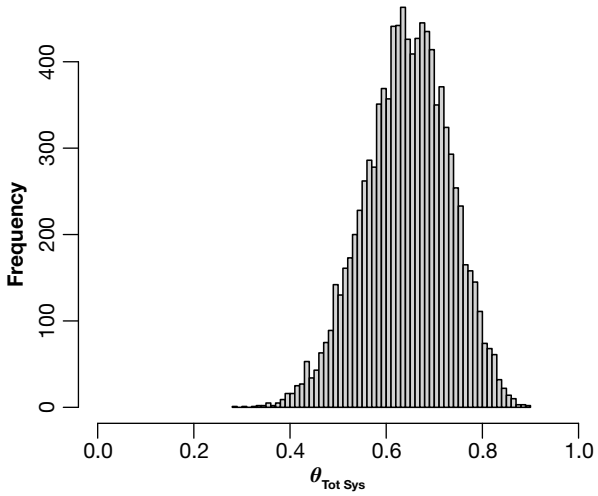

Figure 12. The resulting distribution of $\theta_{\text{Tot Sys}}$ when $n_1 = 11$, $n_2 = 14$, $n_3 = 12$, and $n_{\text{Tot}} = 7$.

To apply this algorithm in our case, we begin by writing the posterior for $\theta_{\text{Tot Sys}}$ as

$$\pi\left(\theta_{\text{Tot Sys}}|x_{\text{TS}}\right) \propto p\left(x_{\text{TS}}|\theta_{\text{Tot Sys}}\right)\pi\left(\theta_{\text{Tot Sys}}\right),$$

where

$$p\left(x_{\text{TS}}|\theta_{\text{Tot Sys}}\right) = \binom{n_{\text{TS}}}{x_{\text{TS}}}\theta_{\text{Tot Sys}}^{x_{\text{TS}}}\left(1-\theta_{\text{Tot Sys}}\right)^{n_{\text{TS}}-x_{\text{TS}}}, \quad (6)$$

and n_{TS} and x_{TS} are the number of tests (and successes) of the total system. Given that x_{TS}^* successes have been observed from n_{TS} trials of the total system, we sample from the posterior $\pi(\theta_{\text{Tot Sys}}|x_{\text{TS}} = x_{\text{TS}}^*)$ by first simulating from the prior of $\theta_{\text{Tot Sys}}$, $\pi(\theta_{\text{Tot Sys}})$. (An algorithm similar to the ones shown in Procedure 1 or 2 could be used to sample from the prior, $\pi(\theta_{\text{Tot Sys}})$.) We then condition on these sampled values of $\theta_{\text{Tot Sys}}$ to generate candidate values of x_{TS} from the likelihood shown in Eq. 6. The simulated values of $\theta_{\text{Tot Sys}}$ for which the likelihood generates $x_{\text{TS}} = x_{\text{TS}}^*$ are then considered to be an exact sample from the posterior. The details of this algorithm (assuming the subsystems work in series; minor changes to the first "for" loop of the algorithm would be necessary if the subsystems did not work in series) are given in Procedure 3.

The plots in Figures 11 and 12 show how the distribution of $\theta_{\text{Tot Sys}}$ changes when tests on the entire system are executed. The plot in Figure 11 shows how the distribution of $\theta_{\text{Tot Sys}}$ changes from the distribution in Figure 9 when four successes are observed out of four tests on the entire system. Observe that with this extra evidence of success, the distribution of $\theta_{\text{Tot Sys}}$ shifts to the right. The same story is told in Figure 12. It shows how the distribution of $\theta_{\text{Tot Sys}}$ changes from the distribution in Figure 10 when five successes are observed out of seven tests on the entire system. This distribution moves to the right and is also more peaked.

Procedure 3: Simulating n_{Sim} values of $\theta_{\text{Tot Sys}}$ from $\pi\left(\theta_{\text{Tot Sys}} | x_{\text{TS}} = x_{\text{TS}}^*\right)$

input : $\pi(\theta_1|x_1), \pi(\theta_2|x_2), \ldots, \pi(\theta_S|x_S), n_{\text{Sim}}, n_{\text{TS}}, x_{\text{TS}}^*$
output: $\theta_{\text{Tot Sys}}^{(1)}, \theta_{\text{Tot Sys}}^{(2)}, \ldots, \theta_{\text{Tot Sys}}^{(n_{\text{Sim}})}$
$i \leftarrow 1$
while $i < n_{\text{Sim}}$ **do**
  $\theta_{\text{Tot Sys}}^{(i),\ \text{cand}} \leftarrow 1$
  for $j \leftarrow 1$ **to** S **do**
    Generate $\theta_j^{(i)} \sim \pi(\theta_j|x_j)$
    $\theta_{\text{Tot Sys}}^{(i),\ \text{cand}} \leftarrow \theta_{\text{Tot Sys}}^{(i),\ \text{cand}} \cdot \theta_j^{(i)}$.
  Generate $x_{\text{TS}} \sim \text{Binomial}\left(n_{\text{TS}}, \theta_{\text{Tot Sys}}^{(i),\ \text{cand}}\right)$.
  if $x_{\text{TS}} = x_{\text{TS}}^*$ **then**
    $\theta_{\text{Tot Sys}}^{(i)} \leftarrow \theta_{\text{Tot Sys}}^{(i),\ \text{cand}}$
    $i \leftarrow i+1$

It is critical to understand the distinction between the two examples discussed above. In the first case (with the resulting distribution of $\theta_{\text{Tot Sys}}$ shown in Figure 11), the total system was tested four times and four successes were observed. Because the system works in series, a successful test of the entire system implies a successful test of each component. The posterior distribution of $\theta_{\text{Tot Sys}}$ can thus easily be calculated by simply updating the posteriors of the system's three components and then applying Procedure 1. This is not true for the second example (with the resulting distribution of $\theta_{\text{Tot Sys}}$ shown in Figure 12). Recall that in the second example, the entire system was tested seven times, but only five successes were observed. Since it is not clear which component(s) failed (causing the failure of the entire system), the posteriors of the components cannot be updated, and Procedure 1 cannot be applied. In this case, Procedure 3 is necessary in calculating the posterior of $\theta_{\text{Tot Sys}}$.

EXAMPLE

This example illustrates the efficiency of Rubin's algorithm when calculating total system reliability. Consider two types of systems/fault trees, each with m different types of components. The first fault tree works in series, and in the second, every other component operates in series. Figures 13 and 14 illustrate these fault trees.

The efficiency of Rubin's algorithm is demonstrated with a Monte Carlo study. We initially place Beta(α,β) priors on all the components with α = 999 and β = 0.5. (Such priors assume a prior reliability mean of 0.9995, which is not uncommon for highly reliable systems.) For each system/fault tree, and for a specific value of m, n_j (the number of trials for subsystem j) and x_j (the number of successful trials for subsystem j) can then be simulated. The number of trials is simulated from a Poisson(λ = 5) distribution, and the number of successful trials is simulated from a binomial distribution with parameters n_j

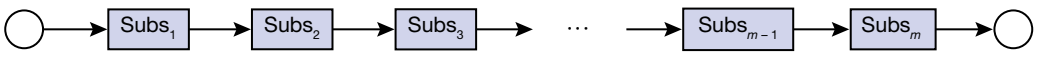

Figure 13. Flowchart of system composed of *m* subsystems working in series.

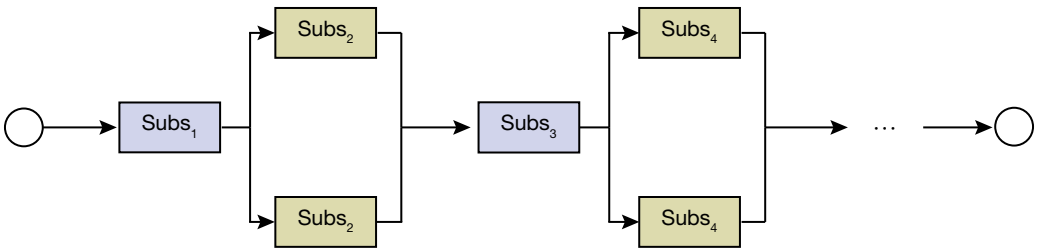

Figure 14. Flowchart of system composed of *m* subsystems with every other subsystem working in series.

Table 1. Average time (in seconds) to generate 10,000 values from the posterior of the total system's reliability

m	Fault Tree 1	Fault Tree 2
10	0.16	0.16
20	0.30	0.31
30	0.45	0.46
40	0.62	0.62
50	0.76	0.77

and $p_j = \alpha/(\alpha + \beta) = 999/(999 + 0.5)$. With these simulated test sizes, and assuming that the entire system is tested twice (with both tests being a success), the time it took to generate 10,000 draws from the posterior of the total system's reliability can be calculated. Table 1 shows the results of doing this 100 times and calculating the average time it took to generate these posterior values for m = 10, 20, 30, 40, and 50.

It is clear from Table 1 that this algorithm efficiently calculates the posterior of the total system's reliability.

CONCLUSION

This article reviews some of the methodologies related to Bayesian reliability. It initially focuses on success/failure data of systems and their subsystems. The article specifically addresses how the number of subsystem (or total system) tests affects the reliability of the entire system. It also presents a simple and efficient Monte Carlo method that can be employed to update the prior of a total system's reliability when only data from the total system are available.

