Multiple Target Tracking: APPENDIX

Tracking and Data Fusion with a Requirements Perspective

Table of Contents

Classic Kalman Filter Equations

The classic extended Kalman filter (EKF) equations consist of extrapolation of a state vector over time, extrapolation of its covariance matrix over time, and an update of the state vector and its covariance by fusing this result with measured data. The update performs a linear, unbiased data fusion of sensor measurements and the state vector at the effective time of the measurements. When transformations in the equations are nonlinear or noises that drive state vectors, or corrupt measurement signals are non-Gaussian, the random errors are non-Gaussian and the EKF equations track variances.

The State Vector and Covariance Extrapolation Equations

The analog, and most general, state vector extrapolation is the differential equation, is

$$\frac{d\vec{x}}{dt}=\vec{f}(\vec{x})+G\cdot\vec{w},\ \vec{x}(t_{OLD})=\vec{x}_{OLD}\ . \tag{1}$$

A process noise driving function $\vec{w}$ is included in (1), which is mapped to the state space by a mapping matrix G. The driving noise vector $\vec{w}$ is made up of elements that are Gaussian noise with mean zero, and $\vec{w}$ has covariance Q. The mapping matrix G in (1) maps the process noise to the state vector when the process noise is not the same rank as the state vector. The state vector equation driving noise is sometimes called "plant noise" because some early Kalman filter applications modeled industrial plants. We will use the term "process noise."

When the differential equation is linear or sufficiently near linear for a linear approximation to achieve sufficient accuracy, we extrapolate the state vector using the discrete state vector extrapolation equation,

$$\vec{x}_{EXT}=\Phi(t_{EXT},t_{OLD})\cdot\vec{x}_{OLD},\ \Phi(t_{EXT},t_{OLD})=\frac{\partial\vec{x}_{EXT}}{\partial\vec{x}_{OLD}}\ . \tag{2}$$

The covariance extrapolation is a matrix equation, the linear variance equation,

$$\frac{dP}{dt}=F\cdot P+P\cdot F^{T}+G\cdot Q\cdot G^{T},\ F=\frac{\partial\vec{f}(\vec{x})}{\partial\vec{x}},\ P(t_{OLD})=P_{OLD}\ . \tag{3}$$

The linearized form of the covariance extrapolation is

$$P_{EXT}=\Phi(t_{NEW},t_{OLD})\cdot P_{OLD}\cdot\Phi^{T}(t_{NEW},t_{OLD})+G\cdot Q\cdot G^{T}\ . \tag{4}$$

The Measurement Equations

We begin with the measurement equation. A vector of measurements $\vec{y}$ is a function of the state vectors, and is corrupted by noise,

$$\vec{y}=\vec{h}(\vec{x})+\vec{v} \tag{5}$$

where $\vec{v}$ is a vector of zero mean Gaussian measurement noise, with covariance R. The linearized form is

$$\vec{y}=H\cdot\vec{x}+\vec{v},\ H=\frac{\partial\vec{h}(\vec{x})}{\partial\vec{x}}\ . \tag{6}$$

The Update Equations

Kalman filtering can be done with analog computers, and this was done in early inertial navigation systems and in state estimation in industrial facilities. A continuously updated state vector estimate is defined by its differential equation,

$$\frac{d\vec{x}}{dt}=F(t)\cdot\vec{x}+K(t)\cdot\left(\vec{y}(t)-\vec{h}(\vec{x})\right)\ . \tag{7}$$

The Kalman gain matrix $K(t)$ is defined by minimizing the trace of the covariance matrix of the updated state vector,

$$\frac{dP}{dt}=F\cdot P+P\cdot F^{T}+G\cdot Q\cdot G^{T}-K\cdot R\cdot K^{T} \tag{8}$$

and is

$$K(t)=P(t)\cdot H^{T}(t)\cdot R^{-1}(t)\ . \tag{9}$$

The linearized update is

$$\vec{x}_{NEW}=\vec{x}_{EXT}+K\cdot\left(\vec{y}-H\cdot\vec{x}_{EXT}\right)\ . \tag{10}$$

The extrapolated and updated covariance are separate with discrete updates. The extrapolated covariance can be seen to be a linearized form of (8),

$$P_{EXT}=\Phi\cdot P_{OLD}\cdot\Phi^{T}+G\cdot Q\cdot G^{T}\ . \tag{11}$$

The updated covariance can be seen from (10) to be

$$P_{NEW}=(I-K\cdot H)\cdot P_{EXT}\cdot(I-K\cdot H)^{T}+K\cdot R\cdot K^{T}\ . \tag{12}$$

