Naval Research Laboratory
Washington, DC 20375-5320

NRL/5340/MR—2023/1

Process Noise Selection Based on Mismatched Filter Design

DAVID FREDERIC CROUSE

Surveillance Technology Branch
Radar Division

March 2, 2023

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
02-03-2023	NRL Memorandum Report	

4. TITLE AND SUBTITLE

Process Noise Selection Based on Mismatched Filter Design

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
63890C

6. AUTHOR(S)

David Frederic Crouse

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER
1V76

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

Naval Research Laboratory
4555 Overlook Avenue, SW
Washington, DC 20375-5320

8. PERFORMING ORGANIZATION REPORT NUMBER

NRL/5340/MR--2023/1

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

U.S. Missile Defense Agency
5700 18th St #245
Fort Belvoir, VA 22060

10. SPONSOR / MONITOR'S ACRONYM(S)

MDA

11. SPONSOR / MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A: Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This report derives a explicit recursions for the MSE and bias of both predicted and posterior estimates of a constant gain discrete-time Kalman filter when run on an arbitrary mismatched model. For constant maneuvers, asymptotic solutions are also provided. These solutions are then used to optimize over process noise parameters for Kalman filtering to either minimize the MSE of a mismatched model or such that the maximum error of a mismatched model equals the measurement MSE. As compared to similar solutions in previous work, this formulation of the problem allows for explicit solutions to be obtained in certain scenarios and also to optimize over more difficult trajectory types than are traditionally considered. A number of examples, including finding the optimal covariance scale factor for an evasive weaving target in 2D, are considered.

15. SUBJECT TERMS

Process noise Filtering Target tracking
Performance estimation Bias estimation

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					David F. Crouse
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	U	23	**19b. TELEPHONE NUMBER** *(include area code)*
U	U	U			(202) 404-1859

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

This page intentionally left blank.

Process Noise Selection Based on Mismatched Filter Design

David Frederic Crouse

I. Introduction

WHEN performing any type of state estimation using discrete-time Kalman filters, the question of how to parameterize the process noise covariance arises. This work focusses on how to set the scale factor of the process noise covariance matrix in the linear Kalman filter when faced with an arbitrary dynamic model. Choosing a reasonable scaling of the covariance matrix in the Kalman filter model is important. If the scale factor is too high, then one will not do better than "connecting the dots" between the measurements. Similarly, if the scale factor is too low, then the performance can be notably worse than just "connecting the dots" between the measurements, possibly leading to filter divergence. Incorrectly scaling the process noise covariance matrix gave Kalman filtering a bad reputation for eventually ignoring measurements and diverging when first used in orbital dynamics applications years ago, a problem deemed "smugness" [43, Ch. 10.6].

The dynamic model used in a linear discrete-time Kalman filter is

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

where $\mathbf{x}_k$ is the $d_z \times 1$ dimensional target state at discrete time k and $\mathbf{x}_{k+1}$ is the $d_z \times 1$ dimensional state at time $k+1$. The $d_x \times d_x$ dimensional state transition matrix at time k is given by $\mathbf{F}_k$ and the $d_x \times 1$ process noise $\mathbf{v}_k$ is taken to be a zero-mean Gaussian random variable with covariance matrix $\mathbf{Q}_k$. It is assumed that $\mathbf{v}_k$ and $\mathbf{v}_j$ for $k \neq j$ are independent. Many common dynamic models define $\mathbf{F}_k$ exactly, conditioned on knowing the continuous time interval T_k between discrete steps k and $k+1$. However, while $\mathbf{Q}_k$ typically also depends on T_k, it is generally only known within a scale factor. This report looks at how, given a particular model, one can choose the best $\mathbf{Q}_k$ according to some criterion given simple linear measurements.

Commonly, simple "rule of thumb" methods of parameterizing the process noise term based on roughly setting the scale factor term of the process noise covariance matrix $\mathbf{Q}_k$ based on a maximum assumed velocity, acceleration, jerk, or similar moment, as described by the rules given in [4, Ch. 6.2, 6.3]. However, better methods exist. For example, in [9], Blair presents a systematic procedure for selecting the optimal process noise term in nearly-constant velocity (NCV) models. It creates a mean-squared position error cost function based on a sensor noise-only covariance matrix and position and velocity lag terms expressed in terms of the performance of an α-β filter.[1] Basically, it assumes that one is running a filter designed according to a particular discrete-time model, but that in the worst-case scenario, the target is performing a sustained maneuver (with a known maximum acceleration), meaning that there is a mismatch between the filter and the model, the error of which can be characterized. The use of a fixed-gain (α-β) model is because the covariance matrix (and thus the gain) of a Kalman filter asymptotically converges to a constant regardless of the model mismatch. Thus, the design is for a filter that has been running long enough to settle. Two optimality criteria are considered:

- Choose the scale factor for $\mathbf{Q}_k$ to minimize the asymptotic (as in for a prolonged maneuver) mean-squared position error of the filter. This forms an upper bound.
- Choose the scale factor for $\mathbf{Q}_k$ such that the asymptotic mean squared position is no worse than the variance of the measurement. That is, no worse than "connecting the dots." This forms a lower bound.

The two optimality criteria lead to upper and lower bounds for the scale factor of $\mathbf{Q}_k$, which are given in terms of interpolation formula due to the difficulty in obtaining explicit solutions. Rather than being parameterized directly in terms of a maximum acceleration and a prediction time, the main interpolation portion of the algorithms is parameterized in terms of a deterministic extension of the "tracking index" of [33].

This general approach to process noise design of [9] has been summarized in [16] and has been extended over the years. Whereas [9] focused on when sustained (long) maneuvers are expected, a similar design methodology for brief maneuvers is addressed in [10]. Indeed, for brief maneuvers, the two optimality criteria above are just slightly modified to be

- Choose the scale factor for $\mathbf{Q}_k$ to minimize the *peak* mean-squared position error of the filter of a maneuver of an n discrete step maneuver. This forms an upper bound.
- Choose the scale factor for $\mathbf{Q}_k$ such that the *peak* mean squared position is no worse than the variance of the measurement. That is, no worse than "connecting the dots." This forms a lower bound.

The design methodology is extended to nonlinear measurements in [11]. A combination of many of these concepts is given in [18]. In [13], the specific case of handling range-Doppler coupling given linearly frequency modulated (LFM) measurements

[1] Derivations for some of the key terms in [9] come from [7] (particularly the appendices), which is given in an abbreviated form in [8].

Manuscript approved February 28, 2023.

is addressed. The topic of when the upper and lower bounds for the process noise according to the two criteria mentioned above meet is discussed in the context of the optimal positioning of mobile sensors in [21]. Dynamic models besides NCV models have also been considered. In [12] and [14], filters with exponentially-correlated acceleration errors (ECAE) are considered. Other than being one order higher than NCV models, these have an additional correlation time constant. In [12] and [14], the focus was on setting the scale factor for $\mathbf{Q}_k$ for a fixed correlation time constant. In [15] and [17], nearly constant acceleration (NCA) models are considered.

The general framework thus described is not the only approach that has been taken to process noise selection for filters. For example, in [41], very complicated expressions for choosing the gain of fixed-gain filters when tracking 1D sinusoidal motion are presented. Directly producing a gain bypasses the process noise model entirely. However, those expressions are based on ensemble averages, do not minimize the peak MSE of the filter, and can lead to peak MSE values that exceed the measurement error. The consideration of spiraling targets can be important as they have posed problems for systems in the first Gulf War [34], though as noted in [39], many lessons were learned from the challenges faced back then. In [23], it is suggested that the process noise of a Kalman filter be chosen based on assumptions regarding model mismatch and expressions for the mean squared error of a mismatched filter. However, it is not the same as [9], because it does not consider deterministic maneuvers. In [42], a formula is provided that provides Kalman filter parameters to obtain the same performance as if one were using a so-called "reduced state estimator" as described in [37]. The basic reduced state estimator is parameterized in terms of maximum values related to target maneuvers. For example, in terms of maximum accelerations. The algorithm of [42] allows one to parameterize a Kalman filter that has the same performance as a reduced state estimator, specifically when using a discrete white noise acceleration (DWNA) dynamic model.

Another approach to solving the design problem is to try to adaptively estimate parameters of the process noise covariance matrix. Some approaches to that are described in [36], [28], [29], and [2], but are probably of limited utility in radar and sonar tracking applications. Another approach is a two-stage filter, where one filter (with fixed parameters) tries to adaptively adjust the parameters of a second filter [7], [8]. That unto itself is reminiscent of the strong tracking filter, which is designed with a rather low process noise and adaptively adjusts the process noise in its measurement update step to prevent significant filter divergence [25].

The approach taken in this report is to rederive the basic method of [9] in a more versatile manner that lends itself to solving more general problems and to sometimes obtaining explicit solutions where only approximations were hitherto possible. After some basic models and assumptions are reviewed in Section II, Section III describes the key tool used in this report to determine "good" process noise scale terms: expressions for the MSE and bias of the posterior estimate of the Kalman filter when run with a mismatched model. In addition to deriving a recursive formulation, asymptotic expressions under a constant additive maneuver are also provided in Section III. The same expressions for the bias and MSE of the predicted estimate in a mismatched Kalman filter are also provided. Though expressions for the MSE performance with a mismatched model are given in [19], those are not as versatile and require speific derivations of terms for different state dimensionalities and dynamic models. Beyond setting the process noise covariance terms, such performance prediction as in Section III can be useful for systems level analysis of defense architectures as discussed in [30].

Given the new expressions for the MSE and bias of mismatched filter Section IV describes the optimization approaches for determining optimal process noise scale terms. It is really just an application of the methods employed by Blair to obtain his interpolatory formulas, but applied more generally in light of the availability of explicit expressions for the MSE. The solutions are derived assuming a constant gain. Once a Kalman filter with a uniform revisit rate has been running for a while, the gain converges to a constant. Section V discusses the direct computation of such a gain.

To see the potential of the optimization approach given the new explicit MSE expressions, Section VI reviews a number of common linear dynamic models for the Kalman filter. Some of these are used in Appendix A to demonstrate that the new formulation can lead to explicit solutions for some models where no explicit solutions were previously possible. However, the main result is in simulations in Section VII, which demonstrate the filter. Notably the simulation section specifically includes the optimization of the coefficients for a 2D weaving dynamic model and is shown to be superior to past approximations. Thus, while previous work mostly focused on 1D optimization, the new approach in this report can be used for arbitrary highly nonlinear models in 2D or 3D models. This opens the door to optimizing the process noise terms over actual trajectories rather than simplifies assumptions of constant maneuvers of a finite duration. The results are then concluded in Section VIII.

II. Basic Assumptions and Models

Though in some instances, the derivations of this report can apply to nonuniform sampling periods when one adjusts $\mathbf{F}_k$ and $\mathbf{Q}_k$ at each step, when dealing with the mean squared error derivations, it is assumed that the sampling interval is uniform, so the discrete dynamic model (1) just becomes

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{v}_k \tag{2}$$

where the subscript k has been removed from the state transition matrix $\mathbf{F}$ and similarly, we assume that the covariance matrix of the additive Gaussian noise is $\mathbf{Q}$, that is, it does not depend on k. The time interval between steps is taken to be a constant T. The mismatched filter equations are derived for a general linear measurement model of

$$\hat{\mathbf{z}}_k = \mathbf{H}\mathbf{x}_k + \mathbf{w}_k \tag{3}$$

where $\hat{\mathbf{z}}_k$ is the $d_z \times 1$ noisy measurement vector at time k, $\mathbf{H}$ is the $d_z \times d_x$ measurement matrix, and $\mathbf{w}_k$ is a $d_z \times 1$ zero-mean Gaussian noise vector with $d_z \times d_z$ covariance matrix $\mathbf{R}$. It is assumed that all $\mathbf{w}_k$ and $\mathbf{w}_j$ are independent for $k \neq j$ and that all $\mathbf{w}_k$ and $\mathbf{v}_j$ are independent. In all of the simulations, it is assumed that the $\mathbf{H}$ matrix just returns the position components of the target state.

The finite step and asymptotic mismatched Kalman filter expressions of Section IV are valid for the case where the Kalman filter is run assuming a stochastic dynamic model as in (2), but actually, the dynamic model is a "worst-case" maneuver of the form

$$\mathbf{x}_{k+1} = \mathbf{F}\mathbf{x}_k + \mathbf{a}_k \tag{4}$$

For asymptotic solutions, we take $\mathbf{a}_k = \mathbf{a}$ to be a constant vector. Though the bias formulation in Section III is correct for any general $\mathbf{a}$, $\mathbf{H}$, $\mathbf{F}$, $\mathbf{Q}$, and $\mathbf{R}$ values, sometimes, particularly in the explicit solutions in Appendix A, the focus is on 1-dimensional dynamic models, as was done with the constant bias model in [9] and other papers. Many common dynamic models, such as those given in [4, Ch. 6], or [40, Ch.3.2.1], are provided in 1 dimension, as say $\mathbf{F}_{1\text{D}}$ and $\mathbf{Q}_{1\text{D}}$, and they are generalized to $d > 1$ dimensions as

$$\mathbf{F} = \mathbf{F}_{1\text{D}} \otimes \mathbf{I} \tag{5}$$

$$\mathbf{Q} = \mathbf{Q}_{1\text{D}} \otimes \mathbf{I} \tag{6}$$

where $\mathbf{I}$ is a d-dimensional identity matrix and $\otimes$ is a Kronecker product (for example, the `kron` function in Matlab). Consequently, an algorithm designed for a single dimension can be applied to multiple dimensions. A motivation for focusing on the 1D case is that the cross terms in the cross terms in $\mathbf{a}\mathbf{a}'$, which appears in Section IV can be direction dependent. That dependence is avoided in 1D.

