A Survey of Parallelization of Kalman Filters

Brian J Gravelle

June 9, 2017

Abstract

Kalman Filters have been an important aspect of many computer systems since they were first developed in the 1960s. Many of these applications require real-time speeds which has necessitated the parallelization of the Kalman filter from early in its development. This survey explores efforts to accelerate the algorithm with parallel techniques over the last 30 years and discusses of the domains to which those parallelizations were applied. Many of the techniques are highly specialized to a particular domain so special care is given to how the parallelizations could be applied to different areas or are restricted to a specific application. Additionally, some discussion of future research is provided.

1 Introduction

Since their inception in 1960 Kalman filters have been heavily used for the estimation of states in dynamic systems. These filters have numerous applications especially in tracking various objects; they have been used to track object in conjunction with computer vision, radar tracking for airplanes and ships, vehicle and missile guidance systems, orbit calculation for satillites, and particles in HEP. Additionally, they have been used for demographic and economic studies, industrial process control, climate science, and geological surveys. Often these applications, the tracking ones especially, require real-time performance, pushing the performance demands of Kalman filtering.

At first the filtering was done primarily using customized circuits, each designed for the specific application to match the number of states in use. As modern computers developed, new opportunities for flexibility and improved performance were discovered. In many cases parallelism at different levels features in the performance enhancement of the Kalman filter. This survey reviews some of the implementations and applications of parallel Kalman filters.

Table 1: List of matrices involved in the Kalman filter.

Name	Description	dimensions
A	system dynamics	nxn
C or H	measurement matrix	mxn
Q	process (system) noise	nxn
R	measurement noise	mxm
P	error covariance	nxn
K	Kalman gain	nxm

2   Background

Kalman filters [8] were originally presented in 1960 to provide a solution to filtering and prediction problems in dynamic systems. In particular, the results targeted signals involved in communication and control systems. Since then, the Kalman filter has been applied to many dynamic systems in fields including physics, signal processing, economics, and others. In this section, we present some background information to enable the reader to better understand the rest of the survey. Interested readers are directed to the following sources: [1, 20, 3, 8].

The Kalman filter is used to estimate the state of some dynamic system based on measurements, models, and associated errors of that system. This information is combined into a set of matrices that represent the whole system (Table 1). The sizes of the matrices are based on the number of states in the system model (n) and the number of states measured in each time step (m). While some applications have larger state sizes, most require only a small number of states, resulting in matrices that are at the largest 20x20. In addition to the matrices there are also two vectors; x of length n that holds the estimated state and y of length m for the current measurements.

At each time step the predicted state and the most recent measurement are compared and these values, combined with the error matrices are used to predict the next state. This prediction is based on weighting the measurement and model (expressed in the Kalman gain) and emphasizing one over the other based on error values. The accuracy of the Kalman filter relies heavily of the accuracy of the error and covariance matrices since these are integral in determining the Kalman gain with each step.

The Kalman equations are shown in Equations 1 - 5. Equations 1 and 2 are the predict portion, estimating the next state based on past data. Equations 3 - 5 and the correct section, updating the estimated state and

Kalman gain based on the measurement and predicted state. There are additional versions such as the Extended Kalman Filter that implement improvements such as time variant error functions, or other changes to improve performance or accuracy.

$$\hat{x}_{new} = A\hat{x} \tag{1}$$

$$P = APA^T + Q \tag{2}$$

$$K = PC^T(CPC^T + R)^{-1} \tag{3}$$

$$\hat{x} = \hat{x}_{new} + K(y - C\hat{x}_{new}) \tag{4}$$

$$P = (I - KC)P \tag{5}$$

3 Early Work

Since the kalman filter has been around for so long, they have been the subject of many research projects seeking to improve the performance and/or accuracy. Two of the early results are discussed in this section. Both works are intended for Multiple Target Tracking of planes, satellites, or other rapidly moving targets.