REFERENCES

1 H. F. Martz, R. A. Waller, and E. T. Fickas, "Bayesian reliability analysis of series systems of binomial subsystems and components," *Technometrics*, vol. 30, no. 2, pp. 143–154, 1988, https://doi.org/10.1080/00401706.1988.10488361.

2 H. Cao, C. Yao, and Y. Yuan, "Bayesian approach for design and analysis of medical device trials in the era of modern clinical studies," *Med. Rev.*, vol. 3, no. 5, 2023, pp. 408–424, https://doi.org/10.1515/mr-2023-0026.

3 T. R. Bayes, "An essay towards solving a problem in the doctrine of chances," *Philos. Trans. Roy. Soc. London*, vol. 53, pp. 370–418, 1763, https://doi.org/10.1098/rstl.1763.0053.

4 A. Gelman, J. Carlin, H. Stern, D. Dunson, A. Vehtari, and D. Rubin, *Bayesian Data Analysis*, 3rd ed. New York: Chapman and Hall/CRC, 2013.

5 B. Carlin and T. Louis, *Bayes and Empirical Bayes Methods for Data Analysis*, 3rd ed. New York: Chapman and Hall, 2008.

6 L. Leoni, F. B. Toroody, S. Khalaj, F. De Carlo, A. B. Toroody, and M. Abaei, "Bayesian estimation for reliability engineering: Addressing the influence of prior choice," *Int. J. Environ. Res. Public Health*, vol. 18, no. 7, art. 3349, 2021, https://doi.org/10.3390/ijerph18073349.

7 S. Burke and M. Harman, "Bayesian reliability for complex systems," Scientific Test and Analysis Techniques Center of Excellence (STAT COE) Report, STAT Center of Excellence, Wright-Patterson Air Force Base, OH Aug. 31 2019, https://www.afit.edu/stat/statcoe_files/Bayesian%20Reliability%20for%20Complex%20Systems1.pdf.

8 H. F. Martz, R. A. Wailer, and E. T. Fickas, "Bayesian reliability analysis of series systems of binomial subsystems and components," *Technometrics*, vol. 30, no. 22, pp. 143–154, 1988, https://doi.org/10.1080/00401706.1988.10488361.

9 V. E. Johnson, T. Graves, M. S. Hamada, and C. S. Reese, "A hierarchical model for estimating the reliability of complex systems," in *Bayesian Statistics 7: Proc. Seventh Valencia Int. Meeting*, eds. J. Bernardo, M. Bayarri, J. Berger, A. David, D. Heckermann, A. Smith, and M. West, 2003, pp. 199–213.

10 D. Fan, "The distribution of the product of independent beta variables," *Commun. Stat. Theory Methods*, vol. 20, no. 12, pp. 4043–4052, 1991, https://doi.org/10.1080/03610929108830755.

11 H. F. Martz, R. A. Waller, and E. T. Fickas, "Bayesian reliability analysis of complex series/parallel systems of binomial subsystems and components," *Technometrics*, vol. 32, no. 4, pp. 407–416, 1990, https://doi.org/10.1080/00401706.1990.10484727.

12 M. D. Springer and W. E. Thompson, "The distribution of products of beta, gamma, and Gaussian random variables," *SIAM J. Appl. Math.*, vol. 18, no. 4, pp. 721–737, 1970, https://doi.org/10.1137/0118065.

13 J. Tang and A. K. Gupta, "On the distribution of the product of independent beta random variables," *Statist. Probability Lett.*, vol. 2, no. 3, pp. 165–168, 1984, https://doi.org/10.1016/0167-7152(84)90008-7.

14 A. R. Abdel-Wahid and A. Winterbottom, "The approximation of system reliability posterior distributions," *J. Statist. Planning Inference*, vol. 16, pp. 267–275, 1987, https://doi.org/10.1016/0378-3758(87)90079-6.

15 C. Coelho, "The generalized near-integer gamma distribution: A basis for 'near-exact' approximations to the distribution of statistics which are the product of an odd number of independent beta random variables," *J. Multivariate Anal.*, vol. 89, no. 2, pp. 191–218, 2004, https://doi.org/10.1016/j.jmva.2003.12.001.

16 F. Marques, I. Ghosh, J. Ferreira, and A. Bekker, "A note on the product of independent beta random variables," in *Advances in Statistics—Theory and Applications. Emerging Topics in Statistics and Biostatistics*, eds. I. Ghosh, N. Balakrishnan, H. Ng, 2021, Springer, pp. 69–83.

17 J. Tukey and S. Wilks, "Approximation of the distribution of the product of beta variables by a single beta variable," *Ann. Math. Statist.*, vol. 17, no. 3, pp. 318–324. 1946, https://doi.org/10.1214/aoms/1177730944.

18 R. Zoh, A. Wilson, S. Vander Wiel, and E. Lawrence, "The negative log-gamma prior distribution for Bayesian assessment of system reliability," in *Proc. Institution Mech. Eng., Part O: J. Risk Rel.*, vol. 232, no. 3, pp. 308–319, 2018, https://doi.org/10.1177/1748006X17692154.

19 D. B. Rubin, "Bayesianly justifiable and relevant frequency calculations for the applied statistician," *Ann. Statist.*, vol. 12, no. 4, pp. 1151–1172, 1984, https://doi.org/10.1214/aos/1176346785.

Carsten H. Botts, Air and Missile Defense Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Carsten H. Botts is an analyst and statistician in APL's Air and Missile Defense Sector. He holds a BS in chemistry from Georgetown University and a PhD, in statistics from Iowa State University. His work centers on analyzing and modeling weapon accuracy using modern computational and statistical methods. His email address is carsten.botts@jhuapl.edu.

Tracking Methods for Converted Radar Measurements

Alexander J. Pei

ABSTRACT

Target tracking is a critical component in defense and airspace protection. To provide awareness of potential enemy threats through target tracking, dynamic states are repeatedly updated based on observations. Because common dynamic models of moving objects typically use Cartesian coordinates, target tracking systems typically use this coordinate system as well. This presents a statistical challenge, however, when observations are recorded with different coordinate systems. This is the case with radar measurements, which use spherical coordinates (range, bearing, and elevation) instead of Cartesian coordinates (x, y, z). The main problem is integrating the statistics of new measurements with a priori state estimates to provide an updated a posteriori estimate. This article focuses on a converted-measurement approach to compute descriptive Cartesian statistics from spherical measurements for updates in a linear tracking system. Converted-measurement tracking, compared with mixed-coordinate tracking, can facilitate multisensor fusion in complex sensor networks. Various converted-measurement methods were evaluated, including Taylor approximations, unscented transforms, and debiased statistical methods, in a simple tracking scenario. Tracking performance varied across these three methods depending on the geometry of the scenario, so users of converted-measurement methods should evaluate the performance of each method for their given domain and application.

INTRODUCTION

Airspace protection starts with a comprehensive understanding of the potential threats. This involves understanding not only the location of objects but also how the objects' motion evolves over time, enabling differentiation between a fast-moving missile and a slow-moving weather balloon, for example. Tracking a moving object involves real-time updates to its dynamic states. A dynamic model, typically expressed in Cartesian coordinates and described by Newton's equations of motion, is used to model and predict the trajectory of an object.

Tracking using Kalman filtering techniques iteratively compares the predicted dynamic state of the trajectory with measurements of the dynamic states to update the state estimation at a particular time.[1] A common challenge is that the measurements of the dynamic states

are not direct; rather, they are indirect measurements of the true dynamic state through some physical phenomenon. Radar measurements are a classic example of indirect measurements of Cartesian coordinates. Instead of directly measuring Cartesian coordinates, radars provide range, bearing, and elevation (RBE) through the reflection of transmitted radio waves off the target object.[2]

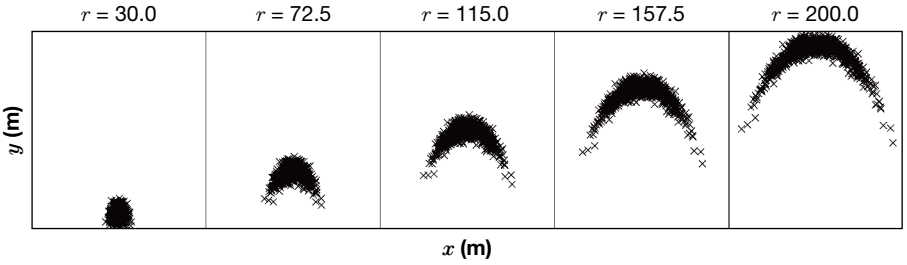

Figure 1. Effect of spherical-to-Cartesian-coordinate transformation at varying ranges. $\sigma_r = 5$, $\sigma_\theta = 0.1$, $\theta = 0$.

Integrating indirect measurements to update the estimated dynamic state is not a trivial problem. With limited measurements, it might not be possible to completely observe the dynamic states; for instance, if a radar provides only range and bearing, the z-coordinate cannot be inferred.[3] Additionally, the nature of the indirect measurement modality, specifically function mapping the states to these measurements, plays a crucial role in the measurements' integration into state estimation by the tracker. This is because tracking is done in a statistical framework, where there is uncertainty associated with each estimated state. Both the ensemble of predicted measurement uncertainties and the actual measurement uncertainties contribute to the overall uncertainty in the estimated state.