Using (12) to define the Kalman gain matrix K to minimize the trace of P_{NEW} results in

$$K=P_{EXT}\cdot H^{T}\cdot\left(H\cdot P_{EXT}\cdot H^{T}+R\right)^{-1}\ . \tag{13}$$

Substituting the Kalman gain matrix from (13) into the covariance equation (12) gives alternative forms, with some algebra, including the Matrix Inversion Lemma (see below):

$$P^{-1}{}_{NEW}=P^{-1}{}_{EXT}+H^{T}\cdot R^{-1}\cdot H\ . \tag{14}$$

One form that is obtained using the Matrix Inversion Lemma, $K=(I-K\cdot H)\cdot P_{EXT}$, is often given in the literature because of its simplicity and the resulting usefulness in some algebraic developments, but use of this simple form for computation in software almost universally results in failure of the Kalman filter because of numerical errors. The form in (12) is called the Joseph Stabilized Form and should always be used in first-cut implementations. Equation (9) also holds for the discrete case, but the Kalman gain must be available first to compute P_{NEW}. For deliverable software for deployed systems, a square root filter is far more robust than any of these equations.

The Discrete Kalman State Update as a Markov Process

We show the stability of the Kalman filter here. Equation (10) can be written as a Markov process update,

$$\vec{x}_{NEW}=\left(I-K\cdot H\right)\cdot\vec{x}_{EXT}+K\cdot\vec{y}\ . \tag{15}$$

The Markov transition matrix M is $\left(I-K\cdot H\right)$,

$$\begin{aligned} M&=\left(I-K\cdot H\right)=\left(I-P_{NEW}\cdot H^T\cdot R^{-1}\cdot H\right)\\ &=P_{NEW}\cdot P_{EXT}^{-1} \end{aligned} \tag{16}$$

Using (14) for P_{NEW} in (16) gives a form for M that is simple and well-conditioned numerically,

$$M=P_{NEW}\cdot P_{EXT}^{-1}=\left(I+P_{EXT}\cdot H^T\cdot R^{-1}\cdot H\right)^{-1}\ . \tag{17}$$

Here we will state a theorem that shows the even part of $M=P_{NEW}\cdot P_{EXT}^{-1}$ to be positive definite. We can add consideration of the state transition matrix,

$$\vec{x}_{EXT}=\Phi\cdot\vec{x}_{OLD} \tag{18}$$

to make an equation updating $\vec{x}_{NEW}$ from $\vec{x}_{OLD}$ if desired, but the key concern is showing stability properties of the Kalman filter, by looking at the eigenvalues of the Markov transition matrix. The eigenvalues of the state transition matrix are all one, and consideration of the eigenvalues of M as given by (17) suffices for that purpose.

THEOREM: When matrices A and B are symmetrical positive definite matrices, the even part of the matrix product $A\cdot B$ is positive definite.

PROOF: We examine the quadratic form

$$q=\vec{v}^T\cdot A\cdot B\cdot\vec{v} \tag{19}$$

which is equal to a quadratic form on the even part of $A\cdot B$. We select the vector $\vec{v}$ as the i^{th} right eigenvector of B,

$$\vec{v}=\vec{b}_i\ , \tag{20}$$

which, left-multiplied by B,

$$B\cdot\vec{b}_i=\left[\sum_j\vec{b}_j\cdot\lambda_{B,j}\cdot\vec{b}_j^T\right]\cdot\vec{b}_i=\lambda_{B,i}\cdot\vec{b}_i \tag{21}$$

is $\vec{b}_i$ scaled by the positive eigenvalue $\lambda_{B,i}$. This leads to an intermediate result for the quadratic form,

$$q=\lambda_{B,i}\cdot\vec{b}_i^T\cdot A\cdot\vec{b}_i\ . \tag{22}$$

Since A is positive definite, then the value q of the quadratic form is positive. This shows that q is positive whenever $\vec{v}$ is an eigenvector of either A or B. Since both sets of eigenvectors span the vector space, and q is a quadratic polynomial in each of the elements of the vector $\vec{v}$, then if there existed a vector $\vec{v}$ for which q was negative, there would be a vector $\vec{v}$ for which q is zero, which cannot exist because, by hypothesis, both A and B are full rank and positive definite.

Q.E.D.