While the derivations in terms of $\mathbf{a}$ are fairly generic, two specific choices of $\mathbf{a}$ are considered most of the time for the solutions discussed in this report. Except for the weaving dynamic model used in the simulations of Section VII, the examples considered are filters where $\mathbf{x}_k$ consists of 1D position followed by $d_x - 1$ derivatives of position. Thus, the first form of $\mathbf{a}$ chosen reflects the effect of a constant maneuver of value A applied to one moment higher than the highest moment of the state. That is specifically,

$$\mathbf{a} = \begin{bmatrix} \displaystyle\int_0^T \int_0^{t_1} \int_0^{t_2} \cdots \int_0^{t_{d_x-1}} A \, dt_{d_x} \ldots dt \ldots dt_3 \, dt_2 \, dt_1 \\ \vdots \\ \displaystyle\int_0^T \int_0^{t_1} \int_0^{t_2} A \, dt_3 \, dt_2 \, dt_1 \\ \displaystyle\int_0^T \int_0^{t_1} A \, dt_2 \, dt_1 \\ \displaystyle\int_0^T A \, dt_1 \end{bmatrix} = \begin{bmatrix} \dfrac{AT^{d_x}}{d_x!} \\ \vdots \\ \dfrac{AT^3}{6} \\ \dfrac{AT^2}{2} \\ AT \end{bmatrix} \tag{7}$$

where A is the value applied to the $(d_x + 1)$th moment, which will typically be taken as a maximum acceleration, jerk, etc. that one expects the target might maneuver with. That is the type of maneuver originally considered in [9].

On the other hand, when considering nearly constant acceleration (NCA) models as in [15] and [17], second order models for 1D motion are considered. However, the maneuver itself is an acceleration. Thus, the general model considers a non-maneuvering model as not accelerating. Then, a single step application of an acceleration $\mathbf{a}$ that changes the state from having a zero acceleration to an acceleration of A is applied. After that, $\mathbf{a} = \mathbf{0}$ and the filter error will eventually peak and then settle.

For the general idea of replacing the final moment of the state with a value of A, the $\mathbf{a}$ vector is

$$\mathbf{a} = \begin{bmatrix} \int_0^T \int_0^{t_1} \int_0^{t_2} \cdots \int_0^{t_{d_x-2}} A \, dt_{d_x} \ldots dt \ldots dt_3 \, dt_2 \, dt_1 \\ \vdots \\ \int_0^T \int_0^{t_1} \int_0^{t_2} A \, dt_3 \, dt_2 \, dt_1 \\ \int_0^T \int_0^{t_1} A \, dt_2 \, dt_1 \\ \int_0^T A \, dt_1 \end{bmatrix} = \begin{bmatrix} \frac{AT^{d_x-1}}{(d_x-1)!} \\ \vdots \\ \frac{AT^2}{2} \\ AT \\ A \end{bmatrix} \tag{8}$$

Finally, in a number of areas of the report, expressions are given in terms of the deterministic target maneuvering index. This is because many expressions in terms of it are invariant with respect to aspects of the scale of the problem. For example, the units used for time and distance. The target maneuvering index as originally defined by Kalata [33] is

$$\Gamma_D = \frac{AT^2}{\sqrt{R}} \tag{9}$$

when considering an target with maximum acceleraton A. Kalata demonstrated the applicability of the tracking index to first order models as well as to a second order direct discrete dynamic model, which is dubbed the discrete Wiener process acceleration (DWPA) model in [4, Ch. 6.3.3]. The perturbation of the DWPA model is an acceleration increment.

However, Appendix A demonstrates that when the constant maneuver term $\mathbf{a}$ is assumed to be one order higher than the order of the state as in (7), it makes sense to use a different definition of the target maneuvering index. When considering a second order model, where $\mathbf{a}$ in (7) represents the effects of a constant-jerk maneuver, which differs from the second order mode of [15], which uses a single step maneuver of (8), it makes sense to express the MSE matrix in terms of a second order tracking index defined as

$$\Gamma_{D,2} = \frac{AT^3}{\sqrt{R}} \tag{10}$$

III. The Bias and Mean-Squared Error of a Mismatched Filter With A Deterministic Input

A. The Posterior Performance

Expressions for a mismatched Kalman filter are given in [4, Ch. 5.6] and in [23]. However, neither formulation provides expressions that could be used to model an assumed stochastic dynamic model being mismatched to a deterministic model as in (4). When considering only deterministic maneuvers, the model in (4) is as general as possible. Specifically, if one wishes to consider a trajectory with states $\mathbf{x}_k$ and $\mathbf{x}_{k-1}$, then the deterministic input to go between the states is simply

$$\mathbf{a}_{k-1} = \mathbf{x}_k - \mathbf{F}\mathbf{x}_{k-1} \tag{11}$$

Thus, the recursive expressions developed in the section for the MSE matrix and the bias of a mismatched Kalman filter can be used with any linear dynamic model with linear (or Cartesian converted) measurements. This is more general than the performance prediction described for a single sensor in [19] and could presumably be extended to multiple sensors as in [19]. Such modeling can be useful for analyzing the effectiveness of tracking system architectures, as in [30].

We begin by assuming that the Kalman filter has been running for a while so that the gain in the filter has converged to a constant, as described in Section V. The constant gain is denoted $\mathbf{W}$. The Kalman filter update with a constant gain is:

$$\begin{aligned} \hat{\mathbf{x}}_{k|k} =& \hat{\mathbf{x}}_{k|k-1} + \mathbf{W}\left(\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}\right) && (12)\\ =& \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{W}\left(\mathbf{H}\left(\mathbf{F}\mathbf{x}_{k-1} + \mathbf{a}_{k-1}\right) + \mathbf{w}_k - \mathbf{H}\mathbf{F}\hat{\mathbf{x}}_{k-1|k-1}\right) && (13)\\ =& \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{W}\mathbf{H}\left(\mathbf{F}\mathbf{x}_{k-1} + \mathbf{a}_{k-1}\right) + \mathbf{W}\mathbf{w}_k - \mathbf{W}\mathbf{H}\mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} && (14)\\ =& \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{W}\mathbf{H}\mathbf{F}\mathbf{x}_{k-1} + \mathbf{W}\mathbf{H}\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k - \mathbf{W}\mathbf{H}\mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} && (15) \end{aligned}$$

Now, consider the error $\tilde{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k} - \mathbf{x}_k$:

$$\begin{aligned} \tilde{\mathbf{x}}_k =& \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{W}\mathbf{H}\mathbf{F}\mathbf{x}_{k-1} + \mathbf{W}\mathbf{H}\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k - \mathbf{W}\mathbf{H}\mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} - \left(\mathbf{F}\mathbf{x}_{k-1} + \mathbf{a}_{k-1}\right) && (16)\\ =& \mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} + \mathbf{W}\mathbf{H}\mathbf{F}\mathbf{x}_{k-1} + \mathbf{W}\mathbf{H}\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k - \mathbf{W}\mathbf{H}\mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} - \mathbf{F}\mathbf{x}_{k-1} - \mathbf{a}_{k-1} && (17)\\ =& \left(\mathbf{I} - \mathbf{W}\mathbf{H}\right)\mathbf{F}\hat{\mathbf{x}}_{k-1|k-1} - \left(\mathbf{I} - \mathbf{W}\mathbf{H}\right)\mathbf{F}\mathbf{x}_{k-1} - \left(\mathbf{I} - \mathbf{W}\mathbf{H}\right)\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k && (18)\\ =& \overbrace{\left(\mathbf{I} - \mathbf{W}\mathbf{H}\right)\mathbf{F}}^{\mathbf{A}}\tilde{\mathbf{x}}_{k-1} - \overbrace{\left(\mathbf{I} - \mathbf{W}\mathbf{H}\right)}^{\mathbf{B}}\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k && (19) \end{aligned}$$

$$=\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{B}\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k \tag{20}$$

The expected value of the error (the bias), $\mathbf{b}_k$ is

$$\mathbf{b}_k \triangleq \mathrm{E}\left\{\tilde{\mathbf{x}}_k\right\} \tag{21}$$
$$=\mathbf{A}\mathbf{b}_{k-1} - \mathbf{B}\mathbf{a}_{k-1} \tag{22}$$
$$=\mathbf{A}^k \,\mathrm{E}\left\{\tilde{\mathbf{x}}_0\right\} - \left(\sum_{n=0}^{k-1} \mathbf{A}^n\right)\mathbf{B}\mathbf{a}_n \tag{23}$$

Assuming that the filter does not diverge, and that the control input $\mathbf{a}_k = \mathbf{a}$ is a constant, the asymptotic bias is such that $\mathbf{b}_k = \mathbf{b}_{k-1}$ in (22) as $k \to \infty$. Consequently, one can solve for the asymptotic bias from

$$\left(\mathbf{A} - \mathbf{I}\right)\mathbf{b}_\infty = \mathbf{B}\mathbf{a} \tag{24}$$

to get

$$\mathbf{b}_\infty = \left(\mathbf{A} - \mathbf{I}\right)^{-1}\mathbf{B}\mathbf{a} \tag{25}$$

The mean squared error (MSE) matrix at time k, $\mathbf{M}_k$ is thus the expected value of the outer product of $\tilde{\mathbf{x}}_k$, which is

$$\mathbf{M}_k \triangleq \mathrm{E}\left\{\tilde{\mathbf{x}}_k\tilde{\mathbf{x}}_k'\right\} \tag{26}$$
$$=\mathrm{E}\left\{\left(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{B}\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k\right)\left(\mathbf{A}\tilde{\mathbf{x}}_{k-1} - \mathbf{B}\mathbf{a}_{k-1} + \mathbf{W}\mathbf{w}_k\right)'\right\} \tag{27}$$
$$=\mathbf{A}\,\mathrm{E}\left\{\tilde{\mathbf{x}}_{k-1}\tilde{\mathbf{x}}_{k-1}'\right\}\mathbf{A}' + \mathbf{B}\mathbf{a}_{k-1}\mathbf{a}_{k-1}'\mathbf{B}' + \mathbf{W}\,\mathrm{E}\left\{\mathbf{w}_k\mathbf{w}_k'\right\}\mathbf{W}' - \mathbf{A}\,\mathrm{E}\left\{\tilde{\mathbf{x}}_{k-1}\right\}\mathbf{a}_{k-1}'\mathbf{B}' - \mathbf{B}\mathbf{a}_{k-1}\,\mathrm{E}\left\{\tilde{\mathbf{x}}_{k-1}\right\}'\mathbf{A}' \tag{28}$$
$$=\mathbf{A}\mathbf{M}_{k-1}\mathbf{A}' + \mathbf{B}\mathbf{a}_{k-1}\mathbf{a}_{k-1}'\mathbf{B}' + \mathbf{W}\mathbf{R}\mathbf{W}' - \mathbf{A}\mathbf{b}_{k-1}\mathbf{a}_{k-1}'\mathbf{B}' - \mathbf{B}\mathbf{a}_{k-1}\mathbf{b}_{k-1}'\mathbf{A}' \tag{29}$$
$$=\overbrace{\mathbf{A}\mathbf{M}_{k-1}\mathbf{A}' + \mathbf{W}\mathbf{R}\mathbf{W}'}^{\text{Maneuver-Free Contribution}} + \overbrace{\left(\mathbf{A}\mathbf{b}_{k-1} - \mathbf{B}\mathbf{a}_{k-1}\right)\left(\mathbf{A}\mathbf{b}_{k-1} - \mathbf{B}\mathbf{a}_{k-1}\right)' - \mathbf{A}\mathbf{b}_{k-1}\mathbf{b}_{k-1}'\mathbf{A}'}^{\text{Maneuver Contribution}} \tag{30}$$

This, the MSE depends recursively both on the previous MSE as well as the previous bias.

Asymptotically, if the MSE converges, then $\mathbf{M}_k = \mathbf{M}_{k-1}$ as $k \to \infty$ and one can substitute in the asymptotic expression for $\mathbf{b}_\infty$ to obtain the matrix relation

$$\mathbf{M}_\infty = \mathbf{A}\mathbf{M}_\infty\mathbf{A}' + \mathbf{B}\mathbf{a}\mathbf{a}'\mathbf{B}' + \mathbf{W}\mathbf{R}\mathbf{W}' - \mathbf{A}\left(\mathbf{A} - \mathbf{I}\right)^{-1}\mathbf{B}\mathbf{a}\mathbf{a}'\mathbf{B}' - \mathbf{B}\mathbf{a}\mathbf{a}'\mathbf{B}'\left(\left(\mathbf{A} - \mathbf{I}\right)^{-1}\right)'\mathbf{A}' \tag{31}$$

which is a discrete Lyapunov equation in terms of $\mathbf{M}_\infty$. Note that if one is optimizing only over a dynamic model with 1D position, so the element in the first row and column of $\mathbf{M}_\infty$ is the position MSE, then when considering maneuvers as in (7), the terms involving A, which is in $\mathbf{a}$ in (7) and (8) in the position MSE, will typically be squared, so only the magnitude, not the direction of acceleration matters.