3.1 Circuit Parallelization

In [17], Shaffer presents a circuit design for an extended Kalman filter. While this work is a hardware implementation rather than software, it still applies the theoretical concepts and parallel patterns that are common in modern parallel software. In particular, Shaffer uses a pipeline pattern to take advantage of the concurrency of the algorithm and streaming nature of the data. Figure 3.1 shows the data flow diagram of the system.

The pipelining of the circuit which was enabled by the removal of recursion from the implementation, resulted in a 360x increase in throughput for the system. This improvement allowed the authors to over 500 targets while collecting 1188 samples/sec.
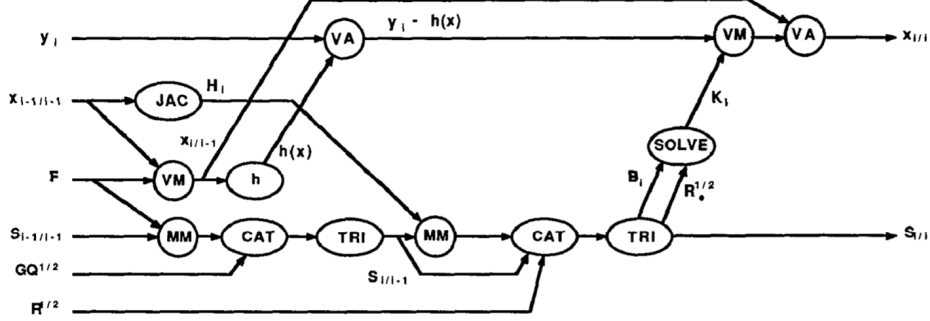
Figure 1: Dataflow diagram from [17]

3.2 Early Vector Parallelism

Palis and Krecker, [14], use the SIMD arrays available in the Connection Machine (a supercomputer manufactured in the late 1980s) to parallelize the work of a single filter and the many-cores of the machine to parallelize the operation of numerous independent filters. The authors use the Square Root Kalman filter which is a variation that allows it to be processed using single-precision calculations without information loss.

The first step to parallelism is to use the SIMD operations to improve the speed of the linear algebra operations. These operations take advantage of the independence of row and elements in matrix operations and can improve the speed of a single Kalman filter operating on a single node. The authors show that the time for computation grows linearly with the number of states in the parallel version rather than cubicly as it does in the serial version.

Second, the authors demonstrate that when tracking multiple targets the Kalman filters (one or more for each target) can be parallelized by running them on independent processors in the Connection Machine cluster. Their results show that this method can track many more targets than a serial machine.

4 Modern Implementations and Applications

4.1 High Energy Physics

Cerati et al. have published several similar papers [6, 4, 5] on parallelizing Kalman filters for use at CERN. The physicists utilize Kalman filters to build

tracks of subatomic particles during the runs of the LHC. Given the size and speed at which these events occur, the filtering must occur as rapidly as possible. Similar to Shaffer's work in the 90s, these authors explloit two levels of parallelism on XeonPhi; the vector instructions for the matrix operations and the concurrent execution

4.2 Climate Science

[13, 12, 11]

4.3 Smart Grid and GPUs

[9]

4.4 Cell Tower Power Management

[16]

5 Sensor Applications

[15, 19, 18, 7]

6 FPGA Implementations

[2, 10]

7 Conclusion

Sorry I didn't finish in time. Stay tuned for updates!

References

[1] Samuel S Blackman. Multiple-target tracking with radar applications. *Dedham, MA, Artech House, Inc., 1986, 463 p.*, 1986.

[2] Vanderlei Bonato, Eduardo Marques, and George A Constantinides. A floating-point extended kalman filter implementation for autonomous mobile robots. *Journal of Signal Processing Systems*, 56(1):41–50, 2009.

[3] Amarjit Budhiraja, Lingji Chen, and Chihoon Lee. A survey of numerical methods for nonlinear filtering problems. *Physica D: Nonlinear Phenomena*, 230(1):27–36, 2007.