We now show that all the eigenvalues of M as given by (16) do not exceed one: equation (14) shows that

$$\vec{v}^T \cdot P_{NEW}^{-1} \cdot \vec{v} \leq \vec{v}^T \cdot P_{EXT}^{-1} \cdot \vec{v} \tag{23}$$

for any vector $\vec{v}$. This means that the localization ellipsoid (see next Section) of the updated state vector is wholly contained within the localization ellipsoid of the extrapolated state vector before update. Since the variance of the state vector in any direction is nonincreasing in the update operation, no eigenvalue of M exceeds one.

The Mahalanobis Distance and the Localization Ellipsoid

The Mahalanobis distance, which is defined for k Gaussian random variables $\vec{x}$, with mean $\vec{\mu}$ and covariance S, is the dimensionless scalar

$$s = \sqrt{(\vec{x} - \vec{\mu})^T \cdot S^{-1} \cdot (\vec{x} - \vec{\mu})} \, . \tag{24}$$

We interpret the Mahalanobis distance by noting that the dimensionless vector $\vec{y}$ found from a vector $\vec{x}$ with zero mean Gaussian errors from mean $\vec{\mu}$ as

$$\vec{y} = S^{-1/2} \cdot (\vec{x} - \vec{\mu}) \tag{25}$$

where $S^{-1/2}$ is a Cholesky factor or any other matrix square root of S^{-1}, and $\vec{y}$ is a set of k Gaussian random variables of mean zero and variance one that are not correlated with one another. Thus, the Mahalanobis distance s is a measure, conceptually, of how many standard deviations that the vector $\vec{x}$ is from its mean $\vec{\mu}$. We also see that $s^2 = \vec{y}^T \cdot \vec{y}$ is a random variable with a chi-squared distribution with k degrees of freedom.

The concept of a localization ellipsoid follows from the multivariate Gaussian distribution,

$$\begin{array}{rcl} p(x_1, x_2, \ldots, x_k) & = & \dfrac{1}{\sqrt{(2\pi)^k \cdot |S|}} \cdot \exp\left(-\dfrac{1}{2} \cdot (\vec{x} - \vec{\mu})^T \cdot S^{-1} \cdot (\vec{x} - \vec{\mu})\right) \\ p(s) & = & \dfrac{1}{(2\pi)^{k/2}} \cdot \exp\left(-\dfrac{s^2}{2}\right) \text{ (with Jacobian determinant } |S|^{1/2}) \end{array} \tag{26}$$

where we have included $p(s)$, the probability density function of the Mahalanobis distance.

The presence of the Mahalanobis distance in the kernel of the probability density means that the commonly used measures of nearness to the mean, one-sigma limit, median and quartiles, probability of containment within a given Mahalanobis distance, etc. are all scaled from the same ellipsoid, defined as the locus of the vector $\vec{x}$ in the quadratic form

$$(\vec{x} - \vec{\mu})^T \cdot S^{-1} \cdot (\vec{x} - \vec{\mu}) = 1 \, . \tag{27}$$

We call the localization ellipsoid the surface or volume defined by the locus of $\vec{x}$ in (27).

Probability Distribution Functions

When looking at association of data to tracks, particularly in cases when k is large such as batch processing, the volume of the localization ellipsoid is an important consideration. The volume of a hypersphere of radius R is

$$V=\frac{\pi^{k/2}}{\Gamma\left(\frac{k}{2}+1\right)}\cdot R^k \tag{28}$$

where $\Gamma(x)$ is the gamma function, $\Gamma(n+1)=n!$. If we look at an ellipsoid with semiaxes of lengths equal to the square root of an eigenvalue of S, we have the principal axes of the ellipsoid defined by setting $s^2=k$, the mean of the chi-squared distribution. Then, the formula for the volume of this hyperellipsoid replaces each R in the R^k in (28) with the square root of an eigenvalue of the covariance matrix S and we have

$$V=\frac{\pi^{k/2}}{\Gamma\left(\frac{k}{2}+1\right)}\cdot|S|^{1/2}\,. \tag{29}$$