Equation (31) can be rewritten as a Sylvester equation as

$$\mathbf{A}^{-1}\mathbf{M}_\infty - \mathbf{M}_\infty\mathbf{A}' = \mathbf{A}^{-1}\left(\mathbf{B}\mathbf{a}\mathbf{a}'\mathbf{B}' + \mathbf{W}\mathbf{R}\mathbf{W}' - \mathbf{A}\left(\mathbf{A} - \mathbf{I}\right)^{-1}\mathbf{B}\mathbf{a}\mathbf{a}'\mathbf{B}' - \mathbf{B}\mathbf{a}\mathbf{a}'\mathbf{B}'\left(\left(\mathbf{A} - \mathbf{I}\right)^{-1}\right)'\mathbf{A}'\right) \tag{32}$$

The Sylvester equation is a matrix equation of the form

$$\mathbf{C}_1\mathbf{X} + \mathbf{X}\mathbf{C}_2 = \mathbf{C}_3 \tag{33}$$

that one solves for $\mathbf{X}$, where $\mathbf{C}_1$, $\mathbf{C}_2$, and $\mathbf{C}_3$ are matrix constants. The Sylvester equation can be solved using the Bartels-Stewart algorithm of [5], where simple expressions for the solution after upper-triangulation of the matrices is given in [31, Ch. 7.6.3].

Note that certain degenerate cases might arise, though they were not observed in the examples given in Section VII. First, the above derivation assumes that neither $\mathbf{A}$ nor $\mathbf{A} - \mathbf{I}$ is singular. Additionally, as noted in [5], if $\lambda_{i,1}^C$ is the ith eigenvalue of $\mathbf{C}_1$ and $\lambda_{j,2}^C$ is the jth eigenvalue of $\mathbf{C}_2$, then a unique solution to the Sylvester equation exists only if there is no combination of i and j such that $\lambda_{i,1}^C + \lambda_{j,2}^C = 0$.

When solving (32), some steps can be taken to make the problem more scale invariant so as to reduce finite precision problems. First, in the case where $\mathbf{R}$ is a scalar R, dividing both sides of (32) by R is equivalent to replacing R with 1 and dividing $\mathbf{a}$ by $\sqrt{R}$. To undo the transformation, $\mathbf{M}_\infty$ should be multiplied by R and $\mathbf{b}_\infty$ by $\sqrt{R}$. Similar scaling can be applied to (30) and (25) to return to its original value. Additionally, when optimizing over a multiplicative coefficient q in $\mathbf{Q}$, such scaling means that the final coefficient found should be multiplied by R. Similarly, whereby $\mathbf{b}$ will need to be scaled by $\sqrt{R}$.

Another thing that can be done to reduce the susceptibility to finite precision errors is to change the units of the time interval T so that the transformed T is 1. This means that $\mathbf{F}$ and the $\mathbf{Q}_{\text{unscaled}}$ will be computed with $T = 1$. Additionally, $\mathbf{a}$ in (7), which is a deterministic input that is one order higher than the state, is computed with $T = 1$ and A replaced with AT^{d_x}. On the other hand, if one is using (8), which is a deterministic input that is the same order as the state, then A should be replaced

with AT^{d_x-1}.

Once one obtains the scaling constant for $T = 1$, q_1, the time units need to be transformed so that T is its original value. The transformation depends specifically on the units of q because, as noted in Section VI, different dynamic models have different units for q. For example, q in a discretized continuous-time linear dynamic model of polynomial order d has units of length^2/time$^{2\text{ order}+1}$. Consequently, in such a model, one transforms q_1 to be in terms of a general T as

$$q_T = \frac{q_1}{T^{2\text{ order}+1}} \tag{34}$$

So a system consisting of position and velocity would divide by T^3 and one consisting of position, velocity and acceleration would divide by T^5. On the other hand, in a direct discrete linear dynamic model of polynomial order d, q has units of length^2/time$^{4\text{order}}$. Thus, the transformation from q_1 to q_T in such models would be

$$q_T = \frac{q_1}{T^{2(\text{ order}+1)}} \tag{35}$$

The final model considered in Section VI is the DWPA model, which is a second order model where q has units of length^2/time^4. Consequently,

$$q_T = \frac{q_1}{T^4} \tag{36}$$

B. The Predicted Performance

Whereas the previous subsection provided expressions for the performance of the posterior MSE matrix and bias vector, here expressions for the MSE and bias of the prediction are provided. This could be particularly useful for designing schedulers for tracking algorithms or for applications such as maximizing the prediction time T given that the minimum MSE scale factor q_T for that T, the MSE is less than or equal to a particular bound, such as the width of a radar beam. In this report, we provide expressions for the MSE matrix and bias of the predicted state, but we do not investigate its uses further.

When considering the prediction, (12) becomes

$$\begin{aligned}\hat{\mathbf{x}}_{k+1|k} =&\mathbf{F}\left(\hat{\mathbf{x}}_{k|k-1} + \mathbf{W}\left(\mathbf{z}_k - \hat{\mathbf{z}}_{k|k-1}\right)\right) &(37)\\ =&\mathbf{F}\left(\hat{\mathbf{x}}_{k|k-1} + \mathbf{W}\left(\mathbf{H}\mathbf{x}_k + \mathbf{w}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right)\right) &(38)\end{aligned}$$

Now, consider the error of the prediction, $\tilde{\tilde{\mathbf{x}}}_{k+1} = \hat{\mathbf{x}}_{k+1|k} - \mathbf{x}_{k+1}$, which we differentiate from the error of the posterior state by using an extra tilde:

$$\begin{aligned}\tilde{\tilde{\mathbf{x}}}_{k+1} =&\mathbf{F}\left(\hat{\mathbf{x}}_{k|k-1} + \mathbf{W}\left(\mathbf{H}\mathbf{x}_k + \mathbf{w}_k - \mathbf{H}\hat{\mathbf{x}}_{k|k-1}\right)\right) - \left(\mathbf{F}\mathbf{x}_k + \mathbf{a}_k\right) &(39)\\ =&\mathbf{F}\left(\mathbf{I} - \mathbf{W}\mathbf{H}\right)\tilde{\tilde{\mathbf{x}}}_k + \mathbf{F}\mathbf{W}\mathbf{w}_k - \mathbf{a}_k &(40)\end{aligned}$$

This can be rewritten in the same format as (20). That is,

$$\tilde{\tilde{\mathbf{x}}}_k = \tilde{\mathbf{A}}\tilde{\tilde{\mathbf{x}}}_{k-1} - \tilde{\mathbf{B}}\mathbf{a}_{k-1} + \tilde{\mathbf{W}}\mathbf{w}_{k-1} \tag{41}$$

where

$$\begin{aligned}\tilde{\mathbf{A}} =&\mathbf{F}\left(\mathbf{I} - \mathbf{W}\mathbf{H}\right) &(42)\\ \tilde{\mathbf{B}} =&\mathbf{I} &(43)\\ \tilde{\mathbf{W}} =&\mathbf{F}\mathbf{W} &(44)\end{aligned}$$

Consequently, by using $\tilde{\mathbf{A}}$, $\tilde{\mathbf{B}}$ and $\tilde{\mathbf{W}}$, instead of $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{W}$, in (30), (24), (32), and (25), one obtains expressions for the recursive predicted MSE, the recursive predicted bias, the asymptotic predicted MSE and the asymptotic predicted bias.

IV. A General Approach to Process Noise Scale Parameterization

When a target is not maneuvering, it is desirable to have the process noise scale term q be low so that once the filter settles, the error floor of the filter is low. At the same time, setting the process noise scale term too low can lead to the position MSE of the filter being worse than just connecting the dots between the measurements. Consequently, when tracking with a single dynamic model, the worst-case maneuver must be considered to ensure that the track error does not become so large the target is lost.

As in [9], [10], and other work, we consider two optimization approaches: Choosing the scale factor q to minimize the peak position MSE and choosing the scale factor such that the measurement MSE equals the position MSE of the filter, with the optimization performed over a worst-case maneuver. Minimizing the peak MSE provides a *maximum* gain and choosing the scale factor such that the position MSE of the filter equals the measurement MSE provides a *minimum* gain. Note that the MSE in $\mathbf{M}_k$ is the first element in 1D, and will be the sum of d diagonal elements in d-dimensional space.

Given those two optimization goals, two types of problems are considered

- A manuver that is a constant input $\mathbf{a}$, whereby we optimize over the *asymptotic* MSE of the filter.
- A maneuver of a finite duration, whereby the offset vector $\mathbf{a}_k$ will typically change over time. For maneuvers of a finite duration, the peak error can occur after the end of the maneuver.

The first problem type is not necessarily as useful. However, it is much more amenable to obtaining explicit solutions, as given in Appendix A. Thus, we begin by describing how to solve such the asymptotic problem.

When minimizing the asymptotic MSE over a scalar scale constant q or determining q such that the asymptotic MSE equals the variance of a position measurement (which is $\mathrm{trace}(\mathbf{R})$, a general approach for optimizing over a scalar scale constant q is

1) Assume a deterministic maneuver as in (4) with the maneuver offset $\mathbf{a}$ being a constant, such as in (7).
2) If solving numerically, determine bounds on the minimum and maximum values of q. This is generally going to be somewhat *ad-hoc*, but if one normalizes the models as described at the end of Section III, then one can typically set a very wide default range such as from 10^{-11} to 10^{8}. Note that setting the upper bound too high can lead to numerical issues when determining convergence of optimizations such that the MSE is the measurement MSE, because the cost functions often asymptotically approach the measurement MSE.
3) The final step just differs a little depending on the optimization desired. In each instance, the evaluation of the cost function for a particular scale factor q hypothesis requires solving for the asymptotic $\mathbf{W}$ given q as in Section V. The final steps are:
   - If choosing the MSE such that it equals the measurement MSE, then simply solve for q such that this is attained using the first element of the solution to (32). For numeric solutions, one can create *logarithmically* spaced points between the minimum and maximum bounds of q, such as using the `logspace` function in Matlab, and search for the first crossing the measurement MSE by the cost function, which is the solution to (32). Having found the first crossing, one can use the `BrentRootFind` function in the TCL, which implements [22], to find the point where the difference between the MSE and the measurement MSE is zero.
   - If minimizing the MSE, using the expression for the position MSE from the solution to the Sylvester equation of (32), minimize the position MSE over q. For all of the asymptotic models solved in this report, the cost function appears to be unimodal, so the minimum MSE can be found numerically by a golden section search. The algorithm is described in [6, Append. C2] and is implemented as the `goldenSectionSearch` function in the TCL. Matlab also has the built in function `fminbnd` which is related and can also be used. If the cost function isn't unimodal, or there are finite precision issues, one might want to perform a search over logarithmically spaced points in q. If q_i is point i among increasing values of q that has the lowest cost, then perform the golden section search with lower bound of q_{i-1} and upper bound of q_{i+1}.

For finite duration maneuvers, the approach is the same as in the asymptotic case, except one no longer assumes that $\mathbf{a}$ is constant and the cost function is no longer a solution to the Sylvester function of (32). The general approach to evaluating a cost function for a particular q hypothesis for a finite duration maneuver is

1) Initialize the MSE matrix at the first step of the problem by solving the Sylvester equation (32) with $\mathbf{a} = \mathbf{0}$ for the MSE matrix $\mathbf{M}$. This will require obtaining an asymptotic $\mathbf{W}$ gain matrix as in Section V. This will be used as the prior MSE matrix at the beginning of the maneuver. The associated asymptotic bias vector $\mathbf{b}$ is also needed and is zero. This assumes that in the absence of a maneuver, $\mathbf{F}$ accurately describes the evolution of the state and that the target hasn't been maneuvering prior to the onset of the worst-case maneuver.
2) Use (30) and (22) to propagate the MSE matrix and the bias for the duration of the maneuver and then, if the maneuver is assumed to end, for a fixed number of steps past the end of the maneuver. Typically, there will only be a single peak in the position MSE after the end of the maneuver, so that one can set a high maximum, such as 200 steps past the end of the maneuver, but can stop evaluation once the MSE starts going down. Over all of the points evaluated, find the maximum position MSE and that is the cost.

Thus, given the maximum position MSE, the same type of golden section search or Brent root finding optimization can be used to find the optimal solution once one has bounded the solution as in the procedure for asymptotic solutions.

When optimizing process noise terms for radar tracking, as was considered in [11], one can typically choose a different q, in a Cartesian coordinate system based on the range of the target (or sometimes multiple q values if one optimizes over different Cartesian direction separately). Though the methods described here are fairly brute-force, they are also fairly fast, given the explicit MSE expressions of Section III. That said, if one wishes to choose a different q based on the location from which the target is being predicted, it can be advantageous to tabulate and/or and interpolate the scale factor values ahead of time.