Herein lies the problem: If the states and measurements are in different coordinate systems, the uncertainty of the updated state may not be well defined.[4] Figure 1 highlights the nature of this problem, where points are drawn from a Gaussian distribution in spherical coordinates and converted to Cartesian coordinates. In Cartesian space, the resulting distribution of points no longer forms an ellipse, indicating that it cannot be accurately represented by a Gaussian distribution. As a result, traditional linear tracking techniques cannot be applied, and the state estimation is no longer optimal.

This problem can be resolved using extended Kalman filters (EKF), unscented Kalman filters (UKF), or particle filters (PF) to handle the nonlinearity introduced by the spherical-to-Cartesian-coordinate transformation.[1,5] A critical aspect of these filtering techniques is the calculation of the innovation, which is the difference between the predicted measurement and the actual measurement. This method of tracking is referred to as mixed-coordinate tracking since the dynamic state space and the measurement space are both maintained throughout the update equations.[4] An alternative method is converted-measurement tracking, where the measurements are converted to the coordinate system the state space uses, allowing for linear Kalman filtering (assuming that the transition function is linear).[4]

Regardless of the tracking method, the problem of transforming the statistics of random variables from one coordinate system to another persists. Although mixed-coordinate tracking is a valid method, practical constraints may limit its usage. In general, mixed-coordinate methods require greater computational power to handle nonlinearities.[6] In addition, multisensor fusion and integration may be easier in a converted-measurement framework, where multiple nonlinear observations are abstracted away before measurement fusion. For example, the potential location of a target from a bistatic radar is an ellipse around the transmitter and receiver pair. Although this measurement is not informative by itself, multiple bistatic radars can localize a target through the intersection of ellipses and provide a single measurement of the target location in Cartesian coordinates to update the tracker.[7] Figure 2 shows a hypothetical scenario fusing both monostatic and bistatic radar measurements to update a track after converting both to Cartesian coordinates.[8]

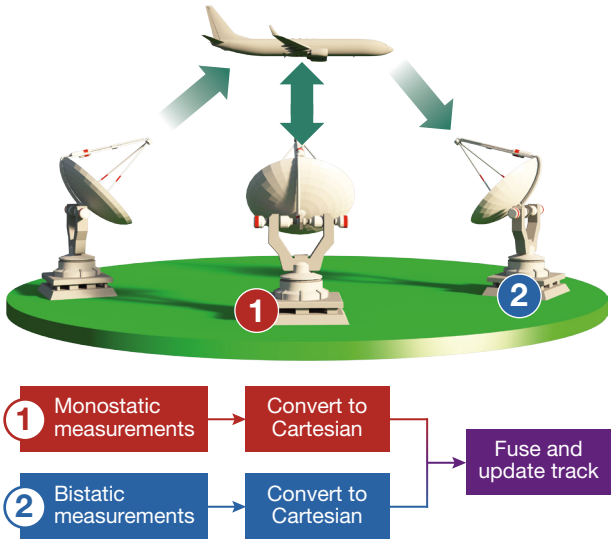

Figure 2. Hypothetical multisensor system tracking a target using both (1) monostatic and (2) bistatic radar measurements. After conversion to Cartesian coordinates, the resulting state estimates and uncertainties can be fused to simultaneously update a track.

This article presents converted-measurement methods for radar tracking applications. First it shows that converting spherical to Cartesian coordinates introduces a bias in the converted measurements and describes how these biases can be accounted for. It then presents three ways to propagate the spherical statistics to Cartesian statistics: debiased measurements, Taylor approximations, and unscented transforms. The three methods were tested in a simple tracking scenario to highlight the strengths and weaknesses of each in practice.

BACKGROUND

Linear State Estimation

Consider a dynamical system with a linear transition function $\mathbf{F}$ and linear observation function $\mathbf{H}$:

$$\boldsymbol{x}_{k+1} = \mathbf{F}\boldsymbol{x}_k + \boldsymbol{w}_k \tag{1}$$

$$\boldsymbol{y}_k = \mathbf{H}\boldsymbol{x}_k + \boldsymbol{v}_k \tag{2}$$

The Kalman filter update equations provide an optimal estimate (in the mean-squared error sense) of the state variables $\boldsymbol{x}_k$, as well as the uncertainties associated with these estimates. The *a priori* and *a posteriori* estimates of the state variables at time k are denoted $\boldsymbol{x}_k^-$ and $\boldsymbol{x}_k^+$ with covariance matrices $\mathbf{P}_k^-$ and $\mathbf{P}_k^+$. These two state estimates represent the optimal estimation of the state variables before and after an observation is made at time k. The model process noise and measurement noise variables $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ are zero-mean with covariance matrices $\mathbf{Q}_k$ and $\mathbf{R}_k$. The optimality is contingent on the following critical assumptions: Both the transition and observation functions are linear, and the state and noise variables are normally distributed. These assumptions yield the Kalman filter update equations[1]:

$$\boldsymbol{x}_k^- = \mathbf{F}\boldsymbol{x}_{k-1}^+ \tag{3}$$

$$\mathbf{P}_k^- = \mathbf{F}\mathbf{P}_{k-1}^+\mathbf{F}^\top + \mathbf{Q}_k \tag{4}$$

$$\mathbf{S}_k = \mathbf{H}\mathbf{P}_k^-\mathbf{H}^\top + \mathbf{R}_k \tag{5}$$

$$\mathbf{K}_k = \mathbf{P}_k^-\mathbf{H}^\top\mathbf{S}_k^{-1} \tag{6}$$

$$\boldsymbol{x}_k^+ = \boldsymbol{x}_k^- + \mathbf{K}_k(\boldsymbol{y}_k - \mathbf{H}\boldsymbol{x}_k^-) \tag{7}$$

$$\mathbf{P}_k^+ = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{P}_k^- \tag{8}$$

Eq. 3 shows how the *a priori* estimate $\boldsymbol{x}_k^-$ is calculated by propagating the *a posteriori* estimate from the previous time point through the linear transition function. The covariance of this state estimate is given in Eq. 4, which incorporates the uncertainties propagated from the previous time point using the transition function ($\mathbf{F}\mathbf{P}_{k-1}^+\mathbf{F}^\top$) along with the process noise of the system $\mathbf{Q}_k$. The resulting $\boldsymbol{x}_k^-$ can also be thought of as the predicted state variable at time k and can be used to generate the predicted measurement $\mathbf{H}\boldsymbol{x}_k^-$ via the observation function. The covariance of the predicted measurement $\mathbf{S}_k$ is given in Eq. 5 and accounts for the uncertainties of the predicted measurement and the measurement noise. The Kalman gain in Eq. 6 determines how much weight the measurement $\boldsymbol{y}_k$ should have when updating the state estimate. For illustration purposes, consider the case where the state is one-dimensional and $\mathbf{H} = \mathbf{1}$. The Kalman gain is the ratio of variances between the *a priori* state estimate variance and the predicted measurement variance. A higher Kalman gain places greater weight on the innovation term $\boldsymbol{y}_k - \mathbf{H}\boldsymbol{x}_k^-$ when calculating the *a posteriori* state estimate in Eq. 7. The covariance of the *a posteriori* estimate in Eq. 8 is a scaled version of the *a priori* covariance, where the scale is determined by the relative values of the *a priori* noise and the measurement noise.

It is useful to reiterate that the *a priori* covariance matrix $\mathbf{P}_k^- := \mathbb{E}\left[(\boldsymbol{x}_k - \boldsymbol{x}_k^-)(\boldsymbol{x}_k - \boldsymbol{x}_k^-)^\top\right]$ reflects the uncertainty in the estimation of the true state variable. The same definition holds for the *a posteriori* state estimate; however, this estimate will have lower uncertainty since it incorporated the information received from the measurement. If the *a priori* states and covariance matrices are normally distributed, the resulting updated *a posteriori* states and covariances will also be normally distributed, and the estimates will remain optimal.[5]

Linear State Transition with Nonlinear Observation

If the strict linear assumptions about the transition function or observation functions are violated, the statistics for the state estimates are no longer normally distributed, and the Kalman filter update equations do not provide an optimal estimate. The remainder of this article assumes a linear state transition function with a nonlinear observation function:

$$\boldsymbol{x}_{k+1} = \mathbf{F}\boldsymbol{x}_k + \boldsymbol{w}_k \tag{9}$$

$$\boldsymbol{y}_k = h(\boldsymbol{x}_k) + \boldsymbol{v}_k \tag{10}$$

This is common in radar tracking applications, where the dynamic models typically use a Cartesian coordinate system and the measurements use a spherical coordinate system. The function $\boldsymbol{h}(\cdot)$ converts the predicted state vector into a predicted measurement. This poses an issue: The Kalman filter considers the uncertainties from the predicted state and the measurement (as shown in Eq. 5), and this fusion is not trivial if these uncertainties are not expressed using the same coordinate system. Mixed-coordinate systems handle this problem in a variety of ways. EKFs make a linear approximation to the observation function and subsequently use linear

Kalman filter update equations.[9] This approximation can suffer if the observation function is highly nonlinear. UKFs use an unscented transform (UT) to estimate the statistics of the transformed random variable through the observation function by using a set of carefully selected and weighted points.[10] PFs use sampling to calculate the posterior distribution of the state after new information is acquired in a Bayesian framework.[5] In linear Kalman filtering, the posterior distribution (specifically the evidence term) is tractable and has a closed-form solution. Thus, it is theoretically possible to achieve optimal nonlinear filtering using PFs, given an infinite number of samples.