For $R=s$ we have the volume of a spheroid within Mahalanobis distance s as

$$V=\frac{\pi^{k/2}}{\Gamma\left(\frac{k}{2}+1\right)}\cdot s^k\,. \tag{30}$$

We will use the probability density function from (26) and the volume equation from (30) to find the probability distribution of data from its mean as follows. We begin by finding the surface area of the spheroid of (30) as its derivative with respect to s,

$$A=\frac{dV}{ds}=\frac{2\pi^{k/2}}{\Gamma\left(\frac{k}{2}\right)}\cdot s^{k-1} \tag{31}$$

and write the probability distribution function as

$$P(|x|\leq s)=\int_{s=0}^{s}\frac{1}{(2\pi)^{k/2}}\cdot\frac{2\pi^{k/2}}{\Gamma\left(\frac{k}{2}\right)}\cdot t^{k-1}\cdot\exp\left(-\frac{t^2}{2}\right)\cdot dt=\frac{1}{2^{k/2-1}\Gamma\left(\frac{k}{2}\right)}\cdot\int_{s=0}^{s}t^{k-1}\cdot\exp\left(-\frac{t^2}{2}\right)\cdot dt\,. \tag{32}$$

With the change of variable $t^2=u$, the integral in (32) is

$$P(|x|\le s)=\frac{1}{2^{\frac{k}{2}}\cdot\Gamma\left(\frac{k}{2}\right)}\cdot\int_{u=0}^{s^2} u^{\frac{k}{2}-1}\cdot\exp\left(-\frac{u}{2}\right)\cdot du\ . \tag{33}$$

We recognize the integral in (33) as the incomplete gamma function (DLMF 8.2.1),

$$\gamma(a,z)=\int_{t=0}^{z} t^{a-1}\cdot\exp(-t)\cdot dt\ . \tag{34}$$

The definition provided by (34) allows us to write the probability distribution function of (33) using the lower incomplete gamma function $\gamma(a,x)$ as

$$P(|x|\le s)=\frac{\gamma\left(\frac{k}{2},\frac{s^2}{2}\right)}{\Gamma\left(\frac{k}{2}\right)}\ . \tag{35}$$

The distribution of s^2 is, as expected, the chi-squared distribution with k degrees if freedom, an example of a special case of the Gamma distribution, specifically the Erlang distribution, which is a scaling generalization of the chi-squared distribution. The distribution function is available in most spreadsheets with proper use of the built-in function $\text{GAMMA.DIST}(x;\alpha;\beta;C)$ where C is a logical flag, 0 for calculating probability density and 1 for calculating probability distribution. The gamma distribution is a more general distribution using a shape parameter α and a rate parameter β,

$$\begin{aligned} f(x;\alpha,\beta) &= \frac{\beta^{\alpha}\cdot x^{\alpha-1}\cdot\exp(-\beta x)}{\Gamma(\alpha)} \\ F(x;\alpha,\beta) &= \int_0^x f(u;\alpha,\beta)\cdot du \quad . \\ &= \frac{\gamma(\alpha,\beta x)}{\Gamma(\alpha)} \end{aligned} \tag{36}$$

The probability density function is helpful in understanding decision thresholds for Kalman updates and sensor fusion. These are found by taking the derivative of the distribution function as given by (33) with respect to either s or s^2. In terms of the Mahalanobis distance s, the distribution function is

$$p(s)=\frac{s^{k-1}\cdot\exp\left(-\frac{s^2}{2}\right)}{2^{\frac{k}{2}-1}\cdot\Gamma\left(\frac{k}{2}\right)}\ . \tag{37}$$

The probability density function of the squared Mahalanobis distance $t=s^2$ is the chi-squared distribution

$$p(t)=\frac{t^{\frac{k}{2}-1}\cdot\exp\left(-\frac{t}{2}\right)}{2^{\frac{k}{2}}\cdot\Gamma\left(\frac{k}{2}\right)} . \tag{38}$$

The Innovations Sequence

The innovations sequence is a name commonly given to the Kalman filter error vector from (7) or (10),

$$\vec{e}=\vec{y}(t)-\vec{h}(\vec{x}) . \tag{39}$$

Analysis shows that, ideally, this quantity is mean zero and uncorrelated from update to update, which is how it was dubbed the “innovations” sequence. Ideally, its covariance is

$$E=Cov\{\vec{y}(t)-\vec{h}(\vec{x})\}=H\cdot P_{EXT}H+R \tag{40}$$

which gives us a handle on a quantity that we can use to determine whether a particular vector $\vec{y}$ is suitable for use in updating the track or not,

$$t=(\vec{y}(t)-\vec{h}(\vec{x}))^T\cdot E^{-1}\cdot(\vec{y}(t)-\vec{h}(\vec{x})) . \tag{41}$$

The quantity t, or some approximation of it, is used in nearly every practical measurement-to-track association algorithm.