V. The Asymptotic Gain and Performance of Filters

The covariance matrix computed by a stable Kalman filter, regardless of whether it is correctly matched to the system it is tracking, asymptotically converges to a constant value when $\mathbf{H}$, $\mathbf{F}$, $\mathbf{R}$, and $\mathbf{Q}$ do not change with time. There are two

related Riccati equations. The prior equation solves for the asymptotic predicted covariance matrix $\mathbf{P}_{k|k-1}$. That is, the predicted covariance matrix before the filter updates with a measurement. The second Riccati equation solves for the asymptotic posterior covariance matrix $\mathbf{P}_{k|k}$. The prior Riccati equation is given by [4, Ch. 5.2.5]

$$\mathbf{P} = \mathbf{F}\mathbf{P}\mathbf{F}' - \mathbf{F}\mathbf{P}\mathbf{H}'\left(\mathbf{H}\mathbf{P}\mathbf{H}' + \mathbf{R}\right)^{-1})\mathbf{H}\mathbf{P}\mathbf{F}' + \mathbf{Q} \tag{45}$$

The posterior Riccati equation can be obtained by substituting the covariance prediction step of the Kalman filter into the measurement update step to get

$$\mathbf{P} = \mathbf{F}\mathbf{P}\mathbf{F}' + \mathbf{Q} - \left(\mathbf{F}\mathbf{P}\mathbf{F}'\mathbf{H}' + \mathbf{Q}\mathbf{H}'\right)\left(\mathbf{H}\mathbf{F}\mathbf{P}\mathbf{F}'\mathbf{H}' + \mathbf{H}\mathbf{Q}\mathbf{H}' + \mathbf{R}\right)^{-1}\left(\mathbf{F}\mathbf{P}\mathbf{F}'\mathbf{H}' + \mathbf{Q}\mathbf{H}'\right)' \tag{46}$$

For the optimizations in this report, the asymptotic Kalman filter gain $\mathbf{W}$ is needed. A simple expression of the gain at time k in terms of the posterior covariance matrix of the filter is

$$\mathbf{W}_k = \mathbf{P}_{k|k}\mathbf{H}'\mathbf{R}^{-1} \tag{47}$$

In this instance, we shall use the $\mathbf{P}$ that is the solution to the posterior Riccati equation of (46).

As noted in [25], these types of Riccati equations can be solved non-iteratively using the algorithm of [3]. An older algorithm that uses an eigenvalue decomposition and is presumably less stable also exists [44] and was used in [32] to find a direct solution for certain exponentially correlated dynamic model. The solution to the predicted Riccati equation is implemented in `RiccatiPredNoClutter` in the Tracker Component Library (TCL) [27], [1]. The solution to the posterior Riccati equation is implemented as `RiccatiPostNoClutter`. Both functions just parameterize the solution of [3], which is implemented in the `RiccatiSolveD` function in the TCL. The function `findAsymptoticGain` directly finds the asymptotic Kalman filter gain $\mathbf{W}$.

It should be noted that a type of Riccati equation to approximate the average asymptotic performance of a target tracking algorithm in the presence of missed detections exists [20] and is available in `RiccatiPredNoClutter` and `RiccatiPostNoClutter`. While this report focusses on choosing the optimal covariance matrix scale factor given that the target is assumed always detected, a Riccati solution as in [20] could be investigated in other algorithms for improving the performance given that the target might not always be detected.

VI. Dynamic Models and Some Associated Gain Expressions

A. Explicit Models

For dynamic models, we consider the discretized continuous-time linear dynamic model of a polynomial order d and the direct discrete linear dynamic model of polynomial order d. When considering the aforementioned direct discrete as well as the discretized models, an order d model has a state dimensionality of $d_x = d + 1$ when considering 1D motion. The $d_x \times d_x$ state transition matrix $\mathbf{F}$ for 1D motion for all of the models is such that if $F_{r,c}$ is the element in row r and column c of $\mathbf{F}$ with indexation starting at 0 and the point $r = 0$, $c = 0$, being in the upper-left of the matrix, then

$$F_{r,c} = \begin{cases} \dfrac{T^{c-r}}{(c-r)!} & \text{If } c - r \geq 0 \\ 0 & \text{Otherwise} \end{cases} \tag{48}$$

which is implemented as `FPolyKal` in the TCL [27], [1]. When considering a discretized continuous-time linear dynamic model of order d such that the noise process is assumed to be drive by a Wiener process in the $(d+1)$th moment, the element in row r and column c of the process noise covariance matrix is

$$Q_{r,c} = \frac{T^{(d-r)+(d-c)+1}}{(d-r)!(d-c)!((d-r)+(d-c)+1)}q \tag{49}$$

where q is the scale factor whose determination is the topic of this report. Equation (49) is implemented as `QPolyKal` in the TCL [1]. When considering an analogous direct discrete linear dynamic model, the process noise covariance matrix is singular and can be expressed as

$$\mathbf{Q} = \mathbf{g}\mathbf{g}'q \tag{50}$$

where the element in row r of the $d_x \times 1$ vector $\mathbf{g}$ is

$$g_r = \frac{T^{d-i+1}}{(d-i+1)!} \tag{51}$$

Equation (50) using (51) is implemented as `QPolyKalDirectDisc` in the TCL [1]. The expressions of (48) and (49) are an application of the discretization technique discussed in [4, Ch. 6.2], [46, Ch. 4], and [35, Ch. 9] to an arbitrary order model and were first given in [24]. The expression in (50) is an obvious generalization of the first order direct discrete model in [4, Ch. 6.3.2].

An alternative category of direct discrete models also exists. In this alternative formulation, the process noise is added directly to the highest order moment of the state (the dth order moment, rather than the $d+1$th order moment, which isn't part of the state). This alternative direct discrete model is the same form for $\mathbf{Q}$ as (50). However, element r of the $d_x \times 1$ vector $\mathbf{g}$ is changed to

$$g_r = \frac{T^{d-i}}{(d-i)!} \tag{52}$$

Under the model of (51), the lower-right entry in the matrix is T, whereas in the model of (52), it is 1. Equation (50) using (52) is implemented as `QPolyKalDirectAlt` in the TCL.

Note that the units of q for the discretized continuous-time models in (49) are length^2/time$^{2\text{order}+1}$ but the units of q in (50) using (51) or (52) are length^2/time$^{2\text{order}}$. These units are important to keep in mind when scaling models to reduce finite precision errors, as discussed in Section III when considering solving for q. If the units of T are changed going into the algorithm, then different dynamic models will require different scale factors to change the units of q back to their original values.

Though the methods in this report will work for arbitrary ordered models, this report only considers first and second order models. For first order models,

$$\mathbf{F}_{\text{First Order}} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \tag{53}$$

and for second order models

$$\mathbf{F}_{\text{Second Order}} = \begin{bmatrix} 1 & T & \frac{T^2}{2} \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \tag{54}$$

The first order discretized continuous-time linear dynamic model is called the discretized continuous white noise acceleration (DCWNA) model and the second order model is called the discretized continuous Wiener process acceleration model (DCWPA). The process noise covariance matrix of the first model is

$$\mathbf{Q}_{\text{DCWNA}} = \begin{bmatrix} \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^2}{2} & T \end{bmatrix} q \tag{55}$$

and the process noise covariance matrix of the second order model is

$$\mathbf{Q}_{\text{DCWPA}} = \begin{bmatrix} \frac{T^5}{20} & \frac{T^4}{8} & \frac{T^3}{6} \\ \frac{T^4}{8} & \frac{T^3}{3} & \frac{T^2}{2} \\ \frac{T^3}{6} & \frac{T^2}{2} & T \end{bmatrix} q \tag{56}$$

For direct discrete models based on the discretization of (51), the first order model, called the discrete white noise acceleration (DWNA) model is considered here. Its covariance matrix is

$$\mathbf{Q}_{\text{DWNA}} = \begin{bmatrix} \frac{T^4}{2} & \frac{T^3}{2} \\ \frac{T^3}{2} & T^2 \end{bmatrix} q \tag{57}$$

Additionally, we consider the discrete Wiener process acceleration model (DWPA), which is a second order model based on the alternate discretization in (52), which is given in [4, Ch. 6.3.3], and which is a direct discrete model. This model is considered simply to demonstrate that with appropriate relations between the gain terms, as given in Section VI-B, it is possible to get an explicit solution for the scale term q in a second order model. That said, one would probably not choose the DWPA

model. The process noise covariance matrix of the DWPA model is

$$\mathbf{Q}_{\text{DWPA}} = \begin{bmatrix} \frac{T^4}{4} & \frac{T^3}{2} & \frac{T^2}{2} \\ \frac{T^3}{2} & T^2 & T \\ \frac{T^2}{2} & T & 1 \end{bmatrix} q \tag{58}$$

B. Constant Filter Gain Relations

Though not required for general numeric solutions when minimizing the asymptotic mean squared error, the explicit solutions of Section A rely on having expressions between the elements of the asymptotic Kalman gain vector $\mathbf{W}$ and the process noise scale term q for 1D motion models. Parameterizing the gain vector as

$$\mathbf{W} = \begin{bmatrix} \alpha \\ \frac{\beta}{T} \end{bmatrix} \tag{59}$$

for a first order model and as

$$\mathbf{W} = \begin{bmatrix} \alpha \\ \frac{\beta}{T} \\ \frac{\gamma}{2T^2} \end{bmatrix} \tag{60}$$

for a second order model. A discussion of deriving some such relations is given in [4, Ch. 6.5]. Solutions for some direct discrete dynamic models are given in [33]. Section A will use the expressions in (59) and (60) as well as relations between α and β for first order models.

When considering, the DCWNA model, one can obtain the following relations from [4, Ch. 6.5.4]

$$\beta = 3(2-\alpha) - \sqrt{3(\alpha^2 - 12\alpha + 12)} \tag{61}$$

$$q = \frac{R\beta^2}{T^3(1-\alpha)} \tag{62}$$

where R is the scalar measurement variance. When considering the DWNA model [4, Ch. 6.5.3], the equivalent relations are

$$\beta = 2(2-\alpha) - 4\sqrt{1-\alpha} \tag{63}$$

$$q = \frac{R\beta^2}{T^4(1-\alpha)} \tag{64}$$

Similarly, when considering the DWPA model [4, Ch. 6.5.5]

$$\beta = 2(2-\alpha) - 4\sqrt{1-\alpha} \tag{65}$$

$$\gamma = \frac{\beta^2}{\alpha} \tag{66}$$

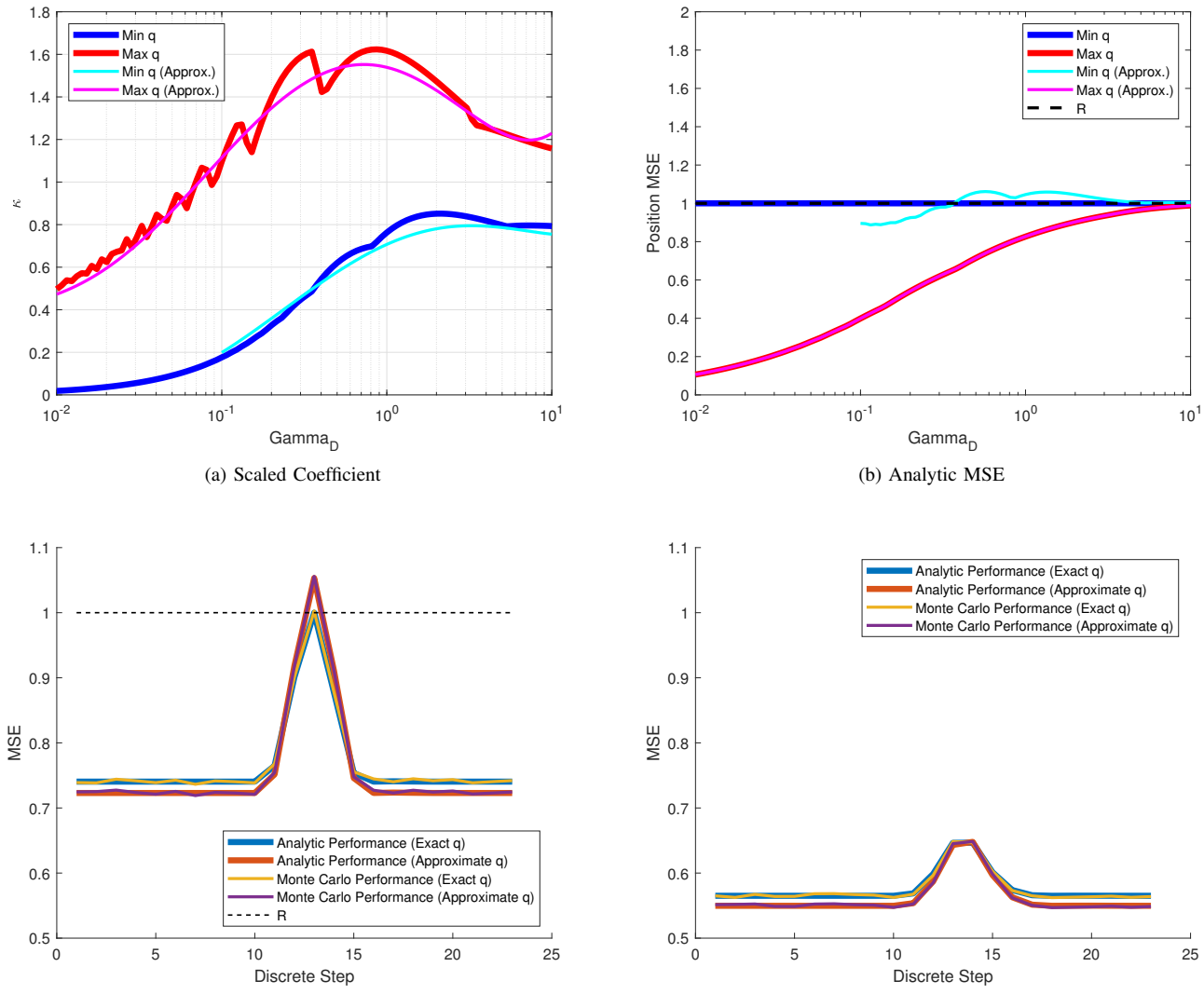
$$q = \frac{R\gamma^2}{4T^4(1-\alpha)} \tag{67}$$

VII. Examples and Simulations

To demonstrate the general approach here, we consider a number of examples. Section VII-A compares the solution obtained by the technique in this report for nearly constant velocity models given a constant acceleration maneuver to the explicit approximations in the literature. One can see good agreement over the tabulated ranges in the literature. However, when considering a finite state maneuver, one sees that the exact solutions are not as monotonic as one might expect (compared to smoothed solutions in the literature), presumably due to the discrete effect of the step of the maximum MSE in the filter changing.

