The Kalman Filter

By Anoushka Alavilli, ECE '22

Introduction

The Kalman Filter (KF), including its purpose, functionality, and applications is discussed in this paper. The Kalman Filter is a tool used by people for various applications to process noisy sensor data measurements. It is often used when the model of the system is unknown as the KF allows information to be uncovered about the system without any prior knowledge of its behavior. The KF is recursively determined in an algorithm where its variables are continuously updated and corrected. In general, the KF works to combine many (noisy) measurements of a system to come up with a better estimate than any of the individual inputted sensor measurements and thus predict the future states of a system. First, the construction is described, followed by the parameters and algorithm of the KF, and lastly, some applications and variations are outlined.

Technical Details

The inputs and formulation of the Kalman Filter are described in this section along with the algorithm through which the filter is evolved to refine the state estimates.

Filter Design

The KF is useful when given a single or multiple sensor readings that are not individually proper estimates of the desired value due to random or systematic error. A state space model of the system is designed to estimate the true measured value in the presence of noise. The KF can be described in two equations, shown in (1) and (2). In these dependent equations, F relates the previous measurement, B relates the optional control input (u_k) with the current state, $H_k x_k$ is the predicted measurement, z_k is the actual (observed) measurement, and K is the Kalman gain (or "blending factor," in reference to the blending of the observed and predicted states) [3].

$$x_{k+1} = Fx_k + Bu_k \quad (1)$$

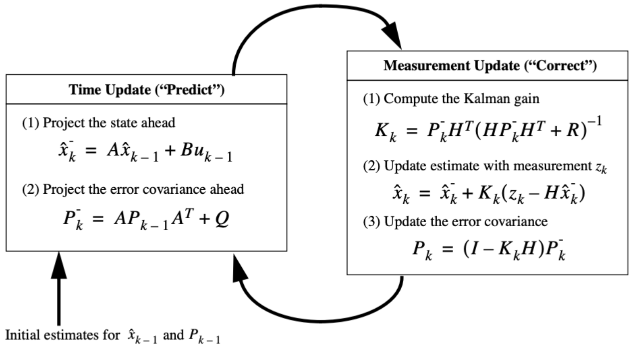
$$x_{k,corrected} = x_k + K_k(z_k - H_k x_k) \quad (2)$$

The matrix K is algorithmically determined to minimize the mean-squared error—i.e. the difference factor in (2)—and is ensured to have negative eigenvalues so that the system is stable (the solution of the system will converge to a finite value) [1, 3]. All matrices in (1) and (2) are determined based on the presence of noise as well.

Algorithm

The use of a Kalman Filter, and the process of evolving the system to make more accurate state estimations, is called *Kalman Filtering* or *Linear Quadratic Estimation* (LQE). LQE is in fact the optimal full state estimator given knowledge of noise in measurements [1]. In essence, LQE involved a cycle of a) measuring the difference between the filter's state prediction and the actual measurement and b) making the prediction based on this correction [3]. This recursive, corrective process edits variables in the algorithm, including K (the Kalman gain), $x_{k,\ corrected}$, and P (another matrix that represents the expected value of the difference between the prediction and a future measurement). The LQE cycle is further detailed in Fig. 1 and the remaining parameters are explained below.

Figure 1

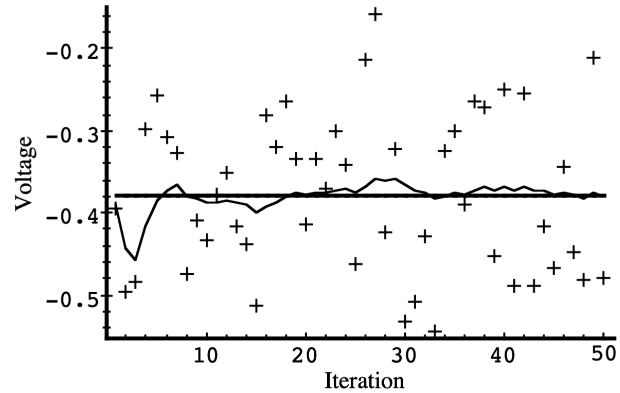


A mathematical description of the LQE cycle [3]

In LQE, there are two important parameters that should be determined (or estimated) *a priori*. These matrices are called Q and R. R is a measurement of the noise variance, which can be determined before implementing the filter. The presence of Gaussian (white) noise is common, however, any distribution of noise can be inputted to the filter. Q is more difficult to calculate: it is the *process* noise covariance (i.e. any inaccuracy that results from the LQE process itself) and it is harder to calculate because of the difficulties of "self-observation"—we implement the LQE to ideally contain zero noise. Because of this, Q is generally set to take into account a significant amount of uncertainty to allow for generous error correction [3]. It is important to note that Q and R remain the same throughout the algorithm. When referring to the aforementioned LQE cycle, R is useful in the first part of the cycle, Q is useful in the second part.