Unbiased, Minimum Variance Data Fusion Simplified

Suppose that we have to estimates of a state variable $\vec{x}_1$ and $\vec{x}_2$ with random errors that are independent between the estimates. We want a linear unbiased, minimum variance estimate of $\vec{x}$. We first define a linear estimator,

$$\vec{x}_{NEW}=A\cdot\vec{x}_1+B\cdot\vec{x}_2 . \tag{42}$$

We define a scalar cost function

$$J=\mathrm{Tr}\{\mathrm{Cov}\{\vec{x}_{NEW}\}\} . \tag{43}$$

We need an expression for the covariance of the estimate,

$$P_{NEW}=\mathrm{Cov}\{\vec{x}_{NEW}\}=\mathrm{Exp}\{(\vec{x}_{NEW}-\vec{x})\cdot(\vec{x}_{NEW}-\vec{x})^T\} \tag{44}$$

from which we have

$$J=\mathrm{Tr}\{Cov\{\vec{x}_{NEW}\}\}=\mathrm{Tr}\{\mathrm{Exp}\{(\vec{x}_{NEW}-\vec{x})^T\cdot(\vec{x}_{NEW}-\vec{x})\}\} . \tag{45}$$

From (42) and the requirement that the estimator be unbiased, we have

$$A+B=I\ . \tag{46}$$

Using (42) in (45) gives us

$$\begin{aligned} J &= \operatorname{Exp}\left\{\left(A\cdot(\vec{x}_1-\vec{x})+B\cdot(\vec{x}_2-\vec{x})\right)^T\cdot\left(A\cdot(\vec{x}_1-\vec{x})+B\cdot(\vec{x}_2-\vec{x})\right)\right\} \\ &= \operatorname{Exp}\left\{(\vec{x}_1-\vec{x})^T\cdot A^T\cdot A\cdot(\vec{x}_1-\vec{x})+(\vec{x}_2-\vec{x})^T\cdot B^T\cdot B\cdot(\vec{x}_2-\vec{x})\right\} \end{aligned} \tag{47}$$

where we have used

$$P_{EXT}=\operatorname{Exp}\left\{\left((\vec{x}_1-\vec{x})^T\cdot A^T\cdot A\cdot(\vec{x}_1-\vec{x})+(\vec{x}_2-\vec{x})^T\cdot B^T\cdot B\cdot(\vec{x}_2-\vec{x})\right)\right\}\ . \tag{48}$$

To find the optimum, we take the gradient of J with respect to the matrix A and set it to zero:

$$\begin{aligned} \frac{\partial J}{\partial A} &= \operatorname{Exp}\left\{2A\cdot(\vec{x}_1-\vec{x})\cdot(\vec{x}_1-\vec{x})^T-2B\cdot(\vec{x}_2-\vec{x})\cdot(\vec{x}_2-\vec{x})^T\right\} \\ &= 2A\cdot P_1-2B\cdot P_2=0 \end{aligned}\ . \tag{49}$$

Combining (46) and (49) gives us, with the aid of the Matrix Inversion Lemma,

$$\begin{aligned} A &= (P_1+P_2)^{-1}\cdot P_2^{-1}=\left(P_1^{-1}+P_2^{-1}\right)^{-1}\cdot P_1^{-1} \\ B &= I-\left(P_1^{-1}+P_2^{-1}\right)^{-1}\cdot P_1^{-1}=\left(P_1^{-1}+P_2^{-1}\right)^{-1}\cdot P_2^{-1} \end{aligned}\ . \tag{50}$$

Evaluating (48) with (50) gives us

$$P_{NEW}^{-1}=P_1^{-1}+P_2^{-1}\ . \tag{51}$$

From (51) we see that

$$\vec{y}^T\cdot P_{NEW}^{-1}\cdot\vec{y}\le\vec{y}^T\cdot P_i^{-1}\cdot\vec{y},\ i=1,2 \tag{52}$$

so that the localization ellipsoid of the merged data is wholly contained within the localization ellipsoid of both contributing state vectors.

The derivation is trivially extended to multiple datasets.

State Vector and Covariance Matrix from Samples

Particle filters and unscented transformations produce examples of state vectors for a variety of values of purpose-defined “noise” driving a Markov process. Here we look at a generic example of finding estimates of a state vector and its covariance matrix from examples of data.