Section VII-B then compares the performance of optimizing for the scale factor in NCA models using the direct discrete DWPA model, as has been done before, as well as when using the discretized continuous DCWPA dynamic model. Surprisingly, the results are nearly identical.

Finally, we apply the techniques discussed in this report to solving for the optimal gains for a weaving dynamic model in Section VII-C. The direct optimization over the actual weaving dynamic model is shown to be superior to optimization performed based only on a peak acceleration, as has been the case in previous work.



(a) Scaled Coefficient

(b) Analytic MSE

(c) Min Gain A and MSE

(d) Max Gain A and MSE

Fig. 1. In (a), exact and approximate, minimum (MSE=measurement MSE) and maximum (minimum MSE) process noise scale factors for the DCWNA model as a function of the maneuvering index are shown after being scaled into κ as in (68) for a 3 step maneuver. The MSE performance is shown in (b). Though the non-monotonicity of the exact solutions might be questionable, choosing points where they differ greatly $\Gamma_D = 1.71488$ for the minimum solution and 0.354446 for the maximum solution, one can see in (c) and (d) that the predicted performance is consistent with Monte Carlo simulations over $100,000$ runs and in (c), it is clear that the exact solution is the one that correctly equals the measurement MSE at the peak. The minimum gain approximation line covers a smaller region than the others to indicate that its region of accuracy is smaller than the others.

A. Finite Step Gain Relations with NCV Models

In this example, we consider an oddity of determining the optimal gains when considering a maneuver of a finite duration of time. Such optimization for the process noise scale factor was first considered for nearly constant velocity models in [10], where interpolatory coefficients for the optimal factors were given and we subsequently updated in [11]. and [18]. Here, we compare against the coefficients from [11].

We consider DCWNA model of Section VI, which is a nearly constant velocity model. For simplicity, so that all results are consistent with the scaled expressions in [10] and [11], instead of obtaining q, we obtain

$$\kappa = \frac{\sqrt{q}}{A} \tag{68}$$

where A is the peak acceleration, which means that the acceleration model of (7) is used. For simplicity in keeping the scaling consistent with plots in [10] and [11], the sample time $T = 1$ and the measurement variance $R = 1$. Considering finding the κ for the minimum scale factor, that is such that the peak MSE of the track equals the measurement variance, as well as the κ for the maximum scale factor, which minimizes the peak MSE, Figure 1a shows κ as a function of the maneuvering indexΓ_D. Both the solutions of this report as well as the approximations of [11] are shown. The solutions of this report are

not all monotonic in Γ_D, which presumably has to do with discrete changes in which step after the start of the maneuver has the peak error.

The correctness of the solutions is demonstrated by considering the performance of the filter as a function of step for a maneuvering problem which starts with the target not maneuvering and all filters initialized with the MSE obtained by solving (32) with $\mathbf{a} = 0$. That is, the performance where there hasn't been a maneuver for a long time. Then, from time step 10 to 11 a maneuver begins lasting 3 steps. After that, there is no maneuver again for the rest of the steps. The initial target state doesn't matter, because the disturbance is one degree higher than the state, using (7). $100,000$ Monte Carlo runs are performed and the analytic MSE solution from (30) is also considered.

When considering the minimum process noise scale factor solution, the point $\Gamma_D = 1.71488$ is chosen for the simulation because one can see a notable gap between the exact solution and the approximation in Fig. 1a. Similarly, for the maximum process noise scale factor solution, the point 0.354446 is chosen as there is again a noticeable gap.

It can be seen that the analytic performance agrees with the Monte Carlo simulations and in Fig. 1c the approximation overshoots the measurement MSE. In (1d), though not particularly visible, the optimal solutions have a marignally lower peak MSE, though noticeably higher MSE floor when not maneuvering. Consequently, it is interesting to see that solutions to terms related to q can be less smooth and monotonic as one might expect. However, existing approximations are good over their valid regions. Outside of those regions, numeric solutions should be preferred, or one might choose to fit interpolatory curves to numeric solutions over a wider range of Γ_D.

B. Comparing NCA Models With a Constant Acceleration Maneuver

In [15] and [17], the authors optimized over the DWPA model, which is a second order direct discrete model in Section VI. An issue with direct discrete models, however, is that they are not internally consistent when the revisit rate of the sensor varies. That is, in a Kalman filter, predictor a tracks covariance matrix an interval of $2T$ forward in time does not provide the same result as performing two predictions of during T to predict forward a target's track. Consequently, we consider the difference in the performance of the DCWA model and the DWPA model.

For the scale of the solution, the 1D problem with $R = 10^3\,\mathrm{m}$ and $T = 4\,\mathrm{s}$ is considered. Though the models are second order, in [15], the disturbance is modeled as going from zero acceleration to A acceleration using a single step disturbance as in (8). Under such a model, we consider choosing the q for each dynamic model to make the position MSE equal R and also to minimize the position MSE. We consider a specific solution instance for $\Gamma_D = 1/2$. We also consider how q changes with Γ_D. The basic results are computed as described in Section IV and are shown in Fig. 2.

One can see that the analytic solutions agree with Monte Carlo simulations (averaged over $10,000$ runs) and that the performance of the DWPA and DCWA models is essentially identical. For the Monte Carlo simulation, the true initial target state was taken to be $\mathbf{x}_0 = [20\,\mathrm{m}, -200\,\mathrm{m/s}, 0]'$ However, the actual position and velocity used don't matter as long as the acceleration is zero. Despite identical performance, the scale factors differ and with the scale factors, the elements of the $\mathbf{Q}$ matrices are not equal. Consequently, one should be able to replace a DWPA noise model with a DCWA noise model without expecting a significant change in performance when considering a constant sampling interval.

C. Performance Prediction of a Weaving Target in 2D with a Linear Filter

In this example, we consider a 2D weaving trajectory. Though simpler than a spiraling trajectory, which in some instances, such as ballistics [38], can be very complicated, weaving trajectories are also difficult to track and have appeared as a difficult trajectory in the tracking benchmark scenario [45]. To obtain the 2D trajectory, we use expressions for a 3D trajectory where all weaving and motion is in the x-y plane and we discard the z component.

For simplicity of simulation, we use the flat-Earth weaving model given in [26, Append. D]. If $\mathbf{r}_t$ is 3D position as a function of continuous time t, and $\ddot{\mathbf{r}}_t$ is velocity, then a 3D model of a weaving trajectory is defined by the acceleration term

$$\ddot{\mathbf{r}}_t = \mathbf{\Omega}_t \times \dot{\mathbf{r}}_t \tag{69}$$

where $\times$ is the cross product operation and

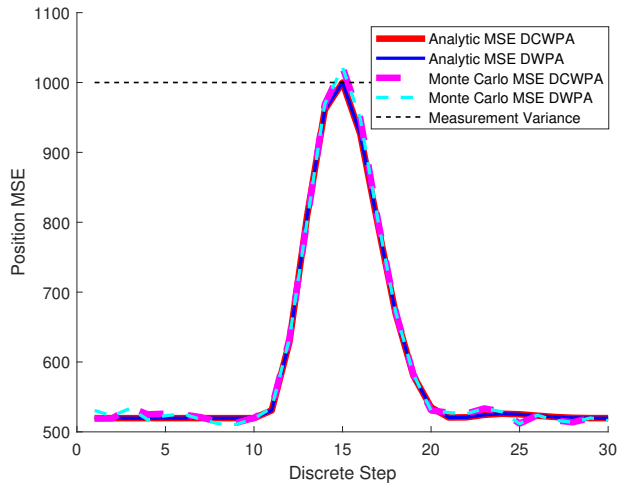
$$\mathbf{\Omega}_t = \begin{bmatrix} 0 \\ 0 \\ A_m \cos(\alpha t + \theta_0) \end{bmatrix} \tag{70}$$

For the scenario here, we choose the same parameters as used for the red line in Fig. 11. in [26], which are about

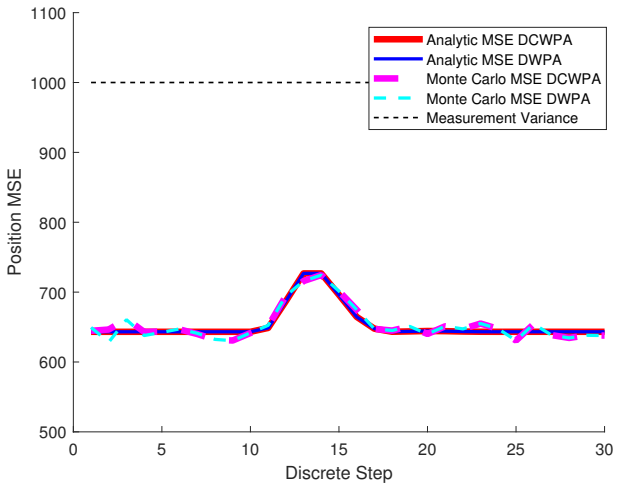
$$\alpha = 1.7794 \tag{71}$$

$$\theta_0 = 0 \tag{72}$$

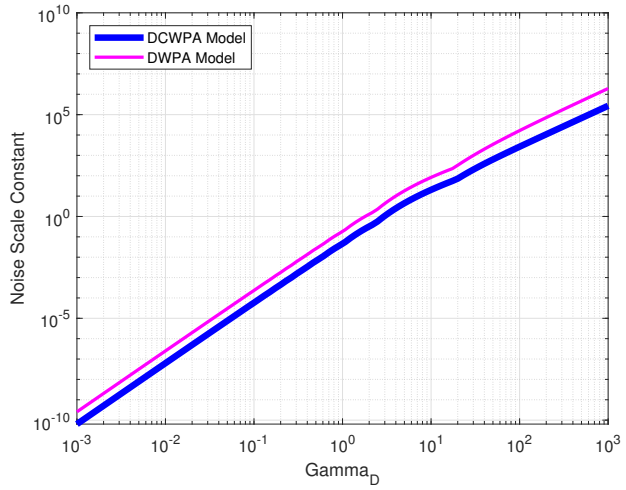
$$A_m = 2.7951 \tag{73}$$



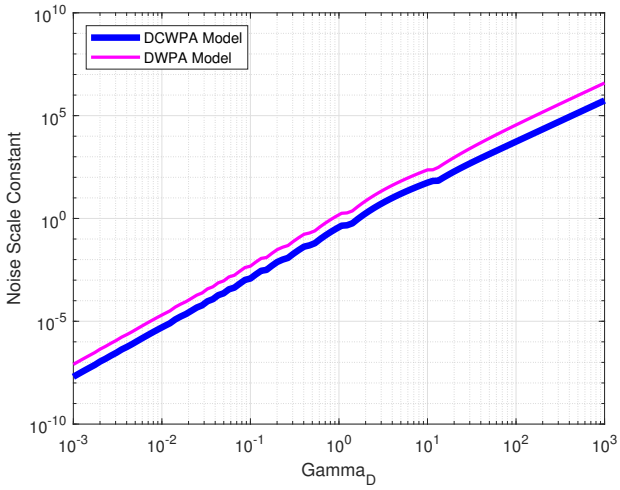
(a) Peak MSE=R



(b) Minimum MSE



(c) Peak MSE=R



(d) Minimum MSE

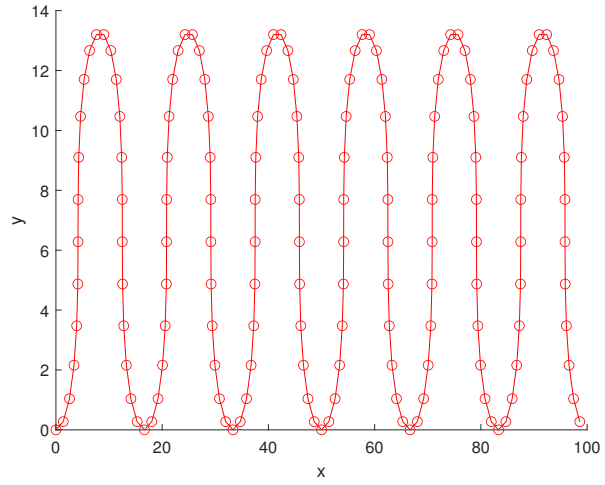
Fig. 2. The MSE (position) performance of a 1D scenario starting with no maneuver (constant velocity) and then a single step acceleration (changing the problem to a constant acceleration scenario) for γ_D is shown in (a) after optimizing for the measurement-equal scale factor in the DWPA model as well as the DCWPA model. In (b), the same thing is shown minimizing the position MSE. Both analytic and Monte Caro results are shown. Given that the performance looks identical, it is shown in (c) and (d) that the scale factors for the covariance matrices of the models do indeed differ when using the parameters of the simulation.

where α and θ are in radians. The total simulation period goes from $t = 0$ to $t = 21.1864$. Units are not specific but we shall assume units of seconds. The initial speed of the target, which is the magnitude of $\dot{\mathbf{r}}_0$, is chosen to be $10\,\mathrm{m/s}$, using assumed distance units of meters. The small distances traveled could be representative of a small UAV. However, one can trivially scale the problem to be representative of an aircraft by scaling the distances and time.

