Track Filtering in the Weapon System

Gregg Bock

Three Topics to Discuss

- Reduced State Estimation
- Track Filtering in the Weapon System
  - Track Filtering Philosophies
  - Tracking Index
- Optimal Cost Function and the Weapon System Designer
  - What is Optimal?
  - What does one Consider When Designing a Filter?

REDUCED STATE ESTIMATOR (RSE)

Reduced State Estimator

- The Reduced State Estimator (RSE) is a recursive filter which is of particular interest to the weapon system designer
- The reduced state estimator falls more inline with the historical perspective of design to the worst case error which was used in the $\alpha - \beta$ tracker days, but it is achieved through recursive algorithms
- The RSE considers the error in the target model to be deterministic rather than zero-mean Gaussian
  - RSE target model error = $\mathcal{N}(\lambda, 0)$
    - λ is often chosen to be the largest non-zero value of the reduced state
    - For example, λ represents the maximum acceleration of the target if the RSE is a two state (position and velocity) estimator
  - Kalman target model error = $\mathcal{N}(0, \sigma)$
- The RSE minimized the track uncertainty matrix but can be approximated for our purposes by minimizing the mean squared error of estimate $(\sigma_p^2 + \sigma_v^2 + bias_p^2 + bias_v^2)$

Reduced State Estimator

- To use an RSE in its recursive form, some additional terms must be defined
  - G is the matrix that describes the growth target model error over time period, T
    - $G = \begin{bmatrix} \frac{1}{2}T^2 \\ T \end{bmatrix}$ can be considered a normalized error due to acceleration
  - D is the matrix that provides an estimate of the filter lag at the current time
    - Obviously, D and G are related
    - If one never "corrected" the state estimates when a measurement was received, $D = G$ at all times
    - When the state estimates are updated, D is adjusted to reflect the filter lag at that time
- The RSE keeps track of the errors (random and bias) separately
  - Random error is in matrix, M
  - Bias is contained in the filter lag matrix, D, such that the $B = D\ Acc$
    - Where B is a matrix of bias due to a persistent acceleration of Acc
    - Note that it is not required for $Acc = \lambda$

Comparison of Recursive Algorithms
Reduced State Estimator

Kalman Filter

- State Prediction

$$\hat{x} = \varphi\,\hat{x}$$
$$P = \varphi\,P\,\varphi^T + Q$$

- Gain Computation

$$K = P\,H^T\,(H\,P\,H^T + R)^{-1}$$

- State Correction

$$\hat{x} = \hat{x} + K\,(z - H\,\hat{x})$$
$$P = (I - KH)P,\ or$$
$$P = (I - KH)P(I - KH)^T + KRK^T$$

Reduced State Estimator

- State Prediction

$$\hat{x} = \varphi\,\hat{x}$$
$$M = \varphi\,M\,\varphi^T$$
$$D = \varphi\,D + G$$

This is running count of the lag in the system

$$S = M + D\,\lambda^2\,D^T$$

- Gain Computation

$$K = S\,H^T\,(H\,S\,H^T + R)^{-1}$$

- State Correction

$$\hat{x} = \hat{x} + K(z - H\,\hat{x})$$
$$M = (I - KH)M\,(I - KH)^T + KRK^T$$
$$D = (I - KH)D$$

Steady State
Reduced State Estimator

- Similar to the Kalman filter, the RSE achieves steady state, given the following conditions are consistent and constant
  - Update rate, T
  - Measurement accuracy, σ_m
  - Target model error, λ
- While the filter reaches steady state, the bias estimate will only be fully realized if the object consistently achieves a constant acceleration for a duration greater than or equal to the filter settling time (approx. 3τ)

Steady State Variance
Reduced State Estimator

- The variance due to due measurement noise, M, is the same as the definition used in the Kalman filter

$$M_{k|k} = \begin{bmatrix} 2\alpha^2 + \beta(2-3\alpha) & \beta(2\alpha-\beta)\frac{1}{T} \\ \beta(2\alpha-\beta)\frac{1}{T} & \frac{2\beta^2}{T^2} \end{bmatrix} \frac{\sigma_m^2}{\alpha(4-2\alpha-\beta)}$$

