The Kalman Filter Online Learning Tool

Greg Welch, client

March 8, 2001

Chris Riley, Director                       Tom Bodenheimer, Producer

Erin Parker, Admin. Leader                 John Carpenter, Librarian
1 Document Change History

- March 8, 2001
  - Annotated Outline

Contents

1 Document Change History 1

2 Introduction 2
  2.1 Statement of purpose 2
  2.2 Document conventions 2
  2.3 Other relevant documents 2

3 System Overview 3
  3.1 High-level view of the system 3
  3.2 Directory structure 4
  3.3 Files 4

4 Procedures 5
  4.1 Editing the source 5
  4.2 Building the system 5
  4.3 Testing the system 5
  4.4 Updating the documentation 5

A The User Manual 6
  A.1 Quick Start 6
  A.2 Descriptions of the parameter settings and control buttons 6
  A.3 Stepping mode 6

B Development System Requirements and Information 7

C Contact Information 8
2 Introduction

2.1 Statement of purpose

The purpose of this document is to provide instructions and guidance for the client (or technical staff appointed by the client) to use when extending/modifying The Kalman Filter Online Learning Tool.

The Kalman filter is a set of mathematical equations that provides an efficient computational estimate of the state of a process (e.g., the position and orientation of an airplane) given a time-varying sequence of noisy measurements (e.g., air speed, pressure, temperature, engine thrust). The filter is a popular mathematical estimator due to its efficiency and robustness. The Kalman Filter Online Learning Tool is a web-based tool to help develop the intuition and insight of novice users regarding the behavior of the Kalman filter. Users have the ability to change various input parameters and then see how the Kalman filter responds for a given set of noisy measurements.

It is assumed that the reader of this document has a general understanding of The Kalman Filter Online Learning Tool. In particular, the reader should be familiar with how the tool works, having used the tool and read the User Manual (see appendix A). It is also advised that the reader have experience with the JavaTM programming language.

2.2 Document conventions

Source code (function names and parameters) will appear in the typewriter font. Other conventions will be introduced here as they are needed.

2.3 Other relevant documents

If the team decides that including some of the other documents (Contract, Design Specification, etc.) is appropriate, they will be mentioned here and attached as an appendix.
3 System Overview

3.1 High-level view of the system

The Kalman Filter Online Learning Tool is divided into four modules (see figure 1), each performing a distinct function. (The high-level diagram may not look exactly like figure 1, but it will be included here.)

![High-level view of the system](image)

Figure 1: High-level view of the The Kalman Filter Online Learning Tool system.

**User interface.** Primarily, the user interface serves to provide the user with choices that govern the execution of the Kalman filter for a specific test problem. The user selects among several options for the actual dynamics, the estimate, the measurement model and the process noise. The user also selects when to run!in UI and step!in UI through the simulation.

Secondarily, the user interface acts as the main controller for the system. It passes parameters to the Kalman filter engine and collects the truth, estimate, covariance and residual values for each time-step of the simulation. The user interface passes such simulation values to the plotter and is responsible for displaying the plots produced. The user interface also passes the values for an entire simulation to the data repository and calls on it for a single time-step’s values when in step mode.

**Kalman filter engine.** The Kalman filter engine implements the Kalman filter equations for all models of actual dynamics, estimate and measurement.

**Plotter.** The plotting tool generates plots of the simulation values passed from the user interface.

**Data Repository.** The data repository stores a record of the Kalman filter iterations and values computed.
3.2 Directory structure

Material for this section will evolve as implementation continues. It will particularly be affected by the installation of revision control software. This section will discuss the organization of project files into directories according to modules, etc.

3.3 Files

This section also has yet to be resolved. It will include a list of all project files, a brief description of their contents and their location.
4 Procedures

4.1 Editing the source

This section will involve an explanation of the how the source code is organized in CVS, as well as, the location of the code. Instructions on using CVS to do the following will be provided.

- checking out the source code
- editing the source code
- checking in the source code
- comment procedures for each revision

4.2 Building the system

This section will layout guidelines for building the executable. In particular, the following will be discussed.

- which departmental machines should be used for this purpose
- JDK: which and how to use

4.3 Testing the system

This section will make suggestions on how to best test the application. The client’s Matlab code and results are a good way to test the accuracy of the data and plots.

4.4 Updating the documentation

The Implementation Manual and Maintenance Manual should be updated when necessary. This section will provide details on where to get the documents and what format they are written in (\LaTeX or Microsoft Word™).
A The User Manual

The User Manual is provided to give the reader some context for how the The Kalman Filter Online Learning Tool is to be used, as well as, to document a set of instructions (should they ever be needed).

A.1 Quick Start

2. Click on Use the The Kalman Filter Online Learning Tool.
3. Leave the controls in their default settings.
4. Click on the run button.
5. View the plots.
6. Close the plot window.

A.2 Descriptions of the parameter settings and control buttons

This section will give short description of each (including options, defaults, actions, etc.)

- truth
- estimate
- measurement
- process noise
- run
- step

A.3 Stepping mode

The stepping mode (possibly being the only complicated feature of the tool) will be carefully explained.
B Development System Requirements and Information

The following are requirements for extensions/modifications to the project.

- Java™ Development Kit: JDK 1.3.0 for Linux™ from Sun™
  (includes Java™ 2 RTE and Java Hotspot™ Client VM)

- Web Browser: TBA
  (versions of Netscape™ or Internet Explorer™, plug-ins)

The following is information about tools used in the initial project development, provided in case they can be useful for further development of the project.

- Integrated Development Environment: Forte™ for Java™ 2.0 from Sun™
  (available at http://www.sun.com/forte/ffj)

- Plotting Tool: Pttplot 3.1
  (available at http://ptolemy.eecs.berkeley.edu/java/ptplot3.1/ptolemy/plot/doc)

- Matrix Package: TBA
  (URL, version)

- Version Control Software: CVS
  (version)
The following students completed *The Kalman Filter Online Learning Tool* as a course project for COMP145 at the University of North Carolina at Chapel Hill.

**DIRECTOR** Christopher Riley, cjriley@cs.unc.edu

**PRODUCER** Thomas Bodenheimer, bodenhei@cs.unc.edu

**ADMINISTRATIVE LEADER** Erin Parker, parker@cs.unc.edu

**LIBRARIAN** John Carpenter, carpente@cs.unc.edu

The client for the project was Greg Welch, a professor in the Department of Computer Science at UNC-CH, welch@cs.unc.edu.
Index

actual dynamics, see truth
   in UI, 3
control button, 6
   in UI, 3

truth, see actual dynamics
   setting, 6

user interface, 3

web browser
   requirements, 7

CVS
   editing the source, 5
   initial project development, 7

IDE
   initial project development, 7

Implementation Manual
   updating, 5

JDK
   building the system, 5
   requirements, 7

Kalman filter
   what is the, 2
   Kalman filter engine, 3
   Kalman Filter Online Learning Tool, what is the,
   2

Maintenance Manual
   updating, 5

matrix package
   initial project development, 7

measurement model
   in UI, 3
   setting, 6

plotter, 3

plotting tool
   initial project development, 7

process noise
   in UI, 3
   setting, 6

run, 3
   control button, 6

step