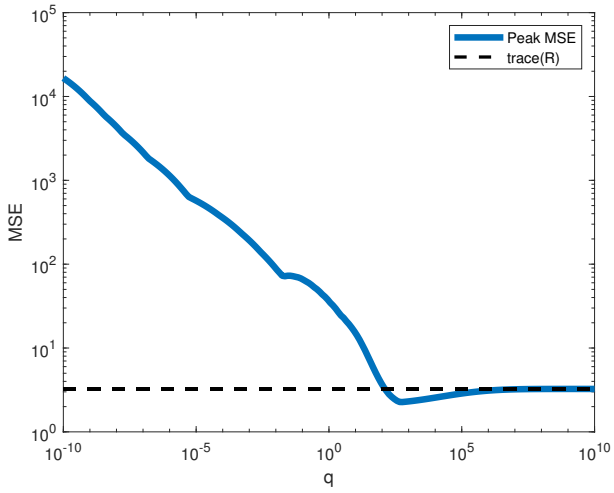
The trajectory is simulated using fourth-order Runge-Kutta integration with $3,000$ steps over the full time interval. The simulated trajectory is similar to one of the examples in the comments to the `aWeave` function in the TCL (though one will have to call `aWeave` to add back in acceleration terms). After simulation, every 20th sample is taken, meaning that the time interval between samples is $T = 0.1412$. Figure 3a shows the trajectory with circles at the samples.

Assuming position measurements of meters, the peak acceleration of the simulated trajectory is A_m times the speed, which is about $A = 27.9508\,\mathrm{m/s^2}$. The measurement matrix extracts the position components of the state

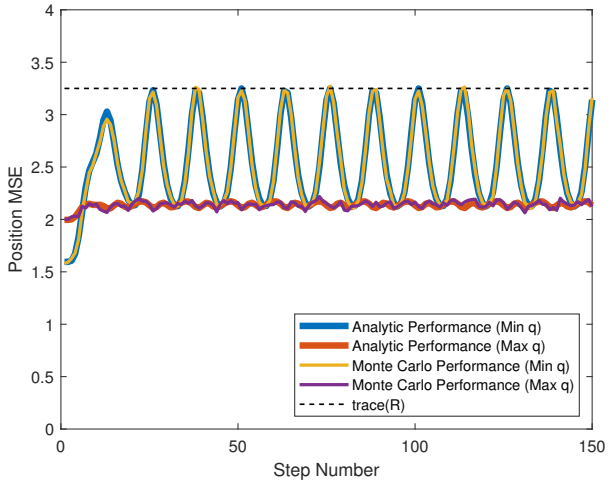
$$\mathbf{H} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{74}$$



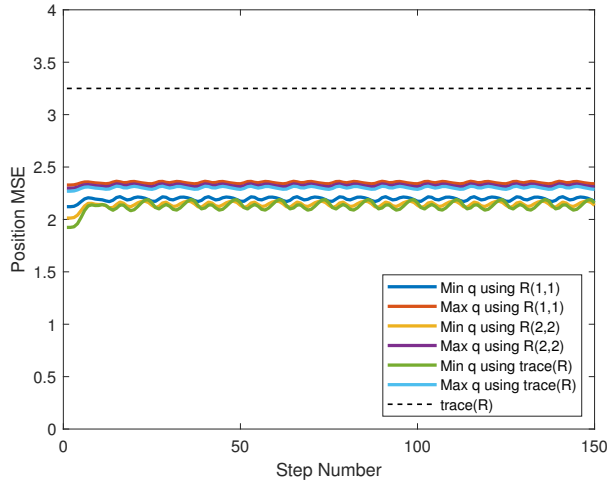
(a) The Weaving Trajectory



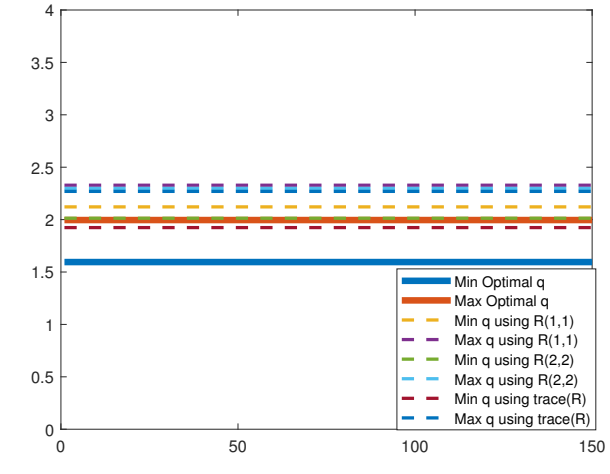
(b) DCWPA Cost Function



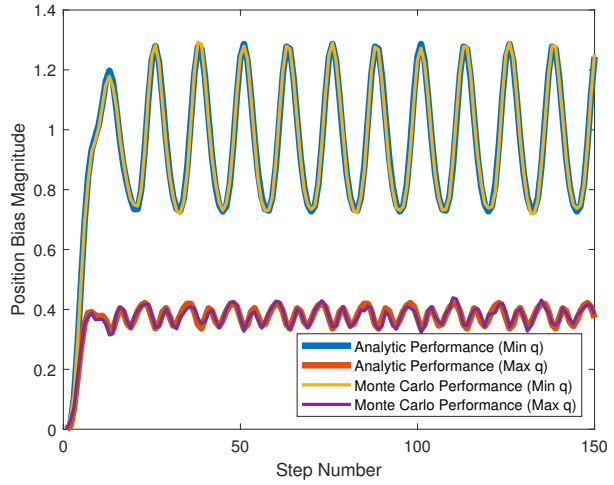
(c) Optimal Gains



(d) Gains Based on Max Acceleration



(e) Performance with No Maneuver



(f) Kalman Filter Bias

Fig. 3. Applying the algorithm of this report to optimize the gain for a 2D weaving trajectory. The trajectory is shown in (a), and (b) shows the cost function. Choosing the intersection of the dotted line and the solid line in provides the minimum gain (measurement MSE =position MSE); choosing the lowest point and provides the maximum gain (minimum MSE) that one should choose. The position MSE performance of the Kalman filter is shown in (c), with the Monte Carlo simulation agreeing with the analytic results. In (d), four suboptimal gains obtained based only on maximum acceleration, as in previous work, are shown. The DCWPA model was used, but very similar results come when using the DWPA dynamic model.

and the measurement covariance matrix for the scenario is taken to be

$$\mathbf{R} = \begin{bmatrix} 1 & 0 \\ 0 & 2.25 \end{bmatrix} \tag{75}$$

where corresponding units are squared meters. The DCWPA dynamic model (a second order model) of Section VI is used. However, the results are similar if one uses the DWPA dynamic model.

The MSE performance of a mismatched Kalman filter, both analytic as well as obtained via averaging over $10,000$ Monte Carlo simulations is shown in Fig. 3c as described in Section IV. For comparison, techniques in that just optimize q assuming a constant acceleration input as was done in Section VII-B are also considered, where the maximum acceleration of the weaving target is used as the parameter over which optimization should be performed. Because the $\mathbf{R}$ matrix in (75) has different values on the diagonal, we choose three scenarios for the scalar R over which optimization is performed

1) Use $R(1,1)$. This leads to a maneuvering index of $\Gamma_D = 0.5576$.
2) Use $R(2,2)$. This leads to a maneuvering index of $\Gamma_D = 0.3717$.
3) Use $\mathrm{trace}(\mathbf{R})$. This leads to a maneuvering index of $\Gamma_D = 0.3093$.

Figure 3d shows the performance based on using these maximum accelerations for optimization. It can be seen that none of the solutions achieve the measurement accuracy and they are also not the MMSE solutions. Thus, one can see value in optimizing over the gains.

Similarly, Fig. 3e displays the position MSE of each of the solutions (optimal and suboptimally based on a maximum acceleration) at the same scale as the other RMSE plot. One can see that the MSE of the optimal measurement equal solution is lower than any of the other solutions. Similarly, the optimal MSE solution given no input is only higher than one of the suboptimal solutions. In other words, there is no real advantage to the suboptimal solutions besides simplicity. Finally, Fig. 3f shows that the computed bias of the optimal solution agrees with the Monte Carlo results, demonstrating the accuracy of the analytic solution.

Finally, it is worth noting that even if both elements of $\mathbf{R}$ in (75) were 1, the optimization based on assuming a constant acceleration input as was done in Section VII-B is not able to find a q such that the position MSE accuracy is even close to the measurement MSE accuracy.

VIII. Conclusion

Recursive expressions for the MSE matrix and bias of a mismatched Kalman filter given an arbitrary true dynamic model were provided when considering both the posterior and the predicted estimates from the filter. Asymptotic solutions for constant maneuvers were also described. Using such solutions, a general approach to determining upper (minimum position MSE) and lower (position MSE =measurement MSE) bounds on the scale factor q of the process noise covariance matrix $\mathbf{Q}$ of the Kalman filter was presented. An example showed that for finite-duration maneuvers, the scale factor q is not always monotonically increasing as a function of the maneuvering index Γ_D. More notably, the optimal scale factor q for a weaving scenario was found under various criteria and was shown to be more accurate than approximations previously suggested in the literature that are based on assuming a constant acceleration. The ability to handle complex models such as weaving could be extended to spiraling models and real recorded trajectories.

Acknowledgements

The author would like to thank W. D. Blair of GTRI for help understanding his algorithms on process noise selection.

Appendix A
Explicit Solutions for Basic Models

Subsections A-A and A-B derive explicit expressions for the asymptotic MSE matrix of a generic filter for first and second order dynamic models with state transition matrix $\mathbf{F}$ given by (53) and (54) expressed in terms of the α, β, and γ of the Kalman filter gain (59). The measurement matrix extracts just the position component of a state with 1D position. That is, for first order models,

$$\mathbf{H}_{\text{First Order}} = \begin{bmatrix} 1 & 0 \end{bmatrix} \tag{76}$$

and for second order models

$$\mathbf{H}_{\text{Second Order}} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \tag{77}$$

Such relations are already present for first order models in [7], [8], and specifically in the context of process noise estimation in [9]. However, they are derived in a more direct manner here. For second order models, such as in [15], no explicit expression for the MSE matrix is provided.

The utility of such expressions is twofold. First, one can combine the first order solutions with the gain relations of Section VI-B to obtain expressions for the "optimal" process noise matrix scaling constant q. Here, "optimal" means that q is either chosen to minimizing the asymptotic position MSE or is chosen such that the asymptotic position MSE equals the measurement variance R. These solutions are given in Appendices A-C, A-D, and A-E. These solutions are new in that they are explicit, though interpolatory formula were already given in [9], [11]. Due to the required polynomial rooting, the explicit solutions prove to be difficult to compute in a numerically stable manner at high target maneuvering indices, which makes a good argument with general dynamic models for numerically computing the solutions as in Section IV and then possibly fitting interpolatory curves as in [9], [11].

However, the second use of explicit MSE solutions is to gain greater insight into how the problem scaled. Comparing the second order MSE matrix of Section A-B with that of Section A-A, one can see that as the order of the system increases, if the input $\mathbf{a}$ is taken to be one order higher than the order of the state, it makes sense to change the definition of the target maneuvering index as the target order increases. Specifically, to replace the target maneuvering index definition of (9) with that of (9). This is relevant, because just as Γ_D plays an important role in expressing the scale of solutions for first order models as in [9], one can see that that the track maneuvering index for second order models $\Gamma_D^{(2)}$ plays a role in determining the scale of second order problems.