DEBIASED CONVERTED MEASUREMENTS

Converted-measurement techniques deal with the nonlinearity in Eq. 2 by converting the measurements to the same coordinate system the state space uses. However, converting measurements from spherical to Cartesian coordinates introduces an unintentional bias, as shown in Figure 3. The following section contains equations from Lerro and Bar-Shalom,[4] and a full derivation for the equations can be found in the original paper. An abridged version is presented here for readability, alongside figures for visualization. Consider a measurement with range $\boldsymbol{r}$ and bearing θ. The measured range and bearing can be thought of as the true underlying measurements with additive noise:

$$r_m = r + v_r \tag{11}$$

$$\theta_m = \theta + v_\theta \tag{12}$$

where $\boldsymbol{r_m}$ and $\boldsymbol{\theta_m}$ are the measured range and bearing, and $\boldsymbol{v_r}$ and $\boldsymbol{v_\theta}$ are the zero-mean additive noise terms with variance $\boldsymbol{\sigma_r^2}$ and $\boldsymbol{\sigma_\theta^2}$. The converted measurements can then be expressed as deviations away from the true Cartesian position:

$$x_m = x + v_x = r_m \cos\theta_m \tag{13}$$

$$y_m = y + v_y = r_m \sin\theta_m \tag{14}$$

where $\boldsymbol{x}$ and $\boldsymbol{y}$ are the true Cartesian coordinates with additive noise $\boldsymbol{v_x}$ and $\boldsymbol{v_y}$. The goal is to properly model the statistics of $\boldsymbol{v_x}$ and $\boldsymbol{v_y}$ to use during the Kalman filter update equations. The mean value of the errors are:

$$\mathbb{E}[v_x] = r\cos\theta(e^{-\sigma_\theta^2/2} - 1) \tag{15}$$

$$\mathbb{E}[v_y] = r\sin\theta(e^{-\sigma_\theta^2/2} - 1) \tag{16}$$

with variances and covariances:

$$\sigma_x^2 = r^2 e^{-\sigma_\theta^2}\left[\cos^2\theta(\cosh(\sigma_\theta^2) - 1) + \sin^2\theta\sinh(\sigma_\theta^2)\right] + \sigma_r^2 e^{-\sigma_\theta^2}\left[\cos^2\theta\cosh(\sigma_\theta^2) + \sin^2\theta\sinh(\sigma_\theta^2)\right] \tag{17}$$

$$\sigma_y^2 = r^2 e^{-\sigma_\theta^2}\left[\sin^2\theta(\cosh(\sigma_\theta^2) - 1) + \cos^2\theta\sinh(\sigma_\theta^2)\right] + \sigma_r^2 e^{-\sigma_\theta^2}\left[\sin^2\theta\cosh(\sigma_\theta^2) + \cos^2\theta\sinh(\sigma_\theta^2)\right] \tag{18}$$

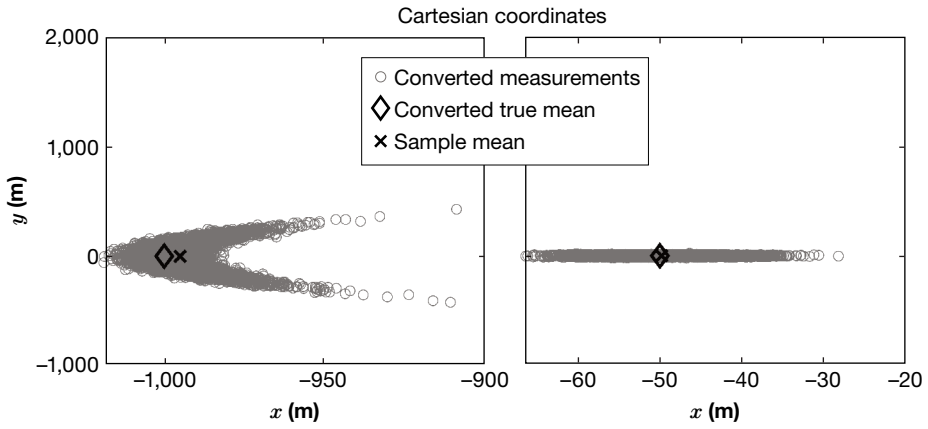

Figure 3. Converted-measurement bias. Ten thousand sample points for a given range and bearing were drawn, with a bearing of π and $\sigma_\theta = 0.1$, $\sigma_r = 5$. Each sample point was converted to $\boldsymbol{x}$ and $\boldsymbol{y}$ coordinates using $\boldsymbol{x} = r\cos\theta$ and $\boldsymbol{y} = r\sin\theta$. The true mean was converted and compared with the calculated sample mean of the transformed points. Left, $r = 1{,}000$ m. Right, $r = 50$ m. The bias offset is more dramatic at farther ranges.

$$\sigma_{xy} = \sin\theta\cos\theta e^{-2\sigma_\theta^2}\left[\sigma_r^2 + r^2(1 - e^{\sigma_\theta^2})\right] \tag{19}$$

There are a few important points about the statistics of the Cartesian errors. The first is that Eqs. 15 and 16 show that the expectation of the errors is nonzero; this is an issue for Kalman filtering, which assumes zero-mean errors. The second is that the Cartesian errors are coupled since changes in range or bearing affect both variances in Cartesian. This is also demonstrated by the fact that the cross-covariance of $\boldsymbol{v_x}$ and $\boldsymbol{v_y}$ is nonzero. Cross-covariance between $\boldsymbol{x}$ and $\boldsymbol{y}$ can make decoupled tracking[11] difficult, which is desirable for easier tuning and computational needs. These equations also quantify how uncertainties in $\boldsymbol{x}$ and $\boldsymbol{y}$ increase as a function of range for a given bearing value, as shown by the $\boldsymbol{r}^2$ term in Eq. 17, 18, and 19.

Finally, and perhaps most importantly, the statistics of the Cartesian errors depend on the true range and bearing, which are obviously not acquirable in realistic tracking scenarios. Instead, the same process can be repeated but using the measured range and bearing as proxies for the true range and bearing. The expected value of the errors given the measurements is:

$$\mathbb{E}[v_x \mid r_m, \theta_m] = r_m \cos\theta_m (e^{-\sigma_\theta^2} - e^{-\sigma_\theta^2/2}) \tag{20}$$

$$\mathbb{E}[v_y \mid r_m, \theta_m] = r_m \sin\theta_m (e^{-\sigma_\theta^2} - e^{-\sigma_\theta^2/2}) \tag{21}$$

Thus, the debiased corrected coordinate conversion is:

