Contract No.:

This manuscript has been authored by Battelle Savannah River Alliance (BSRA), LLC under Contract No. 89303321CEM000080 with the U.S. Department of Energy (DOE) Office of Environmental Management (EM).

Investigating the Application of Kalman Filters for Real-time Accountancy in Fusion Fuel Cycles

H. B. Flynn, George Larsen
Savannah River National Laboratory, Aiken, SC 29808, United States

ABSTRACT

Tritium accountancy in the fusion fuel cycle is a significant concern for the operation of commercial devices. It is expected that a limit on the maximum amount of tritium inventory in the system will be implemented, meaning that accountancy of the tritium inventory in the fuel cycle will need to be as accurate as possible. This is difficult since not all locations along the fuel cycle can benefit from the implementation of a tritium accountancy sensor and measurements will inherently contain error. Commercial fusion plants will also operate continuously, challenging current accountancy techniques that rely on static processes in well-controlled environments. In this paper a simple fusion fuel cycle concept is defined and the tritium inventory over time of each component is modelled using the Euler Approximation of a series of differential Equations in Python. This information is then used to simulate sensor measurements at specific points on the fuel cycle and then passed through a Kalman Filter (KF) to improve the accuracy of the true measurement of the sensors.

1. Introduction

Tritium accountancy is the process of accurately measuring and reporting the amount of tritium contained within a facility, including additions and losses [1]. This is done on a recurring basis to ensure that the facility is meeting environmental and safety regulations since tritium is radioactive. Fusion power plants will require the handling of unprecedented amounts of tritium in extreme temperatures and environments and contain complex cooling, piping, and pumping systems [2]. The tritium inventory will be lost through burnup, implantation, and diffusion while simultaneously tritium inventory will be gained through breeding. This constant gain-loss dynamic makes accurate accountancy difficult, but further complications arise from the continuous operation of a fusion power plant. Continuous operation will challenge current accountancy techniques that rely on static processes in well-controlled environments [3].

One way to account for the tritium in the fuel cycle is by strategically placing sensors at the outlet of certain components. The sensors will record the tritium inventory of the component as a function of time and allow for accounting of the tritium inventory. However, sensors are noisy, especially placed within a system that could have high vibrations, potential ground loop issues, radiofrequency (RF) generating components, etc. The sensor information is also expected to accumulate system uncertainties as they continue to measure over time [4]. Thus, it is important to improve the reliability of data collected from continuous sensors if they are to serve as platforms for tritium accountancy. This improvement can be made by applying techniques that minimize uncertainty and biases surrounding noisy data.

2. Methods

The Kalman Filter (KF), a state prediction algorithm, is a well-known method of taking noisy sensor data and its error and creating estimates of the true value and the true value error [5]. This is a type of model-based data smoothing routine that is applied in many robotics applications [6, 7], radar [8], water run-off [9], and autonomous vehicles [10, 11]. However, its application toward tritium inventory accountancy has not been investigated. The KF has the potential to be a useful technique since tritium accountancy will depend on the sensor data during the continuous operation of a fusion device. The KF, when paired with a model that represents the system, can take noisy sensor data and more accurately track tritium inventory.

The use of the KF for tritium inventory accountancy is investigated in this paper. Since the KF depends on an accurate model of the system, the development of a Python-based tritium fuel cycle model is described in Section 3 [2]. Section 4 explains how the calculated inventory data is used to simulate sensor measurements at specific points along the fuel cycle by applying a Gaussian or Normal distribution error. The assumption of normal error is made based on prior applications of sensors in Kalman Filter applications [12, 13]. The normal distribution also places the true value or model value at the mean. Section 5 describes in greater detail how the sensor measurements and their error are used to make estimates from a Kalman Filter. Finally,

Section 6 will discuss the application of the KF for tritium accountancy in a fusion fuel cycle.

3. Tritium Inventory Model

The fuel cycle utilized herein to develop a model was described by M. A. Abdou et. al 1985. A diagram depicting each component and its inlet flow is shown in Figure 1. The fusion device has a tritium burn up rate, $\dot{N}^- = 500\ g/day$, which equates roughly to a $3\ GW$[2, 14] fusion power plant, assuming $1\ GW$ per 170 of burn up. It is worth noting that an actual fusion fuel cycle may have components and additional complexities that will extend past the model described here. However, for the purpose of this initial paper the simple fuel cycle model will be sufficient to show the benefits of applying the KF for tritium accountancy.

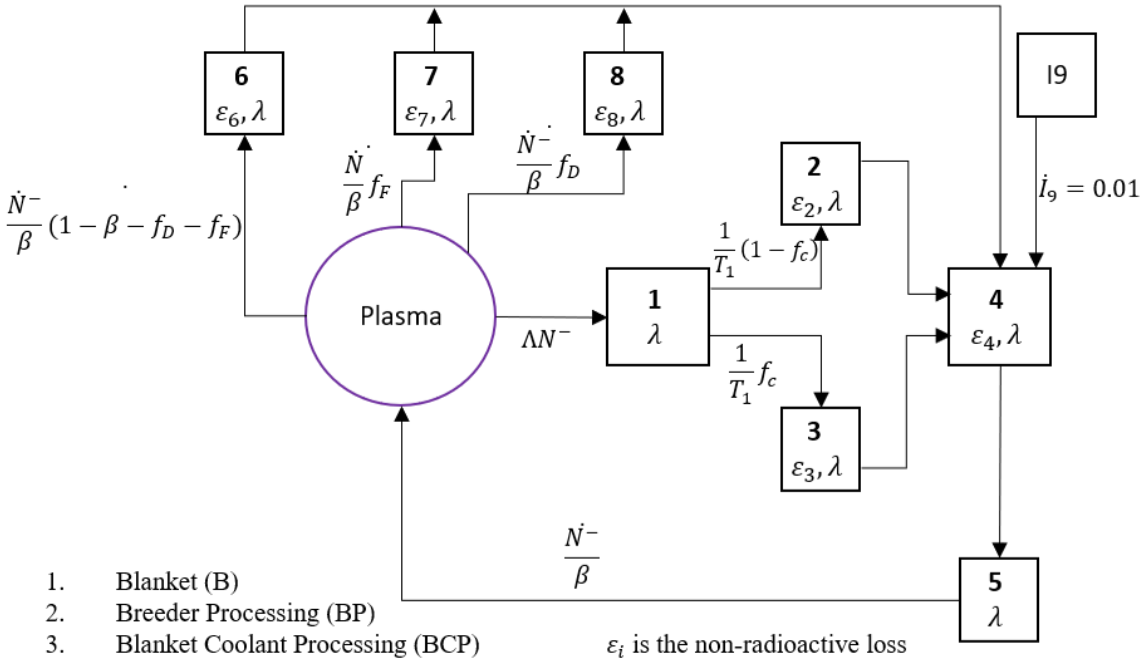
3.1 Python Model

The Python model imports information about the fuel cycle from a user written dictionary and creates the coefficient matrix, $\boldsymbol{C_{ij}}$, in Equation 1. $\boldsymbol{C_{ij}}$ has the shape $c \times c$ where c is the number of components modelled in the fuel cycle and contains the relevant contributing flow and loss terms for each component. The matrix determines what component inventory flows into component i from component j. The change in inventory in the i component over time is then

$$\frac{d\boldsymbol{I_i}}{dt} = \boldsymbol{C_{ij}} * \boldsymbol{I_j} + \boldsymbol{S_i} \qquad \text{Equation (1)}$$