Through the recursive nature of LQE, there is a natural evolution in the filter's prediction of the measured value. We see its prediction eventually converge to the correct value after several iterations, as can be seen in Fig. 2, where the "true" value is marked as the horizontal solid line, the measurements are marked with '+' symbols, and the output of the LQE at various iterations is given by the curve. Note that the estimate of the LQE is not guaranteed to strictly improve with each iteration.

Figure 2.



Evolution of LQE estimate compared to true value [3]

Applications

There are many applications of the Kalman Filter, in areas including signal processing, voltage and frequency control in computers, and the control of autonomous systems such as vehicles [2]. One example is motion control of a robot. An area of study in robot locomotion is self-localization. While there are algorithms, such as SLAM (Simultaneous Localization and Mapping), that can be used for this purpose, they take in data from the robot's, for example, vision sensors or odometry data to determine the location estimate. Issues such as wheel slippage and incorrect motor encoders can give faulty measurements. One can take advantage of a secondary sensor (for example, a camera), however, and use a Kalman filter to combine the multiple measurements to error-check the measurements coming from the odometer and hopefully to provide a better location estimate. As discussed, the Kalman Filter can be applied on one source of data as well (it need not combine measurements from different sensors). Indeed, in [4], radar data is used to track ship maneuvers at "cruising speed."

It is important to note that variations of the Kalman Filter exist, including ones that are well-suited to error-minimization of non-linear systems. Such a Kalman Filter is known as an Extended Kalman Filter (EKF) and linearizes the process with respect to the current mean and covariance and is useful in applications such as navigation systems. More information about the Extended Kalman Filter can be found in [2].

Additionally, there exists the Unscented Kalman Filter (UKF). The Unscented Kalman Filter is most useful when the process is highly non-linear. Though it can be applied to any application needing the EKF, its higher computational power requirement makes it the less desired choice when the EKF suffices [2].

Conclusion

The motivation, technical details, algorithm, and applications for the Kalman Filter are discussed in this paper. Variations of the Kalman Filter such as the Extended Kalman Filter and the Unscented Kalman Filter were also discussed. For accessible sources on the Kalman Filter, see [5-7].

Works Cited

1. "The Kalman Filter [Control Bootcamp]." Youtube, uploaded by Steve Brunton, 6 Feb. 2017, www.youtube.com/watch?v=s_9InuQAx-g. Accessed 18 Jan. 2022.

2. Pei, Yan. "An Elementary Introduction to Kalman Filtering." *Communications of the ACM*, vol. 62, Nov. 2019, pp. 122-33, https://doi.org/10.1145/3363294. Accessed 10 Nov. 2021.

3. Welch, Greg, and Gary Bishop. "An Introduction to the Kalman Filter." 17 Sept. 1997. pdf.

4. Nadzilah, Amicytia, et al. "Application of Kalman filter to track ship Maneuver." *2017 5th International Conference on Cyber and IT Service Management (CITSM)*. *IEEE Xplore*, https://doi.org/10.1109/CITSM.2017.8089287. Accessed 18 Jan. 2022.

5. Babb, Tim. "How a Kalman filter works, in pictures." *Bzarg*, www.bzarg.com/p/how-a-kalman-filter-works-in-pictures/. Accessed 18 Jan. 2022.

6. Hugegene. "Kalman Filter Explained with Easily Understandable Matrices." *Medium*, 18 Oct. 2021, www.cantorsparadise.com/kalman-filter-explained-with-kinematics-and-easily-understandable-matrixes-5bd956c41b07. Accessed 18 Jan. 2022.

7. *Youtube*. www.youtube.com/playlist?list=PLX2gX-ftPVXU3oUFNATxGXY90AULiqnWT. Accessed 18 Jan. 2021