$$\tilde{x}_m = x_m - \mathbb{E}[v_x \mid r_m, \theta_m] \tag{22}$$

$$\tilde{y}_m = y_m - \mathbb{E}[v_y \mid r_m, \theta_m] \tag{23}$$

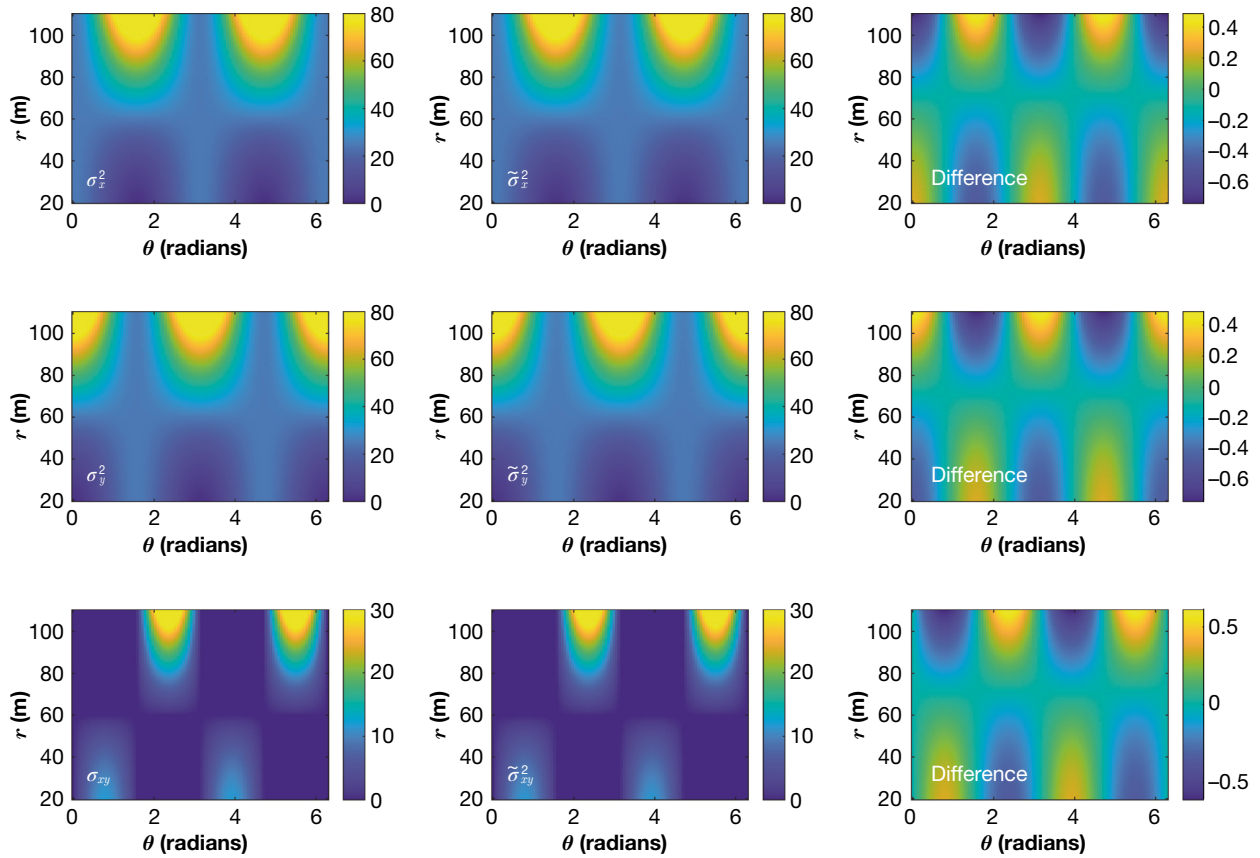

Figure 4. The variances and covariances of the true and debiased error with $\sigma_r = 5$ and $\sigma_\theta = 0.1$. Left plots show the variance values for the true error. Middle plots show the variance values for the debiased error. Right plots show the difference in variance between the left and right plots. Variance in all dimensions increases as a function of range because of the $\boldsymbol{r}^2$ term, which results from angle errors being amplified at farther ranges. For a given range value, there are peaks and troughs in variances as a function of bearing.

with variances and covariances:

$$\begin{aligned}\tilde{\sigma}_x^2 =& r_m^2 e^{-2\sigma_\theta^2}\left[\cos^2\theta_m(\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \sin^2\theta_m(\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)\right] \\ &+ \sigma_r^2 e^{-2\sigma_\theta^2}\left[\cos^2\theta_m(2\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \sin^2\theta_m(2\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)\right]\end{aligned} \tag{24}$$

$$\begin{aligned}\tilde{\sigma}_y^2 =& r_m^2 e^{-2\sigma_\theta^2}\left[\sin^2\theta_m(\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \cos^2\theta_m(\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)\right] \\ &+ \sigma_r^2 e^{-2\sigma_\theta^2}\left[\sin^2\theta_m(2\cosh 2\sigma_\theta^2 - \cosh\sigma_\theta^2) + \cos^2\theta_m(2\sinh 2\sigma_\theta^2 - \sinh\sigma_\theta^2)\right]\end{aligned} \tag{25}$$

$$\tilde{\sigma}_{xy}^2 = \sin\theta_m\cos\theta_m e^{-4\sigma_\theta^2}\left[\sigma_r^2 + (r_m^2 + \sigma_r^2)(1 - e^{\sigma_\theta^2})\right] \tag{26}$$

These equations provide an unbiased coordinate conversion to be used in a linear tracking regime. Figure 4 shows the variances using the true range and bearing values as well as the variances of the unbiased estimate using the measured range and bearing values. The expected value of the biased and debiased errors is shown in Figure 5. Increased areas of uncertainty for the measurements result in a noisier, less smooth track.

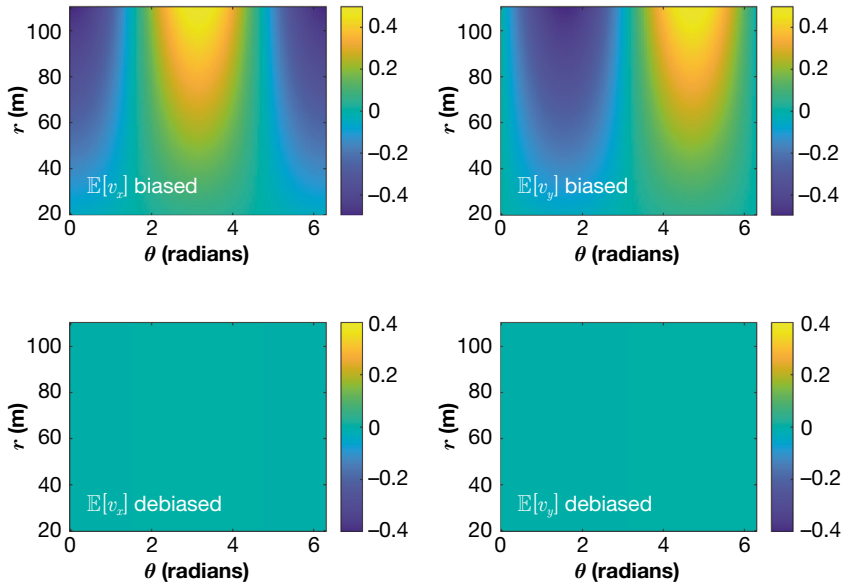

Figure 5. Expected value of the measurement conversion errors for $\boldsymbol{x}$ (left plots) and $\boldsymbol{y}$ (right plots) dimension before (top plots) and after (bottom plots) debiasing. Expected errors also increase as a function of range and vary sinusoidally as a function of bearing. Note the scale differences in the color axis.

TAYLOR APPROXIMATION OF THE SPHERICAL-TO-CARTESIAN TRANSFORM

Converting spherical measurements to Cartesian involves a nonlinear transformation as follows:

$$x = r\cos\theta \tag{27}$$

$$y = r\sin\theta \tag{28}$$

Let $\boldsymbol{f}(\boldsymbol{y})$ denote the function, which maps the measurement vector $\boldsymbol{y}$ containing the range and azimuth measurements to Cartesian coordinates. The notation with the scalar $\boldsymbol{y}$ is the Cartesian coordinate, whereas the vector $\boldsymbol{y}$ is the vector containing the observed spherical measurements. A local linear approximation around $\boldsymbol{y}$ perturbed around the mean by the noise values is as follows:

$$\boldsymbol{y} = \begin{bmatrix} r \\ \theta \end{bmatrix} = \begin{bmatrix} r_m - q_r \\ \theta_m - q_\theta \end{bmatrix}$$

$$\boldsymbol{q} = \begin{bmatrix} q_r \\ q_\theta \end{bmatrix} = \boldsymbol{y} - \boldsymbol{y}_m$$

$$f(\boldsymbol{y}) \approx f(\boldsymbol{y}_m) + \mathbf{J} \times (\boldsymbol{y} - \boldsymbol{y}_m) \tag{29}$$

$$\mathbf{J} = \begin{bmatrix} \frac{\partial x}{\partial r} & \frac{\partial x}{\partial \theta} \\ \frac{\partial y}{\partial r} & \frac{\partial y}{\partial \theta} \end{bmatrix} \Bigg|_{r=r_m, \theta=\theta_m} = \begin{bmatrix} \cos(\theta_m) & -r_m \sin(\theta_m) \\ \sin(\theta_m) & r_m \cos(\theta_m) \end{bmatrix} \tag{30}$$

where $\mathbf{J}$ denotes the Jacobian matrix containing the partial derivatives of $\boldsymbol{f}(\cdot)$ with respect to the spherical variables evaluated around the fixed point. The covariance of the measurement can then be written as:

$$\mathbf{P}_{\text{Cart}} := \mathbb{E}\left[(f(\boldsymbol{y}) - \mathbb{E}[f(\boldsymbol{y})])(f(\boldsymbol{y}) - \mathbb{E}[f(\boldsymbol{y})])^\top\right] \tag{31}$$

$$\approx \mathbb{E}\left[(\mathbf{J} \times (\boldsymbol{y} - \boldsymbol{y}_m) - \mathbb{E}[\mathbf{J} \times (\boldsymbol{y} - \boldsymbol{y}_m)])(\mathbf{J} \times (\boldsymbol{y} - \boldsymbol{y}_m) - \mathbb{E}[\mathbf{J} \times (\boldsymbol{y} - \boldsymbol{y}_m)])^\top\right] \tag{32}$$

$$= \mathbb{E}\left[(\mathbf{J}\boldsymbol{q} - \mathbb{E}[\mathbf{J}\boldsymbol{q}])(\mathbf{J}\boldsymbol{q} - \mathbb{E}[\mathbf{J}\boldsymbol{q}])^\top\right] \tag{33}$$

$$= \mathbf{J}\mathbb{E}\left[(\boldsymbol{q} - \mathbb{E}[\boldsymbol{q}])(\boldsymbol{q} - \mathbb{E}[\boldsymbol{q}])^\top\right]\mathbf{J}^\top \tag{34}$$

$$:= \mathbf{J}\mathbf{P}_{\text{Sphere}}\mathbf{J}^\top \tag{35}$$

where $\mathbf{P}_{\text{Sphere}}$ is the measurement covariance matrix for the range and bearing uncertainties. After applying this transformation, the measurements and descriptive statistics thereof are in Cartesian coordinates, and traditional linear Kalman filtering can be used. Note when calculating the innovation, which is the difference between the predicted state and the converted measurement, either the linear approximation to $\boldsymbol{f}(\boldsymbol{y})$ or the true nonlinear conversion can be used.