$\boldsymbol{I_j}$ is the inventory from all components that contribute or reduce the inventory to $\boldsymbol{I_i}$, where j is an integer value $[1, 2, \ldots, c]$ and $\boldsymbol{S_i}$ is a vector containing the tritium source contribution from the plasma vessel and water detritiation. Only those components directly adjacent to the Plasma vessel (5, 6, 7, 8, and 1) and water detritiation (4) will have contribution from $\boldsymbol{S_i}$. An example of how Equation (1) would look for Block 4 (FCIS) is provided by Equation (2):

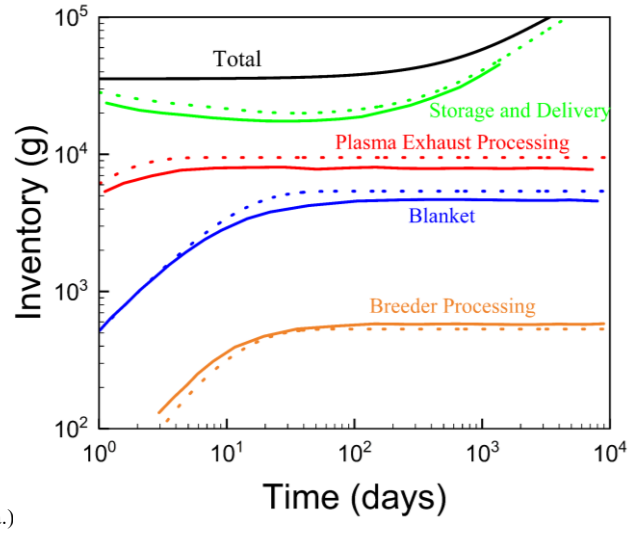
$$\frac{dI_4}{dt} = a_4 I_1 + b_4 I_2 + c_4 I_3 + d_4 I_4 + e_4 I_5 + f_4 I_6 + g_4 I_7 + h_4 I_8 + S_4 \qquad \text{Equation (2)}$$



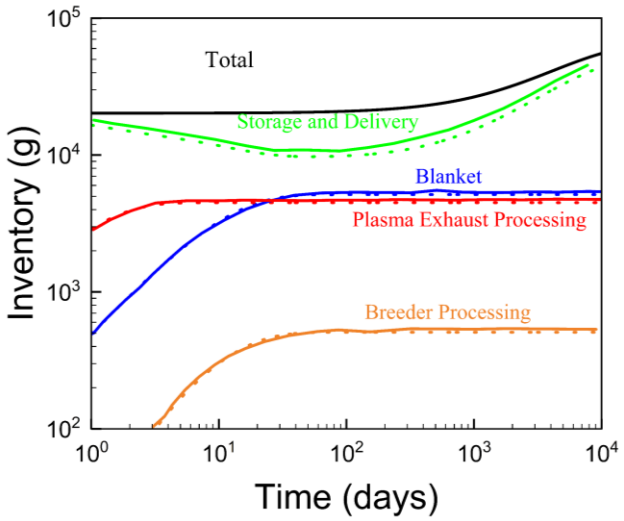
1. Blanket (B)
2. Breeder Processing (BP)
3. Blanket Coolant Processing (BCP)
4. Fuel Cleanup and Isotope Separation (FCIS)
5. Storage and Delivery (SD)
6. Plasma Exhaust Processing (PEP)
7. Divertor Coolant Processing (DCP)
8. First-Wall Coolant Processing (FWCP)
9. I9 = Water Detritiation

ε_i is the non-radioactive loss
λ is the tritium decay constant
$\dot{N}^-$ is the rate of tritium burn up
β is the tritium fractional burnup
Λ is the tritium breeding ratio
f is the tritium fractional leakage
T_i is the mean residence time

Figure 1. *Model of the fusion fuel cycle [2]*



a.)



b.)

Figure 2. a.) *Is Case 1 from with* $\beta = 0.05$ *and* $\Lambda = 1.08$. b.) *Case 3 from with* $\beta = 0.1$ *and* $\Lambda = 1.03$. *The solid lines are digitized data from the plots found in Abdou, et. al. and the dashed lines are from the Python fusion fuel cycle model in this paper [2].*

where $a_4 = 0$, $b_4 = \frac{1}{T_2}$, $c_4 = \frac{1}{T_3}$, $d_4 = \frac{(1-\varepsilon_4)}{T_4} + \lambda$, $e_4 = 0$, $f_4 = \frac{1}{T_6}$, $g_4 = \frac{1}{T_7}$, $h_4 = \frac{1}{T_8}$, and $S_4 = \dot{I}_9$. I_j is the component j inventory, T_i is the mean residence time of tritium in component i, and ε_i is the nonradioactive tritium loss fraction of component i. The component contribution to I_4 is from $I_2, I_3, I_6, I_7, I_8, and\ I_9$, which is reflected in Equation (2) once the coefficient values are inserted.

Once each of the component inventory equations are built, the Euler Method is used to solve for each component inventory over a period of time. The Euler Method uses the previous inventory to calculate the current inventory using a time step, Δt. A $\Delta t = 0.05\ days$ was chosen for the initial analyses of the inventories. At $t = 0$ all $I_i[t] = 0$ except $I_5[0] = 35{,}500\ grams$, which is the initial inventory introduced to the fuel cycle.

To test the model's accuracy, figures were generated showing the inventory as a function of time for two of the cases in Abdou M. A. et. al 1985. These two figures were then compared to the original digitized data from the paper. Figure 2 shows the inventories as a function of time for Case 1 (a) and for Case 3 (b). In each case the final time was $1 \times 10^4\ days$, and for Case 1 the initial inventory was 35.5 kg and for Case 3 the initial inventory was 22.1 kg. Both images displayed in this paper show a near-to equal comparison to the 1985 data.