The Unweighted Case

We will begin with the maximum likelihood estimate for the unweighted case. The data is a set of vectors with additive Gaussian noise,

$$\vec{x}_i=\vec{x}+\vec{v}_i,\ \operatorname{Cov}\{\vec{v}_i\}=P,\ i=[1,2,\ldots,N]\ . \tag{53}$$

The likelihood function, or the probability density function, of the data given the mean, which is the truth data, and the covariance of the measurement vectors, is

$$p(\vec{x}_i|x,P)=\frac{1}{(2\pi|P|)^{N/2}}\cdot\exp\left(-\frac{1}{2}\sum_{i=1}^{N}(\vec{x}_i-\vec{x})^T\cdot P^{-1}\cdot(\vec{x}_i-\vec{x})\right). \tag{54}$$

The log likelihood function is

$$L(x,P)=-\frac{N}{2}\cdot\ln(2\pi)-\frac{N}{2}\cdot\ln(|P|)-\frac{1}{2}\sum_{i=1}^{N}(\vec{x}_i-\vec{x})^T\cdot P^{-1}\cdot(\vec{x}_i-\vec{x}). \tag{55}$$

To find the maximum likelihood estimate of the state vector $\vec{x}$ we take the gradient of the log likelihood function with respect to the state vector $\vec{x}$ and set that to zero and solve for $\vec{x}$,

$$\frac{\partial L}{\partial\vec{x}}=\sum_{i=1}^{N}P^{-1}\cdot(\vec{x}_i-\vec{x})=P^{-1}\cdot\sum_{i=1}^{N}(\vec{x}_i-\vec{x})=\vec{0}. \tag{56}$$

The maximum likelihood estimate of the state vector is simply the sample mean,

$$\vec{x}_{MLE}=\frac{1}{N}\sum_{i=1}^{N}\vec{x}_i. \tag{57}$$

We use the classical identities of matrix gradients (76) to take the gradient of the log likelihood function with respect to P^{-1},

$$\frac{\partial L}{\partial P^{-1}}=\frac{N}{2}\cdot P-\frac{1}{2}\sum_{i=1}^{N}(\vec{x}_i-x)\cdot(\vec{x}_i-x)^T=0. \tag{58}$$

Note that this equation for the covariance matrix P also includes the “truth” information for the state vector, $\vec{x}$, so that the solution for the covariance matrix P includes unavailable data,

$$P_{UHOH}=\frac{1}{N}\cdot\sum_{i=1}^{N}(\vec{x}_i-x)\cdot(\vec{x}_i-x)^T=\frac{1}{N}\cdot\left(\sum_{i=1}^{N}\vec{x}_i\cdot\vec{x}_i^T-N\cdot\vec{x}\cdot\vec{x}^T\right). \tag{59}$$

To deal with the unavailability of a “pure” maximum likelihood estimate for the covariance matrix P, we use the maximum likelihood estimate of the state vector in place of the truth data. The result is

$$P_{MLE}=\frac{1}{N-1}\cdot\left(\sum_{i}\vec{x}_i\cdot\vec{x}_i^T-N\cdot\vec{x}_{MLE}\cdot\vec{x}_{MLE}^T\right). \tag{60}$$

where we have noted that using $\vec{x}_{EST}$ in place of $\vec{x}$ in the sum introduces a multiplicative bias of $(1-1/N)$ because $\vec{x}_{MLE}$ is correlated with all of the $\vec{x}_i$, and we remove this bias by dividing by the multiplicative bias, which results in dividing the sum of outer products by $N-1$ instead of N.

The Weighted case

The methods used in some Chapman-Kolmogorov related estimates require weighted averages. The weighted average is, explicitly,

$$\vec{x}_{WAV}=\frac{1}{\sum_{i=1}^{N} w_i}\cdot\sum_{i=1}^{N} w_i\cdot\vec{x}_i\ . \tag{61}$$

We will use the weighted average as the state vector estimate. We require the covariance of this estimate. The covariance becomes apparent when we use the definition of the data given in (53) in the equation (61) for the weighted average,

$$\vec{x}_{WAV}=\vec{x}+\frac{1}{\sum_{i=1}^{N} w_i}\cdot\sum_{i=1}^{N} w_i\cdot\vec{v}_i\ . \tag{62}$$

The covariance of $\vec{x}_{WAV}-\vec{x}$ is

$$\text{Cov}\{\vec{x}_{WAV}-\vec{x}\}=\frac{\sum_{i=1}^{N} w_i^2}{\left(\sum_{i=1}^{N} w_i\right)^2}\cdot R\ . \tag{63}$$