Steady State Lag
Reduced State Estimator

- The filter matrix, D, describes the lag in the presence of a deterministic model error
- At steady state, the filter lag matrix, D, can be defined as

$$D = \begin{bmatrix} \frac{1}{2}\tau^2 \\ \tau \end{bmatrix} = \begin{bmatrix} \frac{1-\alpha}{\beta} T^2 \\ \left(\frac{2\alpha-\beta}{2\beta}\right) T \end{bmatrix}$$

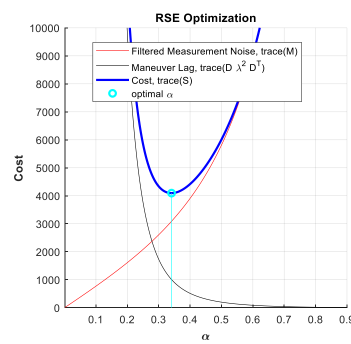
- The bias due to the target model error of the filter design, λ, is

$$B = D\,\lambda$$

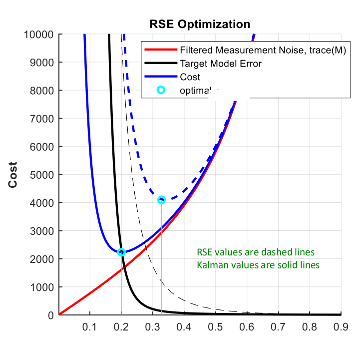
- But, the bias can be evaluated for any target model error, Acc

$$B = D\,Acc$$

RSE Optimization



Kalman and RSE Optimization



General Observations Regarding the RSE

- The filter gains for an RSE are almost always lighter (greater in value) than the filter gains of a Kalman filter given the same magnitude of model error
- This is because the RSE is designed for a persistent (constant) error and the Kalman filter is designed with the assumption that the mean error is zero, but there is some perturbation about the value which is Gaussian in nature
- The RSE has a property similar to the DWNA filter
  - Filter gain relationship is the same

    $$\beta = 2\left(1 - \sqrt{1-\alpha}\right)^2$$
  - This is not surprising given the assumption of the DWNA filter that the target model error is constant between track updates
- While the RSE is wonderful at computing a real-time estimate of filter lag and filter covariance, the optimality criteria is not readily useful unless the mean square error of the state estimate is of importance
  - Most weapons system are designed using the concept of optimal (minimal) uncertainty size for specific confidence thresholds (e.g. minimize the 95% uncertainty area)

TRACK FILTERING AND THE WEAPON SYSTEM

Track Filtering and the Weapon System

- There are two dominant track filtering philosophies in weapon system engineering
  1. Filters are designed for optimal state estimation
  2. Filters are designed to meet weapon system requirements

Designed for Optimal State Estimation

- The filter is designed such that the errors are minimized according to the academic definition of optimal state estimates
- Track data may not meet the needs of specific WCS functions
- WCS functions may need to be designed to compensate for such flaws

Designed for Weapon System Requirements

- The filter is designed to meet specific weapon system requirements
- This may require the weapon system to have more than one filter for each target

Neither Philosophy is Wrong, but the Method Used will Define The Weapon System Architecture

Weapon System Functions Affected By Filter Design

Functions	Desired State Estimate Characteristics
Guidance & Intercept Point Prediction	• Minimize the error in the predicted intercept point • Velocity error is weighted by tgo
Target cueing	• Minimize the error of the position estimate at the time of search
Scheduling	• Consistent time of flight estimates for a given intercept range (velocity error weighted heavily)
Illuminator pointing	• Minimize the error of the position estimate at the time of illuminator transmission (position error weighted heavily)
Kill evaluation	• Highly dependent upon the type of kinematic tests used in kill evaluation • Balance between detection of kinematic change and the desire to have a small lag

Note that None of these Requirements are the "Academic Definition of the Optimal State Estimate"