Even though the expected value of the errors $\mathbb{E}[\boldsymbol{f}(\boldsymbol{y}) - \boldsymbol{f}(\boldsymbol{y}_m)] = \mathbb{E}[\mathbf{J} \times (\boldsymbol{y} - \boldsymbol{y}_m)] = 0$, large errors in the linear approximation can occur at large ranges or with noisy bearing measurements and can cause biases in the estimation. This is shown in Figure 6, where the nonzero-mean errors in the approximation increase as a function of range. Similar to the debiased measurement, the errors also vary as a function of the bearing angle.

UNSCENTED TRANSFORM

As mentioned, the issue is the lack of knowledge of the statistics of the transformed random variable $\boldsymbol{f}(\boldsymbol{y})$. One heuristic Monte Carlo approach is to sample a large number of points from the normal distribution $\mathcal{N}(\boldsymbol{y}, \mathbf{P}_{\text{Sphere}})$ and apply the nonlinear observation function $\boldsymbol{f}(\cdot)$ to each of the points. This approach is similar to PFs in the sense that the propagation of points captures the statistics of the transformation through the function. The mean and covariance of all the resulting samples can be used as a proxy for the statistics of $\boldsymbol{f}(\boldsymbol{y})$ to update the linear Kalman filter equations. This approach is computationally limited, since the number of required points increases exponentially with the number of state variables in order to sufficiently sample the state space.

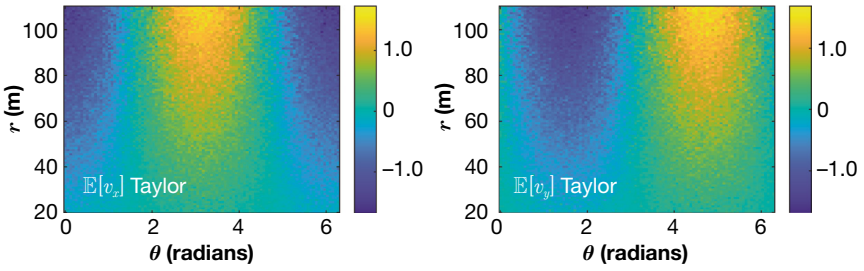

Figure 6. Expected value of the Taylor series approximation error for $\boldsymbol{x}$ (left) and $\boldsymbol{y}$ (right) with $\sigma_r = 5$ and $\sigma_\theta = 0.1$. A Monte Carlo approach was used to calculate the expected error by sampling 10,000 points in the spherical coordinate system and calculating the average error between the true $\boldsymbol{x}$ and $\boldsymbol{y}$ point and the converted points. These errors follow a pattern similar to that of the expected errors in the debiased approach; however, the axis of the color bar is larger for the Taylor approximation.

The UT typically used in UKFs takes a similar approach but carefully selects the sampled points to efficiently calculate the statistics of the transformed random variable. Let L be the number of variables in $\boldsymbol{y}$. A matrix $\mathcal{Y}$ of $2L + 1$ sigma points $\mathcal{Y}_i$ (each the size of the state vector) is constructed to sample the state space. The sigma points are passed through the observation function, after which the sample mean and covariance of the transformed points are used as an estimate for the statistics of $\boldsymbol{f}(\boldsymbol{y})$. Using the equations specified by Simon,[1] the sigma points are calculated as follows:

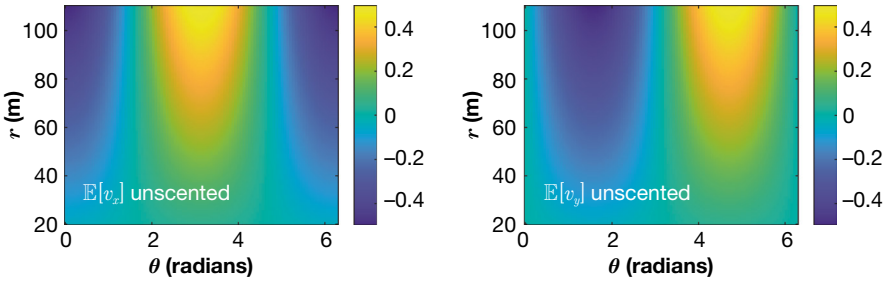

Figure 7. Expected value of the UT error for $\boldsymbol{x}$ (left) and $\boldsymbol{y}$ (right) with $\sigma_r = 5$ and $\sigma_\theta = 0.1$. The expected error matches the patterns and values of the true bias error.

$$\mathcal{Y}_0 = \boldsymbol{y} \tag{36}$$

$$\mathcal{Y}_i = \boldsymbol{y} + \sqrt{L}\mathbf{A}_i \qquad i \in \{1, \ldots, L\} \tag{37}$$

$$\mathcal{Y}_{L+i} = \boldsymbol{y} - \sqrt{L}\mathbf{A}_i \qquad i \in \{1, \ldots, L\} \tag{38}$$

where $\mathbf{P}_{\text{Sphere}} = \mathbf{A}\mathbf{A}^{\mathsf{T}}$ and $\mathbf{A}_i$ is the ith column of $\mathbf{A}$. To understand how the sigma points are calculated, consider the 1D case where there is one state variable ($L = 1$). The covariance matrix is thus a scalar value, and the square root of it gives the standard deviation. The two sigma points are then ±1 standard deviation around the mean value. In higher dimensions, consider the case where $\mathbf{P}_{\text{Sphere}}$ is a diagonal matrix. This means that the eigendecomposition of $\mathbf{P}_{\text{Sphere}} = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^{\mathsf{T}}$ is equal to the Cholesky decomposition[12] $\mathbf{P}_{\text{Sphere}} = \mathbf{A}\mathbf{A}^{\mathsf{T}}$. Thus the columns of $\mathbf{A}$ are the eigenvectors of $\mathbf{P}_{\text{Sphere}}$. As a result, each pair of sigma points symmetric about the mean (since it is plus and minus around the mean) is in equal and opposite directions along the axes of the ellipsoid defined by the original probability distribution. Loosely summarized, the characteristics of the original distribution are captured through the placement of these points along principal the axes of the ellipsoid defined by the covariance matrix.

The mean and covariance of $\boldsymbol{f}(\boldsymbol{y})$ are then estimated by:

$$\mathcal{X}_i = f(\mathcal{Y}_i) \tag{39}$$

$$\bar{\boldsymbol{x}} = \frac{1}{2L+1}\sum_{i=0}^{2L}\mathcal{X}_i \tag{40}$$

$$\mathbf{P}_{\text{Cart}} = \frac{1}{2L+1}\sum_{i=0}^{2L}\left(\mathcal{X}_i - \bar{\boldsymbol{x}}\right)\left(\mathcal{X}_i - \bar{\boldsymbol{x}}\right)^{\top} \tag{41}$$

where $\bar{\boldsymbol{x}}$ is the converted measurement in Cartesian using the UT. This particular formulation of the UT matches the moments of the transformed random variable up to the second order. This means that the terms up to the second order in the Taylor series expansion of the mean and covariance of $\boldsymbol{f}(\boldsymbol{y})$ are equal to those estimated using the UT.[1,6] The variety of UT formulations differ in how they calculate the sigma points and their respective weights. Depending on *a priori* assumptions about the distribution's spread after transformation, it may be desirable to introduce parameters that adjust the scaling of $\mathbf{A}_i$ to reflect an expansion or contraction of the sigma points' distribution around the mean.[6] Other algorithms[10] use more or fewer sigma points than $2L + 1$, and this may be helpful with different computational limitations and requirements.

The bias of the UT is shown in Figure 7 and closely matches the bias of the true transformation. This is consistent with the UT's design, which aims to match the first moment of the nonlinear function.

COMPARISON OF CONVERTED-MEASUREMENT METHODS

Area of Uncertainty

The covariance structure for the Taylor and UT are similar to the covariance structure for the true error statistics. However, the area of uncertainty determined by the covariance matrix varies between the methods. Figure 8 shows the area of uncertainty for the methods presented. Notably, the UT and Taylor approximation are biased away from the mean of the true measurement.

Covariance containment of the measurements is critical to successful tracking. Figure 9 shows the areas of uncertainty as a function of range for the three methods and true statistics. (Varying θ did not affect the area of uncertainty for a fixed range value.) The figure shows that the area of uncertainty for the Taylor approximation and UT is less than the true statistics. This can make the tracker more susceptible to outliers and can lead to losing a track. On the other hand, the debiased area of uncertainty is larger than the true statistics. This can make the tracker more robust to outliers but can also lead to slow responsiveness to new measurements.