We have our best estimate of R, divided by N, from (60), from which the recommended form is

$$P_{WAV}=\frac{\sum_{i=1}^{N} w_i^2}{\left(\sum_{i=1}^{N} w_i\right)^2}\cdot\frac{N}{N-1}\cdot\left(\sum_{i=1}^{N}\vec{x}_i\cdot\vec{x}_i^T-N\cdot\vec{x}_{NEW}\cdot\vec{x}_{NEW}^T\right)\ . \tag{64}$$

Theorem:

$$\frac{1}{N}\leq\frac{\sum_{i=1}^{N} w_i^2}{\left(\sum_{i=1}^{N} w_i\right)^2}\leq 1,\ \text{when}\ w_i\geq 0\ \text{for}\ 1\leq i\leq N\ . \tag{65}$$

The Cauchy-Schwarz inequality is

$$\left(\vec{u}^T\cdot\vec{v}\right)^2\leq\left(\vec{u}^T\cdot\vec{u}\right)\cdot\left(\vec{v}\cdot\vec{v}\right)\ . \tag{66}$$

We see from using $\vec{u}=\vec{w}$, $\vec{v}=\vec{1}$ that the Cauchy-Schwarz inequality shows that the ratio is greater than $1/N$. We prove the ratio is less than or equal to 1 by writing the second inequality in (65) as

$$\sum_{i=1}^{N} w_i^2 \leq \sum_{i=1}^{N} \sum_{j=1}^{N} w_i \cdot w_j = \sum_{i=1}^{N} w_i^2 + \sum_{i=1}^{N} \sum_{\substack{j=1 \\ j \neq i}}^{N} w_i \cdot w_j \qquad (67)$$

which is always true when all the w_i are nonnegative (or, are all of the same sign or zero).

Q.E.D.

We note that we have equality with 1 when only one of the w_i . is nonzero, and equality with $1/N$ when all of the w_i are equal. Since efficiency considerations require keeping N as small as possible in particle and unscented trackers, this may mean that there may be some accuracy disadvantage relative to Kalman filter trackers that use statistically efficient Kalman updates.

Matrix Inversion Lemma

The Matrix Inversion Lemma is useful in algebraic manipulation in estimation theory and other areas. Statement of the lemma:

$$\left(A - B \cdot D^{-1} \cdot C\right)^{-1} = A^{-1} + A^{-1} \cdot B \cdot \left(D - C \cdot A^{-1} \cdot B\right)^{-1} \cdot C \cdot A^{-1} \, . \qquad (68)$$

Proof follows from construction of an augmented matrix and solving for its inverse by submatrices,

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} \cdot \begin{bmatrix} E & F \\ G & H \end{bmatrix} = \begin{bmatrix} A \cdot E + B \cdot G & A \cdot F + B \cdot H \\ C \cdot E + D \cdot G & C \cdot F + D \cdot H \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} \qquad (69)$$

and

$$\begin{bmatrix} E & F \\ G & H \end{bmatrix} \cdot \begin{bmatrix} A & B \\ C & D \end{bmatrix} = \begin{bmatrix} E \cdot A + F \cdot C & E \cdot B + F \cdot D \\ G \cdot A + H \cdot C & G \cdot B + H \cdot D \end{bmatrix} = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix} . \qquad (70)$$

From (69) and (70) the submatrices E , F , G and H of the augmented matrix are

$$\begin{aligned} E &= \left(A - B \cdot D^{-1} \cdot C\right)^{-1} \\ &= A^{-1} + A^{-1} \cdot B \cdot \left(D - C \cdot A^{-1} \cdot B\right)^{-1} \cdot C \cdot A^{-1} \end{aligned} \qquad (71)$$

$$\begin{aligned} F &= -\left(A - B \cdot D^{-1} \cdot C\right) \cdot B \cdot D^{-1} \\ &= -A^{-1} \cdot B \cdot \left(D - C \cdot A^{-1} \cdot B\right)^{-1} \end{aligned} \qquad (72)$$

$$\begin{aligned} G &= -D^{-1} \cdot C \cdot \left(A - B \cdot D^{-1} \cdot C\right)^{-1} \\ &= -\left(D - C \cdot A^{-1} \cdot B\right)^{-1} \cdot C \cdot A^{-1} \end{aligned} \qquad (73)$$

and

$$\begin{aligned} H &= D^{-1} + D^{-1} \cdot C \cdot \left(A - B \cdot D^{-1} \cdot C \right)^{-1} \cdot B \cdot D^{-1} \\ &= \left(D - C \cdot A^{-1} \cdot B \right)^{-1} \end{aligned} . \tag{74}$$