The comparison with Abdou M. A. et. al 1985 demonstrates that the Python tritium inventory system model is a good representation of a fusion fuel cycle and is also sufficient to demonstrate the capabilities of the Kalman Filter, which incorporates the system model equations and sensor data to predict the inventory at the next time step.

4. Simulated Senor Measurements

Sensors provide a measurement of the inventory at a specific location, but the reported results will differ from the actual true values due to the inherent errors in the detection mechanisms. To simulate a sensor, an error distribution is introduced to the true value, which is the model calculated inventory. This is done by choosing a location on the fuel cycle, as shown by the red circles in Figure 3, and adding a Gaussian distribution ($\mathcal{N}(\mu, \sigma^2)$) as noise to the calculated inventory as shown in Equation (3) [10, 12]. In Equation (3), μ is the mean and equal to 0, σ is the variance from the mean, and σ^2 is the covariance.

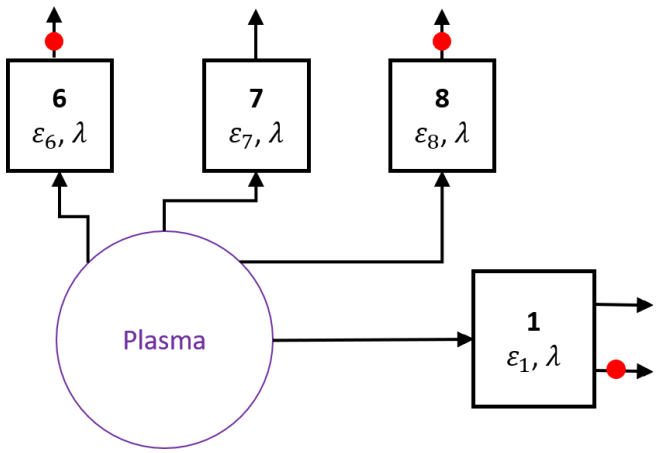


Figure 3. *Location of the sensors (red) that are 'measuring' inventory in the fuel cycle*

For this paper, the blanket inventory (I_1) and sensor measurement will be used to demonstrate the benefits of the KF. However, Figure 4 shows the sensor measurements and inventories for plasma exhaust processing (I_6) and first wall coolant processing (I_8) as well. The Gaussian distribution of the random noise is generated using python's Numpy package. For the function input the mean (μ) is centered at 0 and the sensor variance (σ_{Sen}) is equal to an arbitrarily large value, $\sigma_{Sen} = 100$. The required effect is to demonstrate a sensor with a large variance from the mean, simulating high noise.

$$I_{sensor_i,t} = I_i + \mathcal{N}(\mu, \sigma^2) \qquad Equation\ 3$$

Figure 4 is $I_{model_1,t}$ (blue), $I_{model_6,t}$ (red), and $I_{model_8,t}$ (black) plotted against their simulated sensor measurements (dots) generated using Equation (3). $I_{sensors_1,t}$ has a standard deviation of $\sigma_{StDev} \approx 411$. The sensor measurements, as expected, appear as a normal distribution around the inventory values.

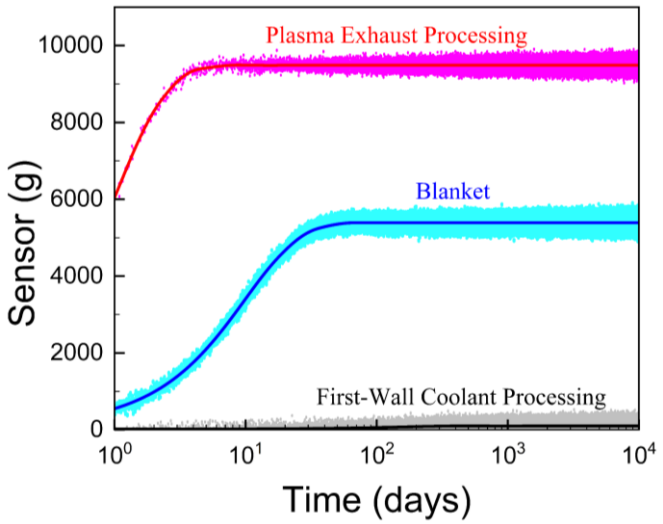


Figure 4. $I_4[t]$ (red) and $I_5[t]$ (blue) plotted against simulated sensor measurements (dots) using Equation 3.

5. The Kalman Filter

The Kalman Filter or Linear Kalman Filter is a state estimation filter, meaning that given an initial state of a system and information about the physical equations of the system one can predict the state of the system over some time or space. The Kalman Filter can be broken down into four stages: Prediction, Kalman Gain (KG), Correction, and Update stage.

The Prediction stage can be described using Equation (4) and Equation (5).

$$\boldsymbol{I}_{t_p} = \boldsymbol{A}_{t-1}\boldsymbol{I}_{t-1} + \boldsymbol{B}_{t-1}u_{t-1} \qquad \text{Equation (4)}$$

$$\boldsymbol{P}_{t_p} = \boldsymbol{A}_{t-1}\boldsymbol{P}_{t-1}\boldsymbol{A}_{t-1}^T + \boldsymbol{Q}_t \qquad \text{Equation (5)}$$

Equation (4) calculates the next predicted inventory state ($\boldsymbol{I}_{t_p}$) from the previous inventory state ($\boldsymbol{I}_{t-1}$) and any control inputs (u_{t-1}). The inventory state matrix is of the shape $N \times 1$, where N is the number of states to solve for. As an example, if the system of interest was a car moving in a 2D plane the kinematic state matrix of the system would either be position (1×1) or position and velocity (2×1). The control in this situation would be when the velocity is not constant and there is some acceleration. This can become increasing complex when you add 3D position (x, y, z), which each have a velocity contribution ($\dot{x}, \dot{y}, \dot{z}$), and angular or rotational contribution (θ, φ).

For the fusion fuel cycle of interest, the state matrix is a 2×1 matrix where the first row is inventory and the second row is inventory flow or the tritium source term, S_i. The transformation matrix **A** has the shape $N \times N$ and contains information about the model. The transformation matrix **B** has the shape $1 \times N$ and describes how the control, u_{t-1}, changes the state from $t-1$ to t. For this system the control is set to zero. Table 1 displays the values used to calculate the state prediction in Equation (4).