[4] Giuseppe Cerati, Peter Elmer, Slava Krutelyov, Steven Lantz, Matthieu Lefebvre, Kevin McDermott, Daniel Riley, Matevž Tadel, Peter Wittich, Frank Würthwein, et al. Kalman filter tracking on parallel architectures. In *EPJ Web of Conferences*, volume 127, page 00010. EDP Sciences, 2016.

[5] Giuseppe Cerati, Peter Elmer, Steven Lantz, Ian MacNeill, Kevin McDermott, Dan Riley, Matevž Tadel, Peter Wittich, Frank Würthwein, and Avi Yagil. Traditional tracking with kalman filter on parallel architectures. In *Journal of Physics: Conference Series*, volume 608, page 012057. IOP Publishing, 2015.

[6] Giuseppe Cerati, Peter Elmer, Steven Lantz, Kevin McDermott, Dan Riley, Matevž Tadel, Peter Wittich, Frank Würthwein, and Avi Yagil. Kalman filter tracking on parallel architectures. In *Journal of Physics: Conference Series*, volume 664, page 072008. IOP Publishing, 2015.

[7] Hamid R Hashemipour, Sumit Roy, and Alan J Laub. Decentralized structures for parallel kalman filtering. *IEEE Transactions on Automatic Control*, 33(1):88–94, 1988.

[8] Rudolph Emil Kalman et al. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[9] Hadis Karimipour and Venkata Dinavahi. Extended kalman filter-based parallel dynamic state estimation. *IEEE transactions on smart grid*, 6(3):1539–1549, 2015.

[10] Yang Liu, Christos-Savvas Bouganis, and Peter YK Cheung. Efficient mapping of a kalman filter into an fpga using taylor expansion. In *Field Programmable Logic and Applications, 2007. FPL 2007. International Conference on*, pages 345–350. IEEE, 2007.

[11] PM Lyster, SE Cohn, R Menard, LP Chang, SJ Lin, and RG Olsen. Parallel implementation of a kalman filter for constituent data assimilation. *Monthly weather review*, 125(7):1674–1686, 1997.

[12] Richard Ménard and Lang-Ping Chang. Assimilation of stratospheric chemical tracer observations using a kalman filter. part ii: χ 2-validated

results and analysis of variance and correlation dynamics. *Monthly weather review*, 128(8):2672–2686, 2000.

[13] Richard Ménard, Stephen E Cohn, Lang-Ping Chang, and Peter M Lyster. Assimilation of stratospheric chemical tracer observations using a kalman filter. part i: Formulation. *Monthly weather review*, 128(8):2654–2671, 2000.

[14] Michael A Palis and Donald K Krecker. Parallel kalman filtering on the connection machine. In *Frontiers of Massively Parallel Computation, 1990. Proceedings., 3rd Symposium on the*, pages 55–58. IEEE, 1990.

[15] BS Rao and Hugh F Durrant-Whyte. Fully decentralised algorithm for multisensor kalman filtering. In *IEE Proceedings D (Control Theory and Applications)*, volume 138, pages 413–420. IET, 1991.

[16] Olov Rosén, Alexander Medvedev, and Torbjörn Wigren. Parallelization of the kalman filter on multicore computational platforms. *Control Engineering Practice*, 21(9):1188–1194, 2013.

[17] P. L. Shaffer. Implementation of a parallel extended kalman filter using a bit-serial silicon compiler. In *Proceedings of the 1987 Fall Joint Computer Conference on Exploring Technology: Today and Tomorrow*, ACM '87, pages 327–334, Los Alamitos, CA, USA, 1987. IEEE Computer Society Press.

[18] Demetri P Spanos, Reza Olfati-Saber, and Richard M Murray. Approximate distributed kalman filtering in sensor networks with quantifiable performance. In *Information Processing in Sensor Networks, 2005. IPSN 2005. Fourth International Symposium on*, pages 133–139. Ieee, 2005.

[19] Demetri P Spanos, Reza Olfati-Saber, and Richard M Murray. Distributed sensor fusion using dynamic consensus. In *IFAC World Congress*. Citeseer, 2005.

[20] Greg Welch and Gary Bishop. An introduction to the kalman filter. 1995.