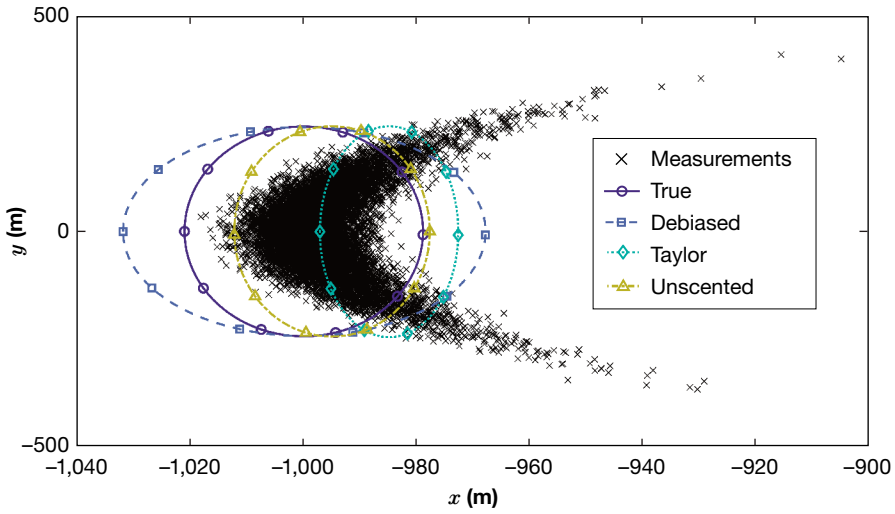

Figure 8. Areas of uncertainty of the covariance matrices for each of the four methods covered. The scattered points are the 10,000 measurements taken from a distribution of $r = 1{,}000$, $\theta = \pi$ and $\sigma_r = 5$, $\sigma_\theta = 0.1$. The four areas of uncertainties defined by a 95% confidence interval are shown around the mean value determined by the respective transformation method. Note the bias offsets from the UT and Taylor approximations away from the true mean.

Tracking Performance

We tested the debiased, Taylor approximation, EKF, and UT methods against the true statistics in a suite of tracking scenarios. A 2D simulated trajectory started at (100 km, 100 km) with an initial *x*-velocity of 50 m/s and *y*-velocity of 100 m/s, with a constant velocity dynamic model and $\sigma_m = 1$ according to Blackman and Popoli.[13] Simulated range and bearing measurements were generated from the ENU (east, north, up) coordinates with random noise of $\sigma_r = 25$ and $\sigma_\theta = 1°$. A total of 1,000 trajectories were simulated for 200 s. An example trajectory is shown in Figure 10. Perfect dynamic model matching and measurement association was assumed for these simulations. The initial track state for each method was initialized identically; the first two measurements were converted to ENU, and the difference in position over the difference in time was used for the initial velocity. The initial position covariance was set to the position covariance of the second measurement, and the velocity covariance was calculated using $(\mathbf{P}_1 + \mathbf{P}_2)/(\Delta t)^2$, where $\mathbf{P}_1$ and $\mathbf{P}_2$ are the converted RBE to ENU measurement covariances using the Taylor approximation.

The simulated data were tracked using the equations specified in this article via a built-in Kalman filter in MATLAB, but the measurement means and covariances were adjusted according to the converted-measurement

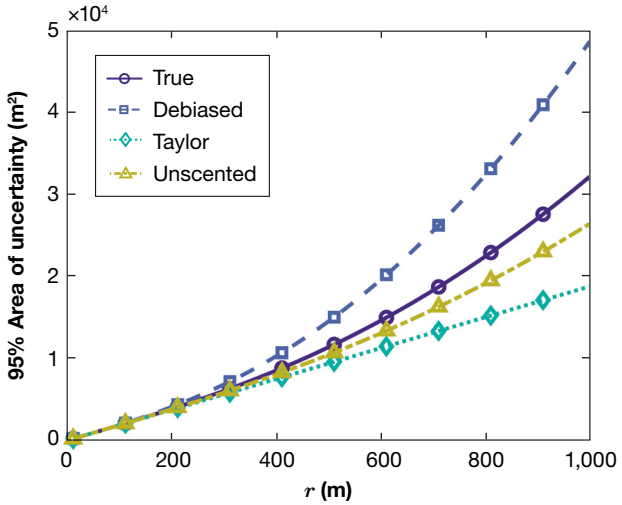

Figure 9. Area of uncertainty as in Figure 8 plotted as a function of range for the four methods.

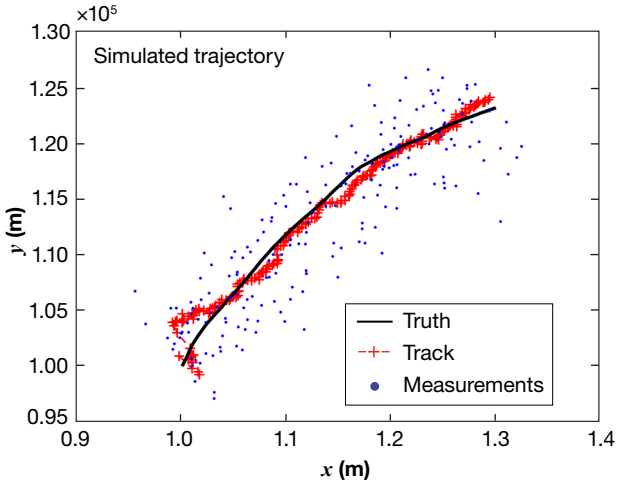

Figure 10. *xy* plot of the simulated trajectory in ENU coordinates. The track is moving farther away from the origin (0,0), toward the top right corner of the plot.

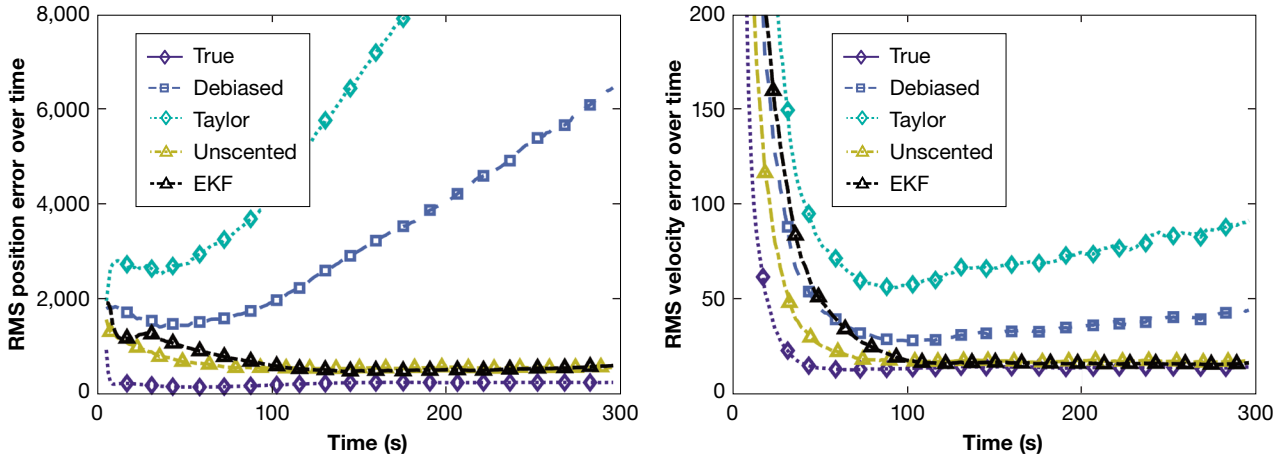

Figure 11. Average root mean square error for the different tracking methods as a function of time. Left, position errors; right, velocity errors.

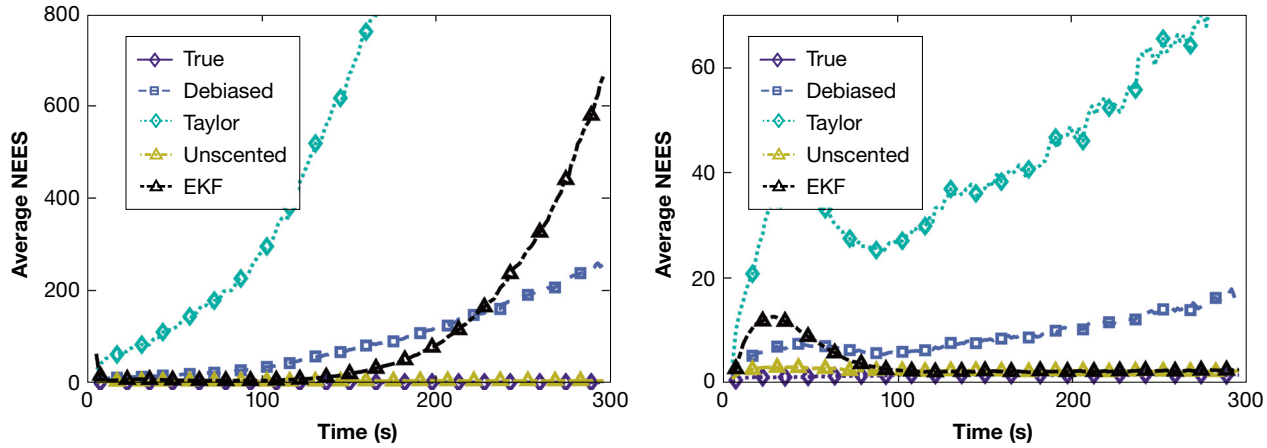

Figure 12. Average NEES for the different tracking methods as a function of time. Left, position NEES; right, velocity NEES.

method. For the Taylor approximation, measurements were converted using Eqs. 13 and 16, and Eq. 35 was used for the covariance of the approximation. Tracking performance was evaluated using the normalized estimation error squared calculation (NEES) and root mean square error. NEES was calculated for each time point and averaged across all time points and all runs. The results are shown in Figures 11 and 12.