Table 1. Values used to calculate the covariance prediction

	P_{t-1}		Q_t	
$n = 1$	σ_I^2	0	30000	0
$n = 2$	0	σ_I^2	0	30000

Equation (5) calculates the next predicted covariance ($\boldsymbol{P}_{t_p}$) from the previous covariance matrix ($\boldsymbol{P}_{t-1}$) and $\boldsymbol{Q}_t$, which is a covariance matrix that tells the Kalman Filter how reliable the predictions are. For example, if the noise on the sensor is large and $\boldsymbol{Q}_t$ is low (high reliability in the prediction) then the Kalman Gain (KG) will converge quickly to zero, which implies the estimations are stable. This will cause the Kalman Filter to weight the predictions more heavily, resulting in an inaccurate filter calculation. This is because the estimations are not stable early in the filtering process; therefore, $\boldsymbol{Q}_t$ must be adjusted accordingly to the expected error in the sensor data. Table 2 displays the missing values used to calculate the state prediction in Equation 5. The covariance, σ_I^2, will be equal to the square of the error in the initial inventory.

Table 2. Values used to calculate the inventory prediction

	I_0	I_{t-1}	A_{t-1}		u_{t-1}	B_{t-1}
$n=1$	I_0	I_{t_0}	$\sum_{i=0} C_{ij}\Delta t$	Δt	0	0
$n=2$	S_i	S_i	0	1		1

The KG stage is described using Equation (6).

$$K_t = \left(\boldsymbol{P}_{t_p}\boldsymbol{H}_t^T\right)\left[\boldsymbol{H}_t\boldsymbol{P}_{t_p}\boldsymbol{H}_t^T + R_t\right]^{-1} \quad \text{Equation (6)}$$

The KG (K_t) is a value between 0 and 1 and is calculated using the predicted covariance matrix ($\boldsymbol{P}_{t_p}$), the observation/measurement matrix ($\boldsymbol{H}_t$), and the measurement covariance (R_t). KG is a weight that tells the filter whether to trust the measurement or the prediction more. As KG approaches 0 the predictions become more stable, and the measurements are less accurate. The inverse is true as well. Table 3 displays the values used to calculate K_t.

Table 3. Values used to calculate the Kalman Gain

	H_t		R_t
$n=1$	1	0	σ_{stDev}^2

The correction stage is described by Equations (7) and Equation (8).

$$\boldsymbol{I}_t = \boldsymbol{I}_{t_p} + K_t\left(y_t - \boldsymbol{H}_t\boldsymbol{I}_{t_p}\right) \quad \text{Equation (7)}$$

$$\boldsymbol{P}_t = (\boldsymbol{U} - K_t\boldsymbol{H}_t)\boldsymbol{P}_t \quad \text{Equation (8)}$$

Equation (7) and (8) calculate the corrected intensity state matrix ($\boldsymbol{I}_t$) and the corrected coherence matrix ($\boldsymbol{P}_t$). Equation 7 uses the sensor measurement at time t (y_t) and Equation (8) uses the identity matrix ($\boldsymbol{U}$). For the system of interest $y_t = I_{sensor_i,t}$. Figure 5 is the corrected Kalman Filter intensity values (red) retrieved from row one of the intensity state matrix plotted against the sensor measurement (cyan) and the model intensity calculation (blue).

Finally, the update stage is performed and is described by Equations (9) and (10). This stage is focused on updating the values used in the prediction stage. Once the values are updated another iteration of the stages is performed.

$$\boldsymbol{I}_{t-1} = \boldsymbol{I}_t \quad \text{Equation (9)}$$

$$\boldsymbol{P}_{t-1} = \boldsymbol{P}_t \quad \text{Equation (10)}$$

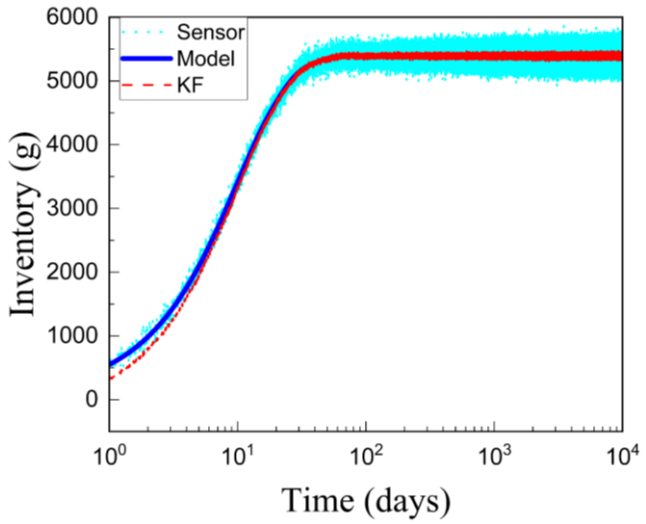


Figure 5. $I_{Sensor_1,t}$ (blue) or the Blanket component plotted against its simulated sensor measurements (dots) generated using Equation 3. The Kalman Filter prediction calculations are show as a red dotted line.

5.1 Linearization

In Figure (5) a discrepancy between the Kalman Filter prediction and the model calculated value is visible early in the trace. This discrepancy is mainly caused by the non-linearity of the system. As the Kalman Filter predictions become more stable and the system becomes more linear the Kalman Filter values are more consistent with the model calculated values. This is demonstrated by the flatter, more-linear portion near the end of the plot.

One way of testing a curve or systems' linearity is to process it through a Gaussian. If the resulting curve is a Gaussian, then the system is linear. On the other hand, if it is anything but a Gaussian, then the system is non-linear. Many systems found in the real world are non-linear. Consequently, there are modified filters such as the Extended Kalman Filter (EKG) [15] and the Unscented Kalman Filter (UKG) [16] that deal with this non-linearity. The EKG uses Taylor expansion Jacobians to linearize the system, which can be complex and difficult to implement. The UKG utilizes a random sampling technique from an arbitrary Gaussian distribution applied to the system to create a Gaussian output and is considered simpler to use than the EKG.

For this paper a different approach will be taken to address the non-linearity of the system instead of applying an adjusted filter. Since the calculated inventory from the model is continuous over a period, and controls would more than likely require the tritium to be checked after a couple days of

operation, the data can be segmented. The idea is that if the data is segmented into three-day intervals these segmentations would be near to linear enough for the Kalman Filter.