A. First Order MSE Matrix

In the first order model, the expressions for $\mathbf{W}$ from (59), for $\mathbf{H}$ from (76) and for $\mathbf{F}$ fom (53) are used. As in (7), $\mathbf{a}$ is taken to be the cumulative effects of a constant acceleration of magnitude A over a time duration of T:

$$\mathbf{a} = \begin{bmatrix} \frac{T^2}{2} \\ T \end{bmatrix} A \tag{78}$$

This leads to $\mathbf{A}$ and $\mathbf{B}$ in (19) being

$$\mathbf{A} = \begin{bmatrix} 1-\alpha & T(1-\alpha) \\ -\frac{\beta}{T} & 1-\beta \end{bmatrix} \tag{79}$$

$$\mathbf{B} = \begin{bmatrix} 1-\alpha & 0 \\ -\frac{\beta}{T} & 1 \end{bmatrix} \tag{80}$$

Using (25), the asymptotic bias term is consequently

$$\mathbf{b}_\infty = \begin{bmatrix} T^2\frac{\alpha-1}{\beta} \\ T\frac{\beta-\alpha}{2\beta} \end{bmatrix} A \tag{81}$$

Substiuting $\mathbf{A}$ and $\mathbf{B}$ into (30), one obtains

$$\begin{aligned} \overbrace{\begin{bmatrix} m_{11} & m_{12} \\ m_{12} & m_{22} \end{bmatrix}}^{\mathbf{M}_\infty} = &\begin{bmatrix} (m_{11}+T(2m_{12}+m_{22}T))(1-\alpha)^2 & \frac{1-\alpha}{T}\left(m_{22}T^2(1-\beta)-m_{11}\beta+m_{12}T(1-2\beta)\right) \\ \frac{1-\alpha}{T}\left(m_{22}T^2(1-\beta)-m_{11}\beta+m_{12}T(1-2\beta)\right) & m_{22}(1-\beta)^2+\frac{\beta}{T^2}(-2m_{12}T(1-\beta)+m_{11}) \end{bmatrix} \\ &+\begin{bmatrix} R\alpha^2+\frac{A^2T^4(1-\alpha)^2(4-\beta)}{4\beta} & \frac{R\alpha\beta}{T}+\frac{A^2T^3(1-\alpha)\left(4+2\alpha-5\beta+\beta^2\right)}{4\beta} \\ \frac{R\alpha\beta}{T}+\frac{A^2T^3(1-\alpha)\left(4+2\alpha-5\beta+\beta^2\right)}{4\beta} & \frac{R\beta^2}{T^2}+\frac{A^2T^2(2-\beta)\left(4\alpha-\beta(4-\beta)\right)}{4\beta} \end{bmatrix} \end{aligned} \tag{82}$$

which in linear in the elements of $\mathbf{M}_\infty$. Solving for the elements leads to explicit expressions for the elements of the MSE matrix for first order models

$$m_{11} = \frac{A^2T^4(1-\alpha)^2}{\beta^2} + \frac{R\left(2\alpha^2+2\beta-3\alpha\beta\right)}{\alpha(4-2\alpha-\beta)} \tag{83}$$

$$m_{22} = \frac{A^2T^2(2\alpha-\beta)^2}{4\beta^2} + \frac{2R\beta^2}{T^2\alpha(4-2\alpha-\beta)} \tag{84}$$

$$m_{12} = \frac{A^2T^3(1-\alpha)(2\alpha-\beta)}{2\beta^2} + \frac{R\beta(2\alpha-\beta)}{T\alpha(4-2\alpha-\beta)} \tag{85}$$

Using the definition of the first order track index of (9), this can be rewritten as

$$m_{11} = R\left(\frac{\Gamma_D^2(1-\alpha)^2}{\beta^2} + \frac{2\alpha^2+2\beta-3\alpha\beta}{\alpha(4-2\alpha-\beta)}\right) \tag{86}$$

$$m_{22} = \frac{R}{T^2}\left(\Gamma_D^2\frac{(2\alpha-\beta)^2}{4\beta^2} + \frac{2\beta^2}{\alpha(4-2\alpha-\beta)}\right) \tag{87}$$

$$m_{12} = \frac{R}{T}\left(\frac{\Gamma_D^2(1-\alpha)}{2\beta^2} + \frac{\beta}{\alpha(4-2\alpha-\beta)}\right)(2\alpha-\beta) \tag{88}$$

For a fixed Γ_D^2, solutions for a process noise covariance matrix scaling that minimizes the position MSE, m_{11} can be obtained for specific noise models by substituting the relation between α and β, as in Section VI-B and either minimizing the results with respect to α or choosing the α such that $m_{11} = R$, depending on the desired optimality criterion.

B. Second Order MSE Matrix

In the second order model, the expressions for $\mathbf{W}$ from (60), for $\mathbf{H}$ from (77) and for $\mathbf{F}$ fom (54) are used. As in (7), $\mathbf{a}$ is taken to be the cumulative effects of a constant jerk (not acceleration) of magnitude A over a time duration of T:

$$\mathbf{a} = \begin{bmatrix} \frac{T^3}{6} \\ \frac{T^2}{2} \\ T \end{bmatrix} A \tag{89}$$

Computing $\mathbf{A}$ and $\mathbf{B}$ using (19)

$$\mathbf{A} = \begin{bmatrix} 1-\alpha & T(1-\alpha) & \frac{1}{2}T^2(1-\alpha) \\ -\frac{\beta}{T} & 1-\beta & T\left(1-\frac{\beta}{2}\right) \\ -\frac{\gamma}{2T^2} & -\frac{\gamma}{2T} & 1-\frac{\gamma}{4} \end{bmatrix} \tag{90}$$

$$\mathbf{B} = \begin{bmatrix} 1-\alpha & 0 & 0 \\ -\frac{\beta}{T} & 1 & 0 \\ -\frac{\gamma}{2T^2} & 0 & 1 \end{bmatrix} \tag{91}$$

Using (25), the asymptotic bias term is thus

$$\mathbf{b}_\infty = \begin{bmatrix} \frac{2T^3(\alpha-1)}{\gamma} \\ \frac{T^2(12\beta-24\alpha-\gamma)}{12\gamma} \\ T\left(\frac{1}{2}-\frac{2\beta}{\gamma}\right) \end{bmatrix} A \tag{92}$$

Substiuting $\mathbf{A}$ and $\mathbf{B}$ into (30), and solving for the elements of $\mathbf{M}_\infty$, one gets

$$m_{11} = \frac{4A^2T^6(-1+\alpha)^2}{\gamma^2} + \frac{4R\beta(-2\alpha^2-2\beta+3\alpha\beta) - R\alpha(-4+2\alpha+\beta)\gamma}{(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)} \tag{93}$$

$$m_{22} = \frac{A^2T^4(24\alpha-12\beta+\gamma)^2}{144\gamma^2} + \frac{R(-8\beta^3+4\beta^2\gamma+(\alpha-2)\gamma^2)}{T^2(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)} \tag{94}$$

$$m_{33} = \frac{A^2T^2(-4\beta+\gamma)^2}{4\gamma^2} - \frac{2R\beta\gamma^2}{T^4(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)} \tag{95}$$

$$m_{12} = \frac{A^2T^5(1-\alpha)(24\alpha-12\beta+\gamma)}{6\gamma^2} - \frac{R(2\alpha-\beta)\beta(4\beta-\gamma)}{T(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)} \tag{96}$$

$$m_{13} = \frac{A^2T^4(1-\alpha)(4\beta-\gamma)}{\gamma^2} - \frac{R\gamma(-4\beta(-2\alpha+\beta)+(-4+2\alpha+\beta)\gamma)}{2T^2(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)} \tag{97}$$

$$m_{23} = \frac{A^2T^3(4\beta-\gamma)(24\alpha-12\beta+\gamma)}{24\gamma^2} + \frac{R\beta\gamma(-4\beta+\gamma)}{T^3(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)} \tag{98}$$

Using the definition of the second order track index in (10), Equations (93) through (98) can be rewritten as

$$m_{11} = R\left(\frac{4\Gamma_{D,2}^2(1-\alpha)^2}{\gamma^2} + \frac{4\beta(-2\alpha^2-2\beta+3\alpha\beta)-\alpha(-4+2\alpha+\beta)\gamma}{(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)}\right) \tag{99}$$

$$m_{22} = \frac{R}{T^2}\left(\frac{\Gamma_{D,2}^2(24\alpha-12\beta+\gamma)^2}{144\gamma^2} + \frac{-8\beta^3+4\beta^2\gamma+(\alpha-2)\gamma^2}{(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)}\right) \tag{100}$$

$$m_{33} = \frac{R}{T^4}\left(\frac{\Gamma_{D,2}^2(4\beta-\gamma)^2}{4\gamma^2} - \frac{2\beta\gamma^2}{(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)}\right) \tag{101}$$

$$m_{12} = \frac{R}{A}\left(\frac{\Gamma_{D,2}^2(1-\alpha)(24\alpha-12\beta+\gamma)}{6\gamma^2} - \frac{(2\alpha-\beta)\beta(4\beta-\gamma)}{(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)}\right) \tag{102}$$

$$m_{13} = \frac{R}{T^2}\left(\frac{\Gamma_{D,2}^2(1-\alpha)(4\beta-\gamma)}{\gamma^2} - \frac{\gamma(-4\beta(-2\alpha+\beta)+(-4+2\alpha+\beta)\gamma)}{2(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)}\right) \tag{103}$$

$$m_{23} = \frac{R}{T^3}\left(\frac{\Gamma_{D,2}^2(4\beta-\gamma)(24\alpha-12\beta+\gamma)}{24\gamma^2} + \frac{\beta\gamma(-4\beta+\gamma)}{(-4+2\alpha+\beta)(4\alpha\beta+(\alpha-2)\gamma)}\right) \tag{104}$$

Thus again, one can see that optimizations related to m_{11} can be expressed in terms of $\Gamma_{D,2}^2$ rather than in terms of R, and T directly.

C. Explicit DWNA Solution

Using (63), one can replace all occurrences of β in (86) with expressions in terms of α to get

$$\frac{m_{11}}{R} = \frac{-2+2\sqrt{1-\alpha}+3\alpha-2\alpha\sqrt{1-\alpha}}{\alpha} + \frac{(\alpha-1)^2\Gamma_D^2}{4(-2+2\sqrt{1-\alpha}+\alpha)^2} \tag{105}$$

Solutions for q are obtained by solving for the appropriate α, using (63) to get β and then using (64) to get q.

To obtain the solution for q such that the asymptotic MSE equals the measurement variance, one simply sets $m_{11} = R$ in (105). The equation is then rearranges to combine isolate the square root term on one side. Both sides are squared and after simplifying, which includes taking a polynomial quotient of the resulting value with respect to $(1-\alpha)^2$ to remove two roots at one (removing them is important for numerical stability), one ends up with the polynomial equation:

$$0 = c_0 + c_1\alpha + c_2\alpha^2 + c_3\alpha^3 + c_4\alpha^4 \tag{106}$$

$$c_0 = 256\Gamma_D^2 \tag{107}$$

$$c_1 = -\Gamma_D^2(576+\Gamma_D^2) \tag{108}$$

$$c_2 = 2\Gamma_D^2(200+\Gamma_D^2) \tag{109}$$

$$c_3 = -\Gamma_D^2(80+\Gamma_D^2) \tag{110}$$

$$c_4 = -64 \tag{111}$$

whereby one just uses (64) with (63) to get the actual gain. Thus, using a scalar polynomial solver, such as the `roots` command in Matlab, one can get all the solutions for α, and then choose the real, positive solution that is less than 1 ($\alpha \geq 1$ implies instability) that provides a result such that m_{11}/R in (105) is closest to 1.

On the other hand, to obtain the solution for q that minimizes the mean squared position error, one just takes the derivative of (105) with respect to α and sets it equal to zero. After coming all fractions and simplifying one gets

$$0 = 4\alpha^3 + (-1+\alpha)(16+5(-4+\alpha)\alpha)\Gamma_D^2 + \sqrt{1-\alpha}(-2\alpha^3(2+\alpha)+(-1+\alpha)(16+(-12+\alpha)\alpha)\Gamma_D^2) \tag{112}$$

Putting the term with the $\sqrt{1-\alpha}$ coefficient on one side of the equation and squaring both sides, one gets rid of the square root. Simplifying, one ends up with the polynomial equation:

$$0 = \sum_{i=0}^{6} c_i\alpha^i \tag{113}$$

$$c0 = 256\Gamma_D^2 \tag{114}$$

$$c1 = -576\Gamma_D^2 \tag{115}$$

$$c2 = 352\Gamma_D^2 - \Gamma_D^4 \tag{116}$$

$$c3 = 12\Gamma_D^2 + 2\Gamma_D^4 \tag{117}$$

$$c4 = -48\Gamma_D^2 - \Gamma_D^4 \tag{118}$$
$$c5 = -12 + 4\Gamma_D^2 \tag{119}$$
$$c6 = -4 \tag{120}$$

As was the case before, this can be solved with a root solver and the valid solution to choose the real, positive solution that is less than 1 that minimizes (105).

D. Explicit DCWNA Solution

The procedure to obtain the explicit solution for DCWNA model is essentially the same as for the DWNA solution of Section A-C. Substituting (61) into (86) one obtains m_{11} just in terms of α

$$\frac{m_{11}}{R} = \frac{12 + 11\alpha^2 - 2\sqrt{36 + 3(\alpha - 12)\alpha} + 3\alpha(-8 + \sqrt{36 + 3(\alpha - 12)\alpha})}{\alpha(-2 + \alpha + \sqrt{36 + 3(\alpha - 12)\alpha})} + \frac{(\alpha - 1)^2\Gamma_D^2}{(-6 + 3\alpha + \sqrt{36 + 3(\alpha - 12)\alpha})^2} \tag{121}$$

Solutions for q are obtained by solving for the appropriate α, using (61) to get β and then using (62) to get q.