DISCUSSION

Tracking using radar measurements is a challenging statistical problem because of the nonlinearities introduced by using spherical coordinate measurements for observations while using a Cartesian coordinate system for the dynamic model. The spherical coordinates can be retained in their original form to be tracked using EKF/UKFs and PFs in a mixed-coordinate regime or converted to Cartesian coordinates in a converted-measurement regime. Both approaches have their benefits. This article details three converted-measurement techniques to address the nonlinear tracking problem: debiased coordinate conversions, Taylor series approximation, and UT. Each of these methods converts the spherical coordinate measurements to Cartesian coordinates and attempts to describe the uncertainties after propagation through the coordinate transform. The performance of these methods was explored in a simple tracking scenario.

Converting range and bearing measurements introduced a bias in Cartesian coordinates; that is, the errors between the converted measurements and the true point in Cartesian coordinates were nonzero. When used in tracking, this violates the Kalman filter's assumption that the errors are zero-mean. All three methods exhibited nonzero errors (though the debiased measurements had much smaller biases) in a similar characteristic pattern, where the magnitude of Cartesian errors increased as a function of range. This makes sense, since errors in bearing are amplified at farther ranges. For example, a point travels a greater distance around a circle at a

farther radius for a given angle difference. For a given range, Cartesian errors varied sinusoidally as a function of bearing. The peak x error occurred at $\theta = \pi$, while the peak y error occurred at $\theta = \pi/2$. These values both corresponded to when $\cos\theta$ and $\sin\theta$ were at their respective maximums, thus amplifying their uncertainties when converting to Cartesian via $r\cos\theta$ and $r\sin\theta$. The errors of the UT and the true conversion were similar in magnitude, in agreement with the UT's matching of the first and second moments of the nonlinear function it is approximating.[1] Although the Taylor approximation matched the pattern, it had an overall larger magnitude of errors, suggesting the importance of the ignored higher-order terms.

The variances of the errors increased with range and followed a sinusoidal pattern. While only the variance patterns for the debiased conversion in Figure 4 are presented, both the Taylor approximation and the UT showed similar patterns. Contrary to what was briefly suggested in the original paper,[4] the difference between the true and debiased variances did not always strictly increase in the debiased case; instead, it varied sinusoidally. The individual debiased variances (each entry in the covariance matrix) were initially expected to be consistently greater, considering they incorporate the measurement uncertainties. However, the analysis revealed that the area of uncertainty was indeed larger in the debiased case across different ranges and bearings. The area of uncertainty as a function of ranges and bearings for each method was further investigated. Notably, varying the bearing while keeping the range constant did not alter the area of uncertainty. The debiased area of uncertainty was larger than the true uncertainty, whereas the UT and Taylor approximation yielded smaller areas. The relative size of the uncertainty area, larger or smaller compared with the true distribution, is not inherently advantageous or disadvantageous; it depends on the specific application. A larger uncertainty area makes the filter less responsive to accurate measurements but more resilient to outliers. Qualitatively, the UT most closely matched the true distribution of uncertainty, as illustrated in Figure 9.

The conversion methods' performance was tested in the simulated tracking scenario. Notably, the EKF and the UT performed best. Despite the debiasing procedure, the debiased converted-measurement method suffered at farther ranges in RMS error. This is potentially due to the Gaussian assumption of the noise failing at longer ranges. The UT can better match the true covariance, which may be more important than correcting for the bias. There was no significant improvement in combining the UT with the debiased measurements. This may suggest that improvements in NEES and RMS error using the UT are due to the more accurate covariance estimation, since the UT retains the bias in the nonlinear conversion as a result of the moment matching. Interestingly, the position NEES for the debiased measurements outperformed the EKF. This was due to an observed bias in the EKF at further ranges. We emphasize that the improvement in error using the EKF over the debiased method is geometry dependent. The nonlinearity using the EKF suffers at closer ranges, whereas the nonlinearity using the debiased method suffers at farther ranges. This is shown in the appendix. An interesting follow-up would be to find the particular range and angle error values where one method would benefit over the other.

CONCLUSION

Tracking using radar measurements can be difficult because of the nonlinear coordinate transfer between Cartesian and spherical coordinates. This article overviews radar tracking methods that convert the spherical radar measurements to Cartesian coordinates to use in a linear tracking framework. Three methods were used to propagate the statistical uncertainties into the Cartesian space: debiased coordinate conversion, Taylor approximation, and UT. The UT outperformed the other methods in a particular tracking scenario, while the EKF and debiased coordinate conversion demonstrated varying performances as a function of geometry.

REFERENCES

[1] D. Simon, *Optimal State Estimation: Kalman, H Infinity, and Nonlinear Approaches*, John Wiley & Sons, 2006.

[2] M. A. Richards, *Fundamentals of Radar Signal Processing*, 1st ed. McGraw-Hill, 2005.

[3] A. Gelb (Ed.), *Applied Optimal Estimation*, MIT Press, 1974.

[4] D. Lerro and Y. Bar-Shalom, "Tracking with debiased consistent converted measurements versus EKF," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, no. 3, pp. 1015–1022, 1993, https://doi.org/10.1016/S1474-6670(17)48288-7.

[5] A. Doucet and A. M. Johansen, "A tutorial on particle filtering and smoothing: Fifteen years later," *Handbook of Nonlinear Filtering*, vol. 12, no. 656–704, p. 3, 2009.

[6] E. A. Wan and R. Van Der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proc. IEEE 2000 Adapt. Syst. Signal Process. Commun. Control Symp.*, 2000, pp. 153–158, https://doi.org/10.1109/ASSPCC.2000.882463..

[7] M. Malanowski and K. Kulpa, "Two methods for target localization in multistatic passive radar," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 48, no. 1, pp. 572–580, 2012, https://doi.org/10.1109/TAES.2012.6129656.

[8] H. Marom, Y. Bar-Shalom, and B. Milgrom, "Unbiased conversion of 3-D bistatic radar measurements to Cartesian position," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 59, no. 2, pp. 1613–1623, 2023, https://doi.org/10.1109/TAES.2022.3203958.

[9] G. A. Terejanu et al., "Extended Kalman filter tutorial," University at Buffalo, p. 27, 2008.

[10] S. J. Julier and J. K. Uhlmann, "Unscented filtering and nonlinear estimation," *Proc. IEEE*, vol. 92, no. 3, pp. 401–422, 2004, https://doi.org/10.1109/JPROC.2003.823141.

[11] F. Daum and R. Fitzgerald, "Decoupled Kalman filters for phased array radar tracking," *IEEE Trans. Automat. Control*, vol. 28, no. 3, pp. 269–283, 1983, https://doi.org/10.1109/TAC.1983.1103242.

[12] B. Ghojogh, F. Karray, and M. Crowley, "Eigenvalue and generalized eigenvalue problems: Tutorial," *arXiv*, submitted Mar 25. 2019 (v1), last revised May 20 2023 (v3), https://doi.org/10.48550/arXiv.1903.11240.

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

APPENDIX. TRACKING SCENARIO AT CLOSER RANGE

The initial position was set to (1 km, 1 km) and σ_θ was set to 0.1°. Results show that the EKF performed worse than the other methods (Figure 13).

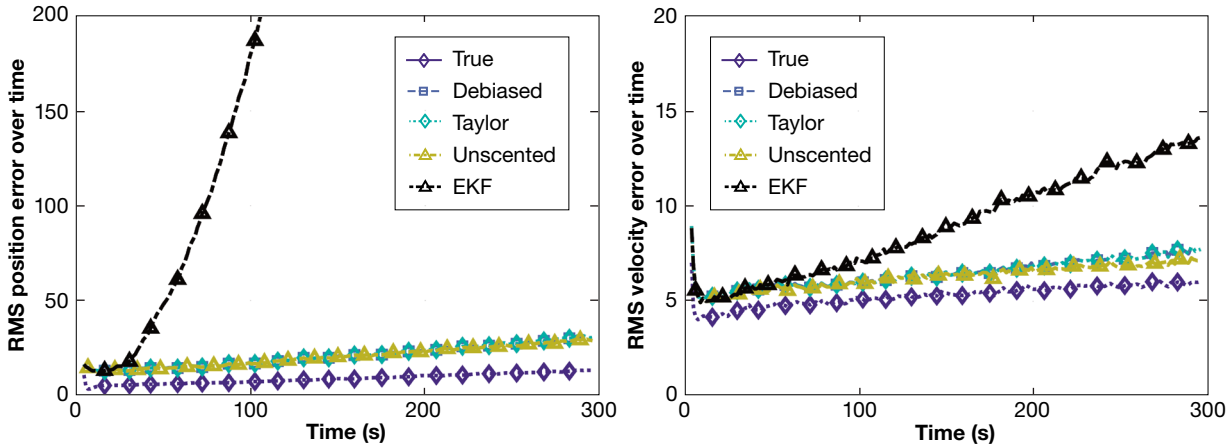

Figure 13. Average RMS error for the tracking methods as a function of time. Left, position RMS error; right, velocity RMS error.

Alexander J. Pei, Air and Missile Defense Sector, Johns Hopkins University Applied Physics Laboratory, Laurel, MD

Alexander J. Pei is a biomedical and electrical and computer engineer in APL's Air and Missile Defense Sector. He has a BS in biomedical engineering from Boston University and an MS in electrical and computer engineering from Carnegie Mellon University. After initially working in neuroscience, machine learning, dynamical systems, and signal processing, he transitioned to working on radar and tracking. His email address is alexander.pei@jhuapl.edu.

JOHNS HOPKINS
APPLIED PHYSICS LABORATORY

www.jhuapl.edu/technical-digest