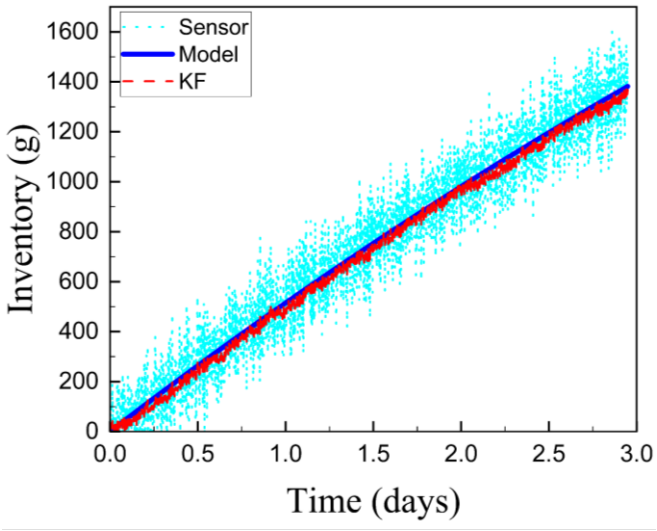


Figure 6. The segmented 3-day $I_{Sensor_1,t}$ (blue) or the Blanket component plotted against its simulated sensor measurements (dots) generated using Equation 3. The Kalman Filter prediction calculations are show as a red dotted line.

Figure 6 is a plot of the first three days for the modeled blanket inventory (I_1) graphed against the sensor measurement inventory (I_{sensor_1}) and the Kalman Filter predictions for the data. As can be seen in the figure, the Kalman Filter predictions (red) are very near the model values. This is an indication that the segmentation will work for applying the Kalman Filter to this data set. Section 5.2 will describe the quantification of accuracy of the Kalman Filter.

5.2 Accuracy using χ^2 and σ_{KF}

Not explicitly stated in the beginning of Section 5 is the measurement stage, which would occur prior to the correction stage. The measurement stage for the system of interest in this paper is defined by Equation (3). However, Equation (11) is a more comprehensive way of addressing the measurement stage for multiple systems.

$$y_t = \boldsymbol{H} \boldsymbol{I}_{t_p} + \nu_t \qquad \text{Equation (11)}$$

Where $\nu_t = \mathcal{N}(\mu, \sigma^2)$ and is called the measurement noise. Depending on what is measured, the values y_t and ν_t can be matrices. For this system $\boldsymbol{H}_t = [1,0]$ because only the inventory is measured and not the inventory flow, implying $\boldsymbol{H}_t \boldsymbol{I}_{t_p} = [1,0]\begin{bmatrix} I_t \\ i_t \end{bmatrix} = [I_t]$. Thus, the inventory measurement, measurement noise, and measurement covariance are scalar quantities. This is significant to understand when calculating the measurement-prediction covariance for χ^2 in Equation (12).

$$\chi^2 = \nu_t \left[\boldsymbol{H}_t \boldsymbol{P}_{t_p} \boldsymbol{H}_t^T + R_t \right]^{-1} \nu_t^T \qquad \text{Equation (12)}$$

The measurement-prediction covariance is the center bracketed portion in Equation (12) and is also the denominator in Equation (6). Figure 7 shows the χ^2 value plotted against the traces from Figure 6. The χ^2 quantity indicates how similar the Kalman Prediction is to the measurement. The lower the χ^2 value the closer it is to the sensor measurement (cyan) and the larger it is the closer the Kalman Filter prediction is to the model calculation or mean.

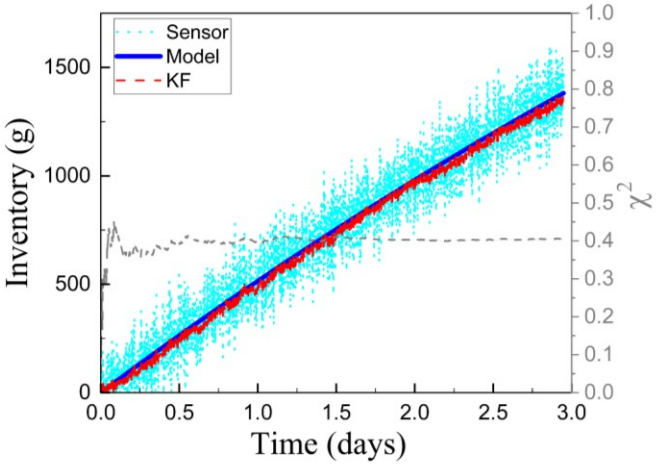


Figure 7. $I_{Sensor_1,t}$ (blue) or the Blanket component plotted against its simulated sensor measurements (dots) generated using equation 3. The Kalman Filter prediction calculations are show as a red dotted line. . χ^2 is plotted on the secondary axis in grey.

There are two important values for determining an optimized Kalman Filter prediction shown in the figure. The first value is the χ^2, which is shown as a dashed grey line on the image and the second value is σ_{KF}, which is shown in the title of the image and is the noise distribution of the Kalman Filter prediction. This, σ_{KF}, describes how far away the Kalman Filter prediction is from the mean or model values (blue). The goal is to minimize both the χ^2 and the σ_{KF} values to optimize the Kalman Filter predictions. This can be done by adjusting the $\boldsymbol{Q_t}$ values in the Kalman Filter calculation. Figure 8 is a plot of the χ^2 and the σ_{KF} quantities as a function of a changing $\boldsymbol{Q_t}$. The R_t value used for the data in the figure is the standard deviation of the sensor measurements. The red line in Figure 8 is the point at which the σ_{KF} reaches a minimum. Even though the χ^2 value continues to decrease and does not reach a minimum at the same point this is considered optimal because it is the

minimum point where the Kalman Filter predictions is nearest the model calculation. Figure 6 and Figure 7 are Kalman Filter predictions using the optimized $\boldsymbol{Q_t}$.

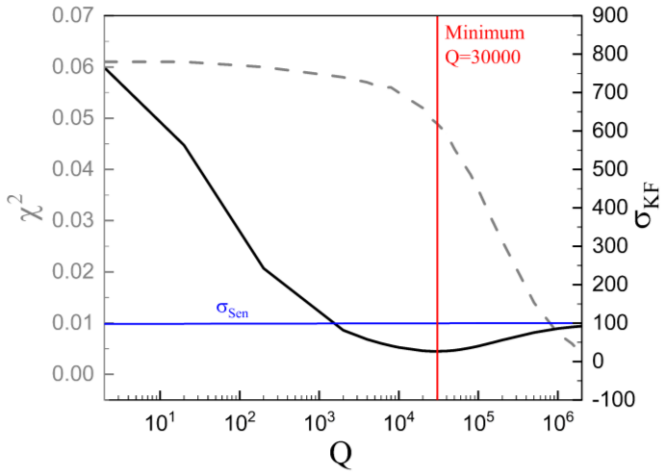


Figure 8. χ^2 and the σ_{KF} as a function of the $\boldsymbol{Q_t}$ value. The red line is the optimized Q value and the blue line is the noise distribution sigma on the sensor $\mathcal{N}(0, \sigma_{Sen}^2)$.