Filter Designed to Meet WCS Requirements
Filter Design Options

- To achieve the specific requirements of the WCS sub-functions, the weapon system designer must either
  - Determine the process noise (or deterministic error of the RSE) to generate the proper error characteristics in a recursive filter
  - Use an $\alpha - \beta$ filter to determine the specific gains required to meet the unique cost function
- Altering the process noise of a recursive filter to meet specific WCS requirements is often a trial and error process as a mapping from process noise to desired steady state characteristics must be developed
- Using an $\alpha - \beta$ filter may require special logic to determine when to use "steady state" gains and when to use transient gains
  - Determining gains to transition from filter initialization to steady state
  - Properly handle changes in the measurement quality or frequency (leaving steady state and going into a transient state)

Unique Filtering Requirements
Filter Design Options

- The $\alpha - \beta$ filter allows the filter designer to create a cost function unique to the problem at hand
  - While we looked at optimizing a cost function in our last lecture, we considered the Kalman filter cost function
  - If one was to use an $\alpha - \beta$ filter, one could develop a steady state cost function for any set of filter performance characteristics
    - $J = n^2\,\sigma_p^2 + m^2\,bias_p^2$
    - $J = n^2\,\sigma_v^2 + m^2\,bias_v^2$
    - $J = n\,\sigma_p + m\,bias_p$
    - $J = n\,\sigma_v + m\,bias_v$
    - Etc.

  In these examples, m and n are confidence factors
- However, the $\alpha - \beta$ filter does not have a recursive algorithm to select filter gains
- A means of selecting "optimal steady state gains for any cost function" is required

Tracking Index

- Is there a way to determine steady state filter gains without running a recursive filter?
  - Yes, it is called the tracking index
  - All steady state filter gains can be described via tracking index
- The previous lecture introduced the concept of "steady state" as a means to estimate filter performance
  - Filter gains dictate steady state performance
  - Filter gains are a function of measurement accuracy, update rate and process noise
- A tracking index is a dimensionless quantity that relates the characteristics of the sensor track filter to a set of filter gains:

$$\Gamma = \frac{position\ error\ due\ to\ process\ noise\ during\ sample\ period}{measurement\ uncertainty}$$

  - None of the parameters in the tracking index indicate a specific cost function
  - It is the ideal method for determining steady state gains for a given cost function

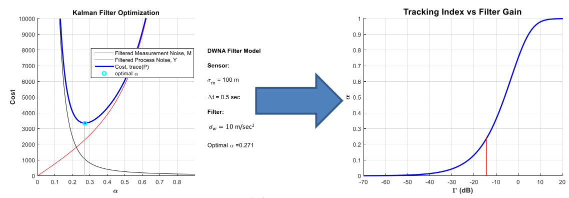
Tracking Index

- The tracking index definition is dependent upon the process noise model
- There is a considerable amount of literature on the tracking index for all of the commonly used track filters (especially the Kalman filter)
- A tracking index can be used to describe the gains of a specific filter
  - $\alpha = f(\Gamma)$
  - $\beta = f(\alpha)$
  - Etc.
- Since the function which relates Γ to α isn't' always conducive to "back of the envelope" work, a table look up can be used

Filter Model	Tracking Index
2-State DWNA	$\Gamma^2 = \frac{\sigma_w^2 T^4}{\sigma_m^2}$
2 State CWNA	$\Gamma^2 = \frac{\sigma_w^2 T^3}{\sigma_m^2}$
2-State RSE	$\Gamma^2 = \frac{Acc^2 T^4}{\sigma_m^2}$

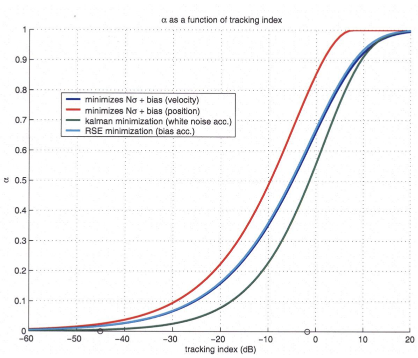
Tracking Index vs Filter Gains

- The same process that was used to determine the optimal gains for a single scenario, can be expanded across a wide range of scenarios to develop a tracking index