To obtain the solution for q such that the asymptotic MSE equals the measurement variance, the solution procedure is as in Section A-C. The polynomial that must be solved for α is

$$0 = \sum_{i=0}^{5} c_i \alpha^i \tag{122}$$
$$c0 = 6912\Gamma_D^2 \tag{123}$$
$$c1 = -16\Gamma_D^2(1368 + \Gamma_D^2) \tag{124}$$
$$c2 = 48\Gamma_D^2(528 + \Gamma_D^2) \tag{125}$$
$$c3 = -\Gamma_D^2(12720 + 49\Gamma_D^2) \tag{126}$$
$$c4 = 18(-96 + 136\Gamma_D^2 + \Gamma_D^4) \tag{127}$$
$$c5 = (1584 - \Gamma_D^2(96 + \Gamma_D^2)) \tag{128}$$

As was the case before, one must then choose the real, positive solution that is less than 1 ($\alpha \geq 1$ implies instability) that provides a result such that m_{11}/R in (121) is closest to 1. Similarly, for the polynomial to solve to get α to minimize the asymptotic MSE is

$$0 = \sum_{i=0}^{9} c_i \alpha^i \tag{129}$$
$$c_0 = -1327104\Gamma_D^2 \tag{130}$$
$$c_1 = 6967296\Gamma_D^2 \tag{131}$$
$$c_2 = 3072\Gamma_D^2(-4941 + \Gamma_D^2) \tag{132}$$
$$c_3 = -256\Gamma_D^2(-68607 + 59\Gamma_D^2) \tag{133}$$
$$c_4 = 128\Gamma_D^2(-88938 + 235\Gamma_D^2) \tag{134}$$
$$c_5 = 32(1944 + 122238\Gamma_D^2 - 959\Gamma_D^4) \tag{135}$$
$$c_6 = 4(-41472 - 134556\Gamma_D^2 + 4139\Gamma_D^4) \tag{136}$$
$$c_7 = 141264 - 7\Gamma_D^2(3048 + 613\Gamma_D^2) \tag{137}$$
$$c_8 = -32400 + 7656\Gamma_D^2 + 386\Gamma_D^4 \tag{138}$$
$$c_9 = -5148 - 480\Gamma_D^2 - 11\Gamma_D^4 \tag{139}$$

E. Explicit DWPA Solution

The procedure to obtain the explicit solution for DWPA model is essentially the same as for the DWNA solution of Section A-C. Substituting (65) and (66) into (93) one obtains m_{11} just in terms of α

$$\frac{m_{11}}{R} = \frac{1}{4}\alpha\left(\frac{-56 + 16\sqrt{1-\alpha} + 60\alpha - 16\alpha\sqrt{1-\alpha}}{-12 + \alpha(12 + \alpha)} + \frac{(\alpha - 1)^2\alpha\Gamma_D^2}{(-2 + 2\sqrt{1-\alpha} + \alpha)^4}\right) \tag{140}$$

Solutions for q are obtained by solving for the appropriate α, using (65) to get β, (66) to get γ and then using (67) to get q.

To obtain the solution for q such that the asymptotic MSE equals the measurement variance, the solution procedure is as in Section A-C, though this time it necessitates dividing out a $(-12+12\alpha+\alpha^2)$ term which doesn't depend on Γ_D. The solutions is

$$0=\sum_{i=0}^{7}c_i\alpha^i \tag{141}$$
$$c_0=-12288\Gamma_D^2 \tag{142}$$
$$c_1=55296\Gamma_D^2 \tag{143}$$
$$c_2=4\Gamma_D^2(-24320+3\Gamma_D^2) \tag{144}$$
$$c_3=83968\Gamma_D^2-36\Gamma_D^4 \tag{145}$$
$$c_4=\Gamma_D^2(-36192+35\Gamma_D^2) \tag{146}$$
$$c_5=6864\Gamma_D^2-10\Gamma_D^4 \tag{147}$$
$$c_6=192-\Gamma_D^2(368+\Gamma_D^2) \tag{148}$$
$$c_7=-256 \tag{149}$$

As was the case before, one must then choose the real, positive solution that is less than 1 that provides a result such that m_{11}/R in (140) is closest to 1. Similarly, for the polynomial to solve to get α to minimize the asymptotic MSE, after dividing out a $(-12+12\alpha+\alpha^2)$ term, is

$$0=\sum_{i=0}^{10}c_i\alpha^i \tag{150}$$
$$c_0=-331776\Gamma_D^2 \tag{151}$$
$$c_1=-432\Gamma_D^2(-4352+\Gamma_D^2) \tag{152}$$
$$c_2=2304\Gamma_D^2(-1936+\Gamma_D^2) \tag{153}$$
$$c_3=72\Gamma_D^2(79296-67\Gamma_D^2) \tag{154}$$
$$c_4=24\Gamma_D^2(-175916+203\Gamma_D^2) \tag{155}$$
$$c_5=3\Gamma_D^2(593792-745\Gamma_D^2) \tag{156}$$
$$c_6=-389320\Gamma_D^2+226\Gamma_D^4 \tag{157}$$
$$c_7=720+32888\Gamma_D^2+85\Gamma_D^4 \tag{158}$$
$$c_8=4(-384+\Gamma_D^4) \tag{159}$$
$$c_9=-16(-51+\Gamma_D^2) \tag{160}$$
$$c_{10}=16 \tag{161}$$

REFERENCES

[1] The tracker component library. [Online]. Available: https://github.com/USNavalResearchLaboratory/TrackerComponentLibrary

[2] T. J. Arnold, "Noise covariance estimation for linear systems," Ph.D. dissertation, University of Wisconsin- Madison, Madison, Wisconsin, 2020.

[3] W. F. Arnold, III and A. J. Laub, "Generalized eigenproblem algorithms and software for algebraic Riccati equations," *Proceedings of the IEEE*, vol. 72, no. 12, pp. 1746–1754, Dec. 1984.

[4] Y. Bar-Shalom, X. Rong Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. New York: John Wilet and Sons, Inc., 2001.

[5] R. H. Bartels and G. W. Stewart, "Solution of the matrix equation $ax + xb = c$ [F4]," *Communications of the ACM*, pp. 820–826, Sep. 1972.

[6] D. P. Bertsekas, *Nonlinear Programming*, 3rd ed. Athena Scientific, 2016.

[7] W. D. Blair, "Fixed-gain, two-stage estimators for tracking maneuvering targets," Naval Surface Warfare Center: Dahlgren Division, Dahlgren, VA, Tech. Rep., Jul. 1992.

[8] ——, "Fixed-gain two-stage estimators for tracking maneuvering targets," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 29, no. 3, pp. 1004–1014, Jul. 1993.

[9] ——, "Design of nearly constant velocity track filters for tracking maneuvering targets," in *Proceedings of the 11th International Conference on Information Fusion*, Cologne, Germany, 30 Jun.–3 Jul. 2008.

[10] ——, "Design of nearly constant velocity track filters for brief maneuvers," in *Proceedings of the 14th International Conference on Information Fusion*, Chicago, IL, 5–8 Jul. 2011.

[11] ——, "Design of nearly constant velocity filters for radar tracking of maneuvering targets," in *Proceedings of the IEEE Radar Conference*, Atlanta, GA, 7–11 May 2012, pp. 1008–1013.

[12] ——, "Design of exponentially-correlation acceleration error filters for tracking maneuvering targets," in *Proceedings of the International Radar Conference*, Toulon, France, 23–27 Sep. 2019.

[13] ——, "NCV filter design for radar tracking of maneuvering targets with LFM waveforms," in *Proceedings of the IEEE Radar Conference*, Boston, MA, 22–26 Apr. 2019.

[14] ——, "Performance of the NCV Kalman filter with ECAE for tracking maneuvering targets," in *Proceedings of the 22nd International Conference on Information Fusion*, Ottawa, Canada, 2–5 Jul. 2019.

[15] ——, "Design of NCA filters for tracking maneuvering targets," in *Proceedings of the IEEE Radar Conference*, Florence, Italy, 21–25 Sep. 2020.

[16] ——, "Industry tip: Picking the minimum process noise variance for your NCV track filter," *IEEE Aerospace and Electronic Systems Magazine*, vol. 36, no. 2, pp. 72–74, Feb. 2021.

[17] W. D. Blair and Y. Bar-Shalom, "On the NCA versus NCV models in tracking maneuvering targets," in *Proceedings of the IEEE Radar Conference*, San Antonio, TX, 2023, submitted.

[18] ——, "MSE design of nearly constant velocity Kalman filters for tracking targets with deterministic maneuvers," *IEEE Transactions on Aerospace and Electronic Systems*, accepted 2023.

[19] W. D. Blair and P. A. Miceli, "Performance prediction of multisensor tracking systems for single-maneuvering targets," *Journal of Advances in Information Fusion*, vol. 7, no. 1, pp. 28–45, Jun. 2012.

[20] Y. Boers and H. Driessen, "Modified Riccati equation and its application to target tracking," *IEE Proceedings Radar, Sonar and Navigation*, vol. 153, no. 1, pp. 7–12, Feb. 2006.

[21] P. Boström-Rost, D. Axehill, W. D. Blair, and G. Hendeby, "Optimal range and beamwidth for radar tracking of maneuvering targets using nearly constant velocity filters," in *Proceedings of the IEEE Aerospace Conference*, Big Sky, MT, 7–14 Mar. 2020.

[22] R. P. Brent, "An algorithm with guaranteed convergence for finding a zero of a function," *The Computer Journal*, vol. 14, no. 4, pp. 422–425, 1971.

[23] C. Chang and K. P. Dunn, "Kalman filter compensation for a special class of systems," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 13, no. 6, pp. 700–706, Nov. 1977.

[24] D. F. Crouse, "Cubature/ unscented/ sigma point Kalman filtering with angular measurement models," in *Proceedings of the International Conference on Information Fusion*, Washington, D.C., Jul. 2015, pp. 1550–1557.

[25] ——, "A general solution for optimal fixed-gain (α-β-γ etc.) filters," *IEEE Signal Processing Letters*, vol. 22, no. 7, pp. 901–904, Jul. 2015.

[26] ——, "Simulating aerial targets in 3D accounting for the Earth's curvature," *Journal of Advances in Information Fusion*, vol. 10, no. 1, pp. 31–57, Jun. 2015.

[27] ——, "The tracker component library: Free routines for rapid prototyping," *IEEE Aerospace and Electronic Systems Magazine*, vol. 32, no. 5, pp. 18–27, May 2017.

[28] J. Duník, O. Straka, and O. Kost, "Measurement difference autocovariance method for noise covariance matrices estimation," in *Proceedings of the IEEE 55th Conference on Decision and Control*, Las Vegas, NV, 12–14 Dec. 2016, pp. 365–370.

[29] J. Duník, O. Straka, O. Kost, and J. Havlík, "Noise covariance matrices in state-space models: A survey and comparison of estimation methods-part I," *International Journal of Adaptive Control and Signal Processing*, vol. 31, no. 11, pp. 1505–1543, Nov. 2017.

[30] T. Ender, R. F. Leurck, B. Weaver, P. Miceli, and W. D. Blair, "System-of-systems analysis of ballistic missile defense architecture effectiveness through surrogate modeling and simulation," *IEEE Systems Journal*, vol. 4, no. 2, pp. 156–166, Jun. 2010.

[31] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 4th ed. Baltimore: The Johns Hopkins Press, 2013.

[32] S. N. Gupta and S. M. Ahn, "Closed-form solutions of target-tracking filters with discrete measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 19, no. 4, pp. 532–538, Jul. 1983.

[33] P. R. Kalata, "The tracking index: A generalized parameter for α-β and α-β-γ target trackers," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 20, no. 2, pp. 174–182, Mar. 1984.

[34] G. N. Lewis and T. A. Postol, "Future challenges to ballistic missile defense," *IEEE Spectrum*, vol. 37, no. 9, pp. 60–68, Sep. 1997.

[35] P. S. Maybeck, *Stochastic Models, Estimation, and Control*. Academic Press, 1982, vol. 2.

[36] R. K. Mehra, "On the identification of variances and adaptive Kalman filtering," *IEEE Transactions on Automatic Control*, vol. 15, no. 2, pp. 175–184, Apr. 1972.

[37] P. Mookerjee and F. Reifler, "Reduced state estimator for systems with parametric inputs," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 40, no. 2, pp. 446–461, Apr. 2004.

[38] D. H. Platus, "Ballistic re-entry vehicle flight dynamics," *Journal of Guidance, Control, and Dynamics*, vol. 5, no. 1, pp. 4–15, Jan. – Feb. 1982.

[39] R. M. Stein and J. Little, "'Future challenges to ballistic missile defense' paints wrong picture," *IEEE Spectrum*, vol. 34, no. 11, pp. 70–72, Nov. 1997.

[40] L. D. Stone, R. L. Streit, T. L. Corwin, and K. L. Bell, *Bayesian Multiple Target Tracking*, 2nd ed. Boston: Artech House, 2014.

[41] J. J. Sudano, "Maneuver-driven α-β and α-β-γ tracking filters," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 31, no. 1, pp. 341–357, Jan. 1995.

[42] L. F. Urbano, P. Kalata, and M. Kam, "Optimal tracking index relationship for random and deterministic target maneuvers," in *Proceedings of the IEEE Radar Conference*, Atlanta, GA, 7–11 May 2012, pp. 999–1003.

[43] D. A. Vallado and W. D. McClain, *Fundamentals of Astrodynamics and Applications*, 4th ed. Hawthorne, CA: Microcosm Press, 2013.

[44] D. R. Vaughan, "A nonrecursive algebraic solution for the discrete Riccati equation," *IEEE Transactions on Automatic Control*, vol. 15, no. 5, pp. 597–599, Oct. 1970.

[45] G. A. Watson and D. H. McCable, "Benchmark problem with a multisensor framework for radar resource allocation and the tracking of highly maneuvering targets, closely spaced targets, and targets in the presence of sea-surface-induced multipath," Naval Surface Warfare Center, Dahlgren Division, Dahlgren, VA, Tech. Rep. NSWCDD/TR-99/32, May 1999.

[46] P. Zarchan and H. Musoff, *Fundamentals of Kalman Filtering: A Practical Approach*, 3rd ed. American Institute of Aeronautics and Astronautics, Inc., 2009.