Early in the trace of Figure 8, σ_{KF} is far away from the model calculation values and the χ^2 is at its highest. This means that the Kalman Filter predictions differ from the model calculations. The red trace that represents the Kalman Filter predictions are far off from the true values as can be seen in Figure 9 graph (a.). At the point where σ_{KF} reaches a minimum ($\sigma_{KF} = 26.57$) is the point where the Kalman Filter predictions are nearest the model calculations. One can see that the Kalman Filter predictions are very near the model calculations in Figure 9 graph (b.).

As the trace in Figure 8 continues to increase in Q value the σ_{KF} begins to trend upwards toward the limit σ_{Sen} and χ^2 reaches a near minimum. Figure 9 (c.) is an image demonstrating what the Kalman Filter predictions look like at the end of the trace in Figure 8. The implication is that no matter how high you get in Q value the Kalman Filter predictions will not fit anything larger than the sensor measurement made. This makes sense if one looks at the Kalman Gain in Equation (6). If $\boldsymbol{Q_t}$ is considerably higher than R_t then K_t will trend to one, which means that the measurements will be weighted heavier in the Kalman Filter predictions and the measurements are very stable. Hence, this is the reason why the Kalman Filter predictions fit more closely with the sensor measurements when Q is high. Similarly, when Q is low like in Figure 9 graph (a.) the Kalman Filter

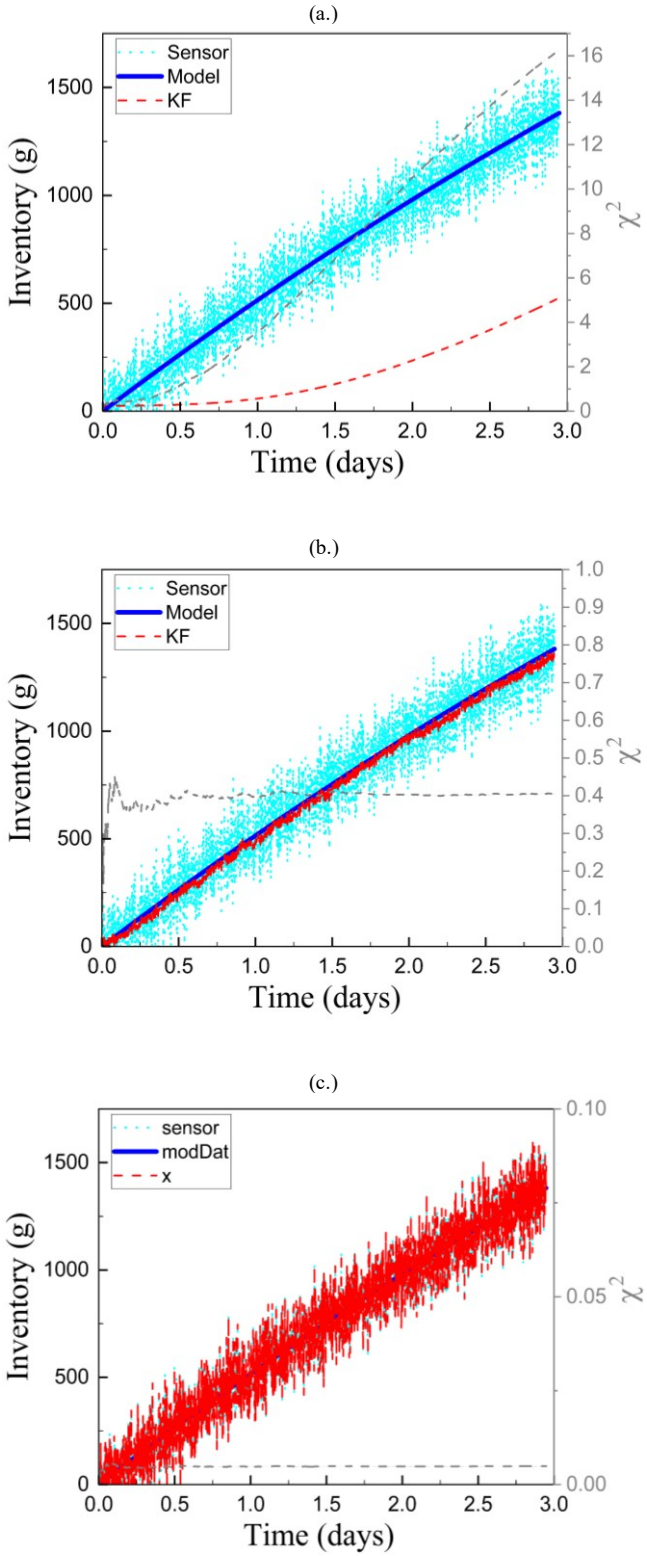


Figure 9. $I_{Sensor_1,t}$, model calculation, and Kalman Filter predictions for the early region (a.), mid region (b), and end region (c) of the trace from Figure 8.

'trusts' the predictions more and the Kalman Filter predictions are unstable.

5.3 Discretization

The amount of sensor data or the frequency at which the sensor data is taken per day plays a large role in the Kalman Filter prediction performance as well. The sensor data displayed in the previous sections was created using a sampling frequency of $f_s \approx 983.33 \frac{samples}{day}$. If the sampling frequency is increased by a factor of ten, $f_s \approx 9833.33 \frac{samples}{day}$, then a change in the Kalman Filter performance can be observed.

Figure 10 is the Kalman Filter predictions plotted against the increased sensor data and the model calculations. When comparing this to Figure 7 the Kalman Filter predictions look much 'tighter' and more comparable to the model calculations. Further comparisons can be made when comparing their χ^2 and the σ_{KF} values in Figure 11. The higher sampling frequency approaches a minimum at an earlier Q value and starts off with a prediction closer to the actual model calculations. The minimum σ_{KF} is also smaller for the higher frequency ($\sigma_{KF} = 9.53$), meaning that the noise spread from the actual value is lower. This leads to the 'tighter' looking Kalman Filter prediction.

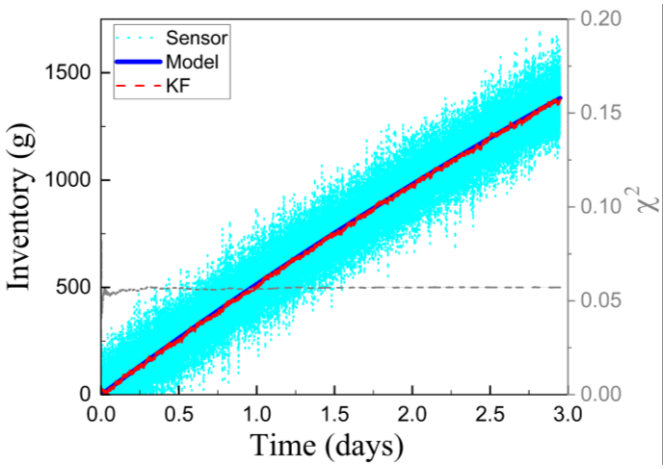


Figure 10. $I_{Sensor_1,t}$, model calculation, and Kalman Filter predictions for the optimized Q value at a sampling frequency of 9833.33 $\frac{samples}{day}$.