$$\Gamma = \frac{2.5}{100} \approx -16\, dB$$

Tracking Index and Cost Functions



Filter Designed for Optimal State Estimates

Filters Design Options

- A filter designed for optimal state estimation often requires significant modifications within many of the weapon system sub-functions
  - Guidance & intercept point prediction
  - Target cueing
  - Scheduling
  - Kill evaluation
- There are many modifications that can mitigate the issues caused by an over (or under) responsive filter
  - An over responsive filter has higher gain values than desired and will allow more noise into the track estimate
  - An under responsive filter has lower gain values than desired and will allow more bias into the track estimate
- The next slide depicts some potential trouble when the filter is over responsive
  - The suggested modification is only one way to mitigate the effect of the over responsive filter on each sub-function

Filter Designed for Optimal State Estimates

Filters Design Options

Functions	Issue and Potential Modifications
Guidance & Intercept Point Prediction	• Excessive movement in the predicted intercept point results in poor guidance efficiency and decreased weapon range • One method of mitigating this effect is to only update the predicted intercept point in the guidance loop when the movement in the intercept point is larger than the noise characteristics predict
Target cueing	• "Holes" can develop in the search algorithm if the changes in cue are too abrupt • One can hold the cue stationary until an entire scan of the uncertainty region is complete, then begin a new scan with the new cue information
Scheduling	• Constant change to the predicted time of intercept creates scheduling issues as "reserved" resources often stolen by higher priority engagements • The scheduler would need to consider introducing a large pad around its reservation time to account for any time of flight noise
Kill evaluation	• One would be reliant upon a high order filter in order to estimate the target's kinematic changes directly rather than infer the changes from the lag

OPTIMAL COST FUNCTIONS AND THE WEAPON SYSTEM DESIGNER

"Optimal" Cost Functions

- Optimal is a term used far too often in a cavalier fashion
  - Optimality has specific criteria and caveats
  - This section explains optimality for specific application in the weapon system which is not the same as optimal state estimation from an academic viewpoint
- The weapon system designer has tough decisions to make which require the sacrifice of optimality for the sake of simplicity
  - The initial intent of a design may be optimality
  - The end result is often "is it good enough to satisfy all the criteria"
- No over-arching optimal set of criteria for weapon system functionality exists
  - Optimality depends upon the system of interest
  - To avoid conflicting requirements (and to make things tractable for our discussion), each function will be evaluated in a vacuum to avoid conflicting requirements

Deriving a Track Accuracy Requirement
"Optimal Cost Functions"

- As an example of the trade-offs one must make can be illustrated using the track accuracy criteria for guidance and intercept prediction

Requirement	Description	Rationale
Capability	Projectile must be able to fly from point A to point B	It is a "top level requirement"
Consistency	Projectile must have similar final flight time each time it flies to a given point	Predicting time of flight accurately is critical to prelaunch functionality
Responsiveness	Must be able to respond to changes in intercept point due to target maneuver	Failure to respond quickly will respond in a large heading error for terminal guidance
Efficiency	Projectile must not "bleed off" too much velocity due to intercept point wandering	Final projectile kinematics properties should be consistent, predictable, and the projectile should have as high a final speed as possible

Track Accuracy Requirements are Derived from Desired Characteristics of Guidance Loop

The Balancing Act
Track Accuracy Requirement for Guidance

- While the characteristics of the guidance loop are well defined, the resultant derived requirements for track accuracy may conflict
- In this example, the desire for high gains to satisfy missile requirements when engaging a maneuvering target directly conflict with the desired low gains to support guidance efficiency, consistency, and maximizing intercept range

Guidance Requirement	→	Track Requirement	→	Filter Characteristics
Capability (max range)		Small Variations in PIP		Small gains (α, β)
Consistency (similar TOF)		Small Variations in PIP		Small gains (α, β)
Responsiveness (small ε)		Quick Reaction Time		* High gains near handover (α, β)
Efficiency (consistent PIP)		Consistent PIP		Small gains (α, β)

* A compromise has already been met by determining higher filter gains are only needed at handover and not throughout the entire flight