The lemma follows from either (71) or (74). Alternative formats follow from (72) and (73). Validity requires that all the matrix products and the inverses exist, including the augmented matrices in (69) and (70). Applicability to Kalman gain identities is via

$$A = P^{-1}{}_{EXT},\ \ B = H^T,\ \ C = H,\ \ D = -R\ . \tag{75}$$

Matrix Gradients

We repeat here some of the most useful classical identities (Gelb, Applied Optimal Estimation, p. 23)

$$\begin{array}{rcl} \dfrac{\partial}{\partial A}\text{trace}[A] & = & I \\ \dfrac{\partial}{\partial A}\text{trace}[B \cdot A \cdot C] & = & B^T \cdot C^T \\ \dfrac{\partial}{\partial A}\text{trace}[A \cdot B \cdot A^T] & = & A \cdot \left(B + B^T \right) \\ \dfrac{\partial}{\partial A}\text{trace}\left(\exp(A) \right) & = & \exp\left(A^T \right) \\ \dfrac{\partial}{\partial A}\left| B \cdot A \cdot C \right| & = & \left| B \cdot A \cdot C \right| \cdot A^{-T} \end{array} \quad . \tag{76}$$

Note that quadratic forms that are scalars are equal to the trace of a 1×1 matrix, so the gradients of a trace also apply to scalar quadratic forms. And, the determinant of a 1×1 matrix is a scalar, so that the last line of (76) can be applied to a scalar quadratic form.

Bibliography

Reference and Tutorial

Blackman, S. S. *Multiple Target Tracking with Radar Applications*, Artech House, Norwood, MA ISBN 978-0890061794 (1986). Detailed development of the MHT

Blackman, S. S., *Multiple Hypothesis Tracking For Multiple Target Tracking*, IEEE A&E Systems Magazine vol. 19, NO. 1 January 2004, pp 5-18.

Bar-Shalom, Y, Wilett, P. K., and Tian, X, Tracking and Data Fusion, A Handbook of Algorithms, ISBN 978-0964831278 (2011).

Koch, W., *Tracking and Sensor Data Fusion: Methodological Framework and Selected Applications*, Springer, ISBN 978-3642392702 (2013).

Blackman, S. S., Popoli, R., *Design and Analysis of Modern Tracking Systems*, Artech House, ISBN 978-1580530064 (1999).

Bierman, G. J., *Factorization methods for discrete sequential estimation*, Academic Press, ISBN 978-0120973507 (1977). Presents SRIF, UDUT and Potter square root filters.

Beard, J. K.,Maritime Situational Awareness, tutorial given at *IEEE/ISIF FUSION2005*, July 2005, Philadelphia. Also a bonus topic, Interactive Multiple Models (IMM) in trackers. Links: Tutorial, IMM.

Beard, J. K., An Efficient, Reliable Approach for Hydroacoustical Bearings Only Target Motion Analysis, *Proceedings of the IEEE Conference on Decision and Control*, 14 December 1984, PP 1251-1256. Click here for paper. Click here for the figures. DOI: 10.1109/CDC.1984.272220

NOTE: Valuable references are found by searching for works of Oliver Drummond, Sam Blackman, Mike Athans, Fred Daum, and others. Industry-sponsored research has priorities in publishing results that make such contributions rarer and occasionally difficult to locate, and a comprehensive technology survey may require more diligence than is often anticipated.

On the Internet

References on ellipsoid volume:

- US Naval Academy: https://www.usna.edu/Users/physics/mungan/_files/documents/Scholarship/HypersphereVolume.pdf
- NASA: https://spacemath.gsfc.nasa.gov/universe/6Page89.pdf

Historical

Astronautical guidance, R. H. Battin, McGraw-hill (1964) documents Potter's introduction of the square root filter

Filtering for stochastic process with applications to guidance, R. S. Bucy and P. D. Joseph, John Wiley & sons, NY (1968) (Ch. 16, "Numerical Errors," presents the Joseph stabilized form)

Agee, W. S., And Turner, R. H., "*Triangular decomposition of a positive definite matrix plus a symmetric dyad with application to Kalman filter*," white sands missile range tech. Rep. No. 38 (1972)

Carlson, N. A., "*Fast triangular factorization of the square root filter*," AIAA J. 11, no. 9, pp 1259-1265 (1973)

Solving least squares problems, C. L. Lawson and R. J. Hanson, prentice-hall, Englewood cliffs, NJ (1974)

Applied Optimal Estimation, A. Gelb, Ed., MIT Press (1974) (printings until at least 1988).