5.4 Percent Error Difference ($E_{\%}$)

The simulated sensor has an average percent error difference ($E_{\%,Avg}$) equal to 64.37%, meaning that the simulated sensor data is far away from the true value calculated by the model. Using sensor data with large, visible error is better to demonstrate the KF's capabilities since the KF is predicting what the true value is at each new time step using the noisy sensor measurement.

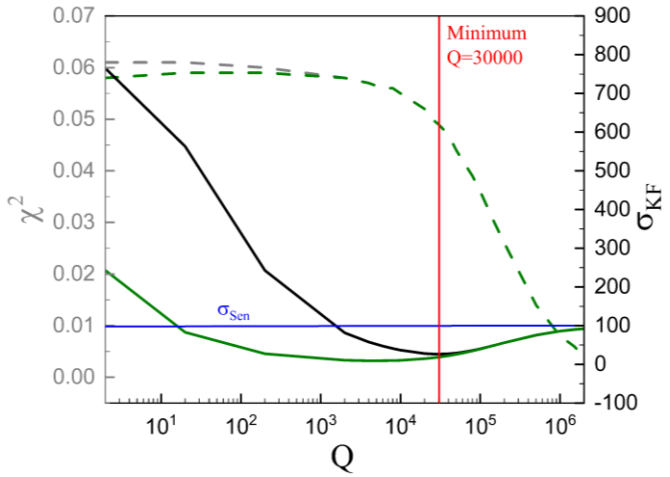


Figure 11. χ^2 and the σ_{KF} as a function of the $\boldsymbol{Q_t}$ value for a sampling rate of 983.33 $\frac{samples}{day}$ (black/grey) and 9833.33 $\frac{samples}{day}$ (green). The red line is the optimized Q value and the blue line is the noise distribution sigma on the sensor $\mathcal{N}(0, \sigma_{Sen}^2)$.

Figure 12 is the model calculated values for inventory plotted against the KF predicted values and the percent error difference, $E_{\%}$, between the two data sets (green). $E_{\%}$ gives a measurement of how near to the true values or the model values the KF predictions are over time. While the initial error is high during very early times, by the end of the day the error is reduced to $\approx 0.633\%$. By the end of the 3 days the KF predictions have an $E_{\%}$ that is near 0.001%. These results show that an accurate model, combined with the KF, can reduce noisy measurement with error around 64% to less than the accountancy standard of 1%.

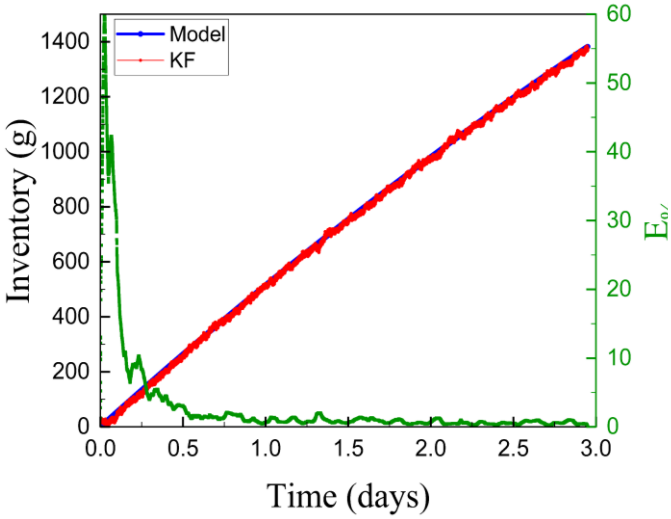


Figure 12. Model calculated values plotted against the KF predicted values, and the smoothed percent error difference, $E_{\%}$ ($green$). Early in time the E_5 is around 60%, but by the end of the first day it has dropped to around 0.663% and by the end of the 3rd day it is near 0.001%.

Since the KF predictions are essentially stabilizing, early in the process the predictions are unreliable, and the values are furthest away from the true model values. This is displayed by the large spike at the beginning of the trace. Early in the filtering process the filter is told that the predictions are not reliable and so the measurement or sensor data is weighted more heavily. As discussed, the sensor measurements have a $E_{\%,Avg} = 64.37\,\%$. This means that relying more heavily on the sensor measurements will not return reliable KF predictions. Therefore, as iterations of the KF continue the weighting between measurement and prediction begins to stabilize and the KF predictions become more accurate.

The decreasing trend of the $E_{\%}$ is a positive outcome, demonstrating quantitatively that the KF predictions are good estimates of the true values calculated by the model.

5.5 Bias

The Kalman Filter is essentially predicting the mean (model calculations) from the noisy sensor measurements and model information. Therefore, if there is any bias or random error the Kalman Filter, as implemented up to this point, will be inaccurate. Figure 13 is an image showing the Kalman Filter predictions for a measurement system that has an accumulating bias over the three-day period. The Kalman Filter accurately predicts the mean of the sensor measurement but has an increasingly large spread from the true value. This is not very useful since the values of interest are the model values or the true values of the system.

Almost every system in the real world will have some random error or some sort of bias that accompanies the measurement. This would essentially make the Kalman Filter useless; however, it is used very successfully in tracking, forecasting, self-driving cars, groundwater, and others. This is because there are ways to either correct the measurements prior to applying the Kalman Filter or improve upon the Kalman Filter to estimate the different random errors and biases [17, 18]. The bias may be treated separately through a two-step Kalman Filter and can either be fed back into the model where it is used to update the biased state estimation, or it is not fed back into the model [19]. When the bias is constant or known, no-feedback is a common approach. In the feedback approach the state is corrected at each time point for the bias.

The Bias-Aware Kalman Filter will not be outlined in this paper due to the need for a thorough investigation of the best approach for the fusion fuel cycle. However, the ongoing investigation and findings will be reported in a follow-on paper.