Arriving at a Design
Track Accuracy Requirement for Guidance

- Our four desired requirements for track filtering in support of guidance is now reduced to two "types" of characteristics
  - A quick response filter to effectively follow maneuvers
  - A slow response filter to reduce noise in the system
- This concept can be used to further define our filter requirements by noting that the quick response is only necessary when a target is maneuvering and the filter cannot recover in time (too much bias in the estimate) to support handover
- Thus, a filter requirement can be generated that states the filter gains are a function of time-to-go

Filter Characteristics	→	Filter Characteristics
High gains near handover (α, β)		$(\alpha, \beta) = f(tgo)$
Small gains (α, β)		

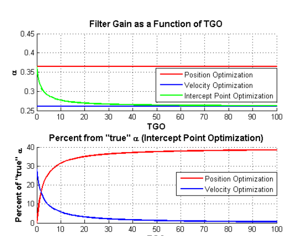
An "Optimal" Track Filtering Cost Function for Guidance

- The filter characteristics have been consolidated into a concept that satisfies missile guidance requirements across a battlespace and for various threat capabilities
- A very simple cost function may look something like this:

$$J = n^2(\sigma_p^2 + 2\sigma_{pv} w + \sigma_v^2 w^2) + m^2 \left(bias_p + bias_v w\right)^2$$

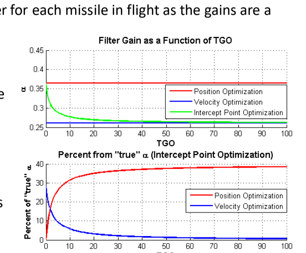
- Where
  - n and m are confidence factors
  - w is the weighting factor, $w = f(tgo)$
  - For simplicity, one could make $w = tgo$

Filter Characteristics
$(\alpha, \beta) = f(tgo)$



The Last Hurdle Before Implementation

- While the "optimal" filter gains are a function of time to go, there is another part of the system we didn't consider – computer resources
- As it turns out, computer resources will ultimately determine the maximum complexity of any design
- Our filtering approach calls for a separate filter for each missile in flight as the gains are a function of time-to-go
  - The need for one filter, per target, per missile is extremely costly, especially if your system is designed to shoot multiple missiles at a given target
  - Sometimes, additional compromises are made to "fit" the design into the hardware
- What are some things you can do to solve this problem and reduce the load on the CPU?



Implementation Strategies

- Derive track processing requirements from the requirements of a specific WCS function
- Determine which type of filter design to use, which will dictate the next course of action

$\alpha - \beta$ Filter

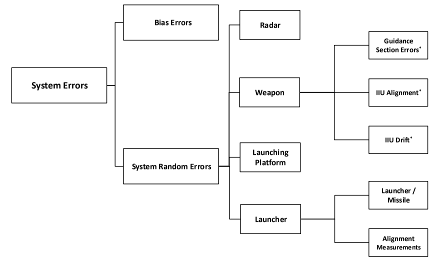
- Use cost function to determine filter gains
- State Uncertainty is calculated from the filter gains
- Special logic must be used to handle transient (non-steady state) environment
- Worst case error estimate are calculated off-line, just as all the state estimation errors are calculated

Recursive Filter

- Find a “mapping” function as a means to set the target model error (process noise) such that the correct filter gains are used to satisfy the cost function
- State uncertainty matrices are calculated in the filter
- Off-line calculations are required for worst case error estimates
  - Complexity of calculations depend upon the filter model

Next Time….

- The knowledge gained throughout the course will be used to explore the concept of error budgets



Resource / Suggested Reading

- <u>Good Reference Material</u>:
  1. Bar-Shalom, *Estimation with Applications to Tracking and Navigation: Theory Algorithms and Software*. 2001
  2. Gelb, *Applied Optimal Estimation*. MIT Press, 1974 (out of print, but easy to find in the library)
- <u>Canonical Track Filtering Documents</u>
  3. Portman, G., Moore, J., Bath, W., *Separated Covariance Filtering*. IEEE International Radar Conference, 1990.
  4. Mookerjee P., Reifler, F., *Reduced State Estimator with Parametric Inputs*. IEEE Transactions of Aerospace and Electronic Systems, Vol. 40 No. 2., April 2004