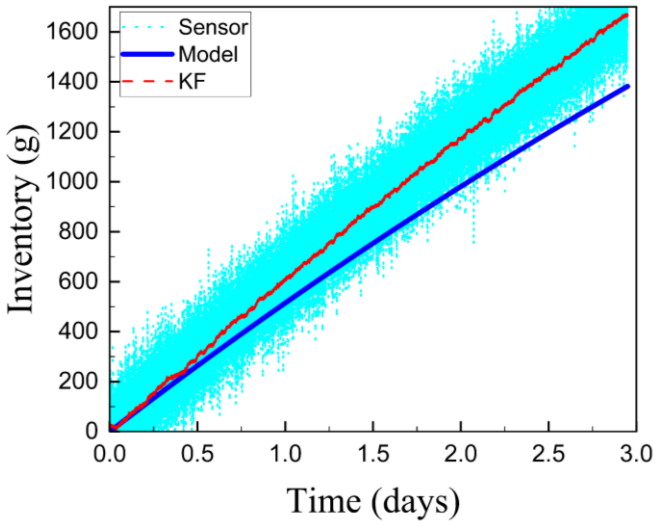


Figure 13. $I_{Sensor_{1},t}$, model calculation with a 0.01 bias accumulated over time and Kalman Filter predictions for the optimized Q value at a sampling frequency of 9833.33 $\frac{samples}{day}$.

6. Discussion

This paper has outlined that the Kalman Filter state prediction algorithm is potentially an option, with continued improvements, for accounting the true inventory of tritium in a dynamic fusion fuel cycle. Implementation of the KF with an accurate model has been shown to reduce noisy sensor data with around 64% error to around 0.001%, which is less than the 1% accountancy standard. Since sensors will be placed in an environment that does not grant ease of access and will need to be online for long periods of time, having an algorithm that can take the sensor data and more accurately estimate the tritium inventory will be invaluable. The continuously increasing tritium inventory will need to be accurately monitored since it is highly likely that a regulated maximum tritium inventory will be put into place for commercial fusion power plants. Accurate measurements will allow for the proper regulation of the tritium inventory in the system, such as moving quantities to onsite storage when thresholds are near to being met.

The sensor, due to the environment of the fusion fuel cycle, will have the potential to be exposed to high vibrations, ground loop issues, radiofrequency (RF) generating components, etc, in addition to the intrinsic accuracy of the sensing device. This will cause the sensor information to accumulate uncertainties (system errors and biases) as they continue to measure over time. The implementation of the Kalman Filter in this paper cannot handle

accumulation of biases; however, ongoing investigations are being performed to implement a Bias-Aware Kalman Filter that accounts for the inherent biases and system errors. These findings will be reported in a follow-on paper.

Furthermore, designs for fusion fuel cycles have become more complex over the years since the 1985 paper by Abdou, et. al. Increasing the complexity of the fusion fuel cycle model will be required to accurately report on ongoing fusion fuel cycle research. Understanding the contribution and losses from the plasma vessel will be required to utilize the tritium accountancy Kalman Filter for commercial fusion power plants. Future work is being performed in both areas of research.

ACKNOWLEDGEMENTS

The authors thank Jim Klein for helpful discussions and direction and are grateful for support from the Office of Fusion Energy Sciences, DOE. This work was produced by Battelle Savannah River Alliance, LLC under Contract No. 89303321CEM000080 with the U.S. Department of Energy. Publisher acknowledges the U.S. Government license to provide public access under the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan). The United States Government retains and the publisher, by accepting this article for publication, acknowledges that the United States Government retains a non-exclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this work, or allow others to do so, for United States Government purposes.

REFERENCES

1. Nishi, M., T. Yamanishi, and T. Hayashi, *Study on tritium accountancy in fusion DEMO plant at JAERI.* Fusion Engineering and Design, 2006. **81**(1-7): p. 745-751.
2. Abdou, M.A., et al., *Deuterium-Tritium Fuel Self-Sufficiency in Fusion Reactors.* Fusion Technology, 1985. **9**(2): p. 250-285.
3. ITER, *ITER research plan within the staged approach (Level III-Provisional Version)*. 2018, .
4. Klein, J.E., et al., *Tritium Accountancy in Fusion Systems.* Fusion Science and Technology, 2017. **67**(2): p. 420-423.
5. Kalman, R.E., *A New Approach to Linear Filtering and Prediction Problems.* Journal of Basic Engineering, 1960. **82**(1): p. 35-45.
6. Matthies, L., T. Kanade, and R. Szeliski, *Kalman filter-based algorithms for estimating depth from image sequences.* International Journal of Computer Vision, 1989. **3**(3): p. 209-238.
7. Moore, T. and D. Stouch, *A Generalized Extended Kalman Filter Implementation for the Robot Operating System.* 2016. **302**: p. 335-348.
8. Seong-Taek, P. and L. Jang Gyu, *Improved Kalman filter design for three-dimensional radar tracking.* IEEE Transactions on Aerospace and Electronic Systems, 2001. **37**(2): p. 727-739.
9. Kelly, A., *A 3D state space formulation of a navigation Kalman filter for autonomous vehicles*. 1994, CARNEGIE-MELLON UNIV PITTSBURGH PA ROBOTICS INST.
10. De Marina, H.G., et al., *UAV attitude estimation using unscented Kalman filter and TRIAD.* IEEE Transactions on Industrial Electronics, 2011. **59**(11): p. 4465-4474.
11. Pauwels, V.R.N., et al., *Simultaneous estimation of model state variables and observation and forecast biases using a two-stage hybrid Kalman filter.* Hydrology and Earth System Sciences, 2013. **17**(9): p. 3499-3521.
12. Meinhold, R.J. and N.D. Singpurwalla, *Understanding the Kalman Filter.* The American Statistician, 1983. **37**(2): p. 123-127.
13. Welch, G. and G. Bishop, *An introduction to the Kalman filter.* 1995: p. 127-132.
14. Abdou, M., et al., *Physics and technology considerations for the deuterium–tritium fuel cycle and conditions for tritium fuel self sufficiency.* Nuclear Fusion, 2020. **61**(1): p. 013001.
15. Lee, J.H. and N.L. Ricker, *Extended Kalman filter based nonlinear model predictive control.* Industrial & Engineering Chemistry Research, 1994. **33**(6): p. 1530-1541.
16. Wan, E.A. and R. Van Der Merwe. *The unscented Kalman filter for nonlinear estimation*. in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373)*. 2000. Ieee.
17. Kollat, J.B., P.M. Reed, and R.M. Maxwell, *Many-objective groundwater monitoring network design using bias-aware ensemble Kalman filtering, evolutionary optimization, and visual analytics.* Water Resources Research, 2011. **47**(2).
18. Ren, H. and Z. Lu, *Measurement Bias Estimation in the Problem of Target Tracking.* Wireless Communications and Mobile Computing, 2019. **2019**: p. 1-10.

19. Drécourt, J.-P., H. Madsen, and D. Rosbjerg, *Bias aware Kalman filters: Comparison and improvements.* Advances in Water Resources, 2006. **29**(5): p. 707-718.