

M-KFTOOL --- KALMAN FILTERING SOFTWARE TOOLBOX for MATLAB®

Version 2



M-KFTOOL is a library of MATLAB m-file modules and main programs used for the implementation of discrete Kalman filtering applications. M-KFTOOL enables you to simulate a specific discrete Kalman filtering application quickly and easily without the need to code and test the basic Kalman filtering algorithms.

Due to the fact that this software tool is primarily addressed to a practicing engineer, effort was made to implement the most efficient algorithms available in the technical literature. Both discrete conventional and U-D form Kalman filter formulations are included. When appropriate, more than one module performs the same mathematical computation by using different methods, so you can select the approach that best meets your specific requirements.

The MATLAB® Kalman Filtering Software Toolbox contains numerous main programs that can be modified to fit specific application needs. Once the model of the application is derived or selected, the practitioner can use the software library to implement and test the validity of the proposed GPS application.

All m-files are compatible with MATLAB version 5.0 and higher, and most of them are also compatible with previous versions and Student Edition of MATLAB.

A complete (more than 150-pages) user's guide and reference manual contains detailed documentation for each module/program included in the library. To facilitate the search for a specific function, module/program or input/output file, the manual contains a complete reference table by function as well as lists of all modules, programs, input/output files in alphabetical order.

There are more than numerous fully explained examples (in excess of 40 pages) with input and output data, and generated plots; also a special section with discrete Kalman filtering applications is included.

The major building functions of the M-KFTOOL are divided in the following categories:

- Matrix storage and allocation
- Specialized matrix operation
- Specialized statistics functions and utilities
- Specialized plotting programs.
- General purpose and conventional Kalman filtering functions

- Specialized U-D Kalman filtering functions
- Application dependent modules
- GPS application modules

For a complete list of the modules and main programs included consult the file contents.m. Overall, there are about 100 m-files (modules and main programs), and more than 50 input and output data files.

The MATLAB source code is royalty free, i.e. the user can incorporate this software in his/her particular application but is not permitted to resell the software as is or with changes.

Here is a summary of the most significant features of GPS Toolbox software:

- all m-file modules and programs are written in MATLAB language/environment
- all m-file modules are free of input/output statements (except, of course, for those utilities that require printed output/graph, if any)
- all m-file modules are provided with detailed documentation, including scope, usage, description of parameters, remarks/notes, references, external MATLAB macros/modules used, and date of last update
- all main/test programs are provided with input/output reference data, and several detailed examples are given
- flexibility, user-friendly, and open-ended strategy. An open-ended strategy was followed which means that the user can either complement the GPS Toolbox services with his own functions or, alternatively, use GPS Toolbox functions as add-ons in conjunction with other libraries.

Computer/Software Requirements:

- IBM hardware compatible PC/386/486 or Pentium with Windows 3.1x or higher, and 640 Kbytes of RAM memory; math coprocessor for faster operation.
- MATLAB version 5.0 or higher; most m-file modules/programs are compatible with previous versions as well as the Student Edition of MATLAB

Technical Support:

- free technical support by e-mail or fax to any registered user
- future versions and updates to keep users current with the latest M-KFTOOL developments to any registered user: free upgrades for one year since purchase

List Price: US\$499. Educational discount of 20% is available

Availability: free demo

Copyright (C) 1992–2008 by L3NAV Systems. All Rights Reserved Worldwide.

=====
% **M-KFTOOL -- Kalman Filter Software Toolbox for Matlab(R)**
% **LIST OF FUNCTIONS**
=====

=====
% **Matrix Storage and Allocation**

% matc2r rectangular matrix storage transformation from one-dimensional column-wise
% to one-dimensional row-wise
% matr2c rectangular matrix storage transformation from one-dimensional row-wise to
% one-dimensional column-wise
% matuc2r upper triangular matrix storage transformation from one-dimensional
% column-wise to one-dimensional row-wise
% matur2c upper triangular matrix storage transformation from one-dimensional
% row-wise to one-dimensional column-wise
% mr1to2 rectangular matrix storage transformation from one-dimensional column-wise
% array to two-dimensional array
% mr2to1 rectangular matrix storage transformation from two-dimensional array to
% one-dimensional column-wise array
% msc2f symmetric matrix storage transformation from one-dimensional array column-
% wise - only the upper triangular part stored, to two-dimensional array
% msf2c symmetric matrix storage transformation from two-dimensional array to one-
% dimensional array - column-wise, only the upper triangular part stored
% msre reconstruct a full symmetric matrix from its stored upper triangular part;
% both input and output matrices are stored column-wise into one-dimensional
% arrays
% mstr extract the upper triangular part from a symmetric matrix; both input and
% output matrices are stored column-wise into one-dimensional arrays
% mudc2f restore full U and D matrices stored as two-dimensional arrays from its
% compact upper triangular part stored column-wise as one-dimensional array
% mudf2c store the full U and D matrices stored as two-dimensional arrays to its
% compact upper triangular part stored column-wise as one-dimensional array
=====

=====
% **Specialized Matrix Operations**

% maat post-multiplication of a rectangular matrix by its transposed matrix; the
% input matrix is stored column-wise, one-dimensional, and the resultant
% symmetric matrix is stored column-wise - only the upper triangular part
% mmab multiplication of two rectangular matrices when the resultant matrix is
% known to be a symmetric matrix; the input matrices are stored one-
% dimensional, column-wise, and the resultant matrix is stored column-wise -
% only the upper triangular part
% mmrt multiplication of a rectangular matrix and an upper triangular matrix; the
% rectangular matrix is stored into two-dimensional array, the upper
% triangular matrix is stored into one-dimensional array column-wise - only
% the upper triangular part, and the resultant matrix is stored into two-
% dimensional array
% mphiu multiplication of a square matrix stored into two-dimensional array and a
% unit upper triangular matrix stored into one-dimensional array
% column-wise -- only the upper triangular part; the resultant matrix is
% stored into two-dimensional array
=====

=====
% **Specialized Statistics Functions and Utilities**

% cep circular error probable (CEP) computation
% convcon setting of most used conversion constants
% gauss_1 probability density function of the normal Gaussian distribution
% genrn generation of random numbers with normal (Gaussian) distribution
% gmp1 generation of first order Gauss-Markov sequence
% gmp2 generation of second order Gauss-Markov sequence
% rms root mean square (RMS) of a sample
% rms2 modified root mean square (modified RMS) of a sample

```

% rss      root sum square (RSS) of a three component vector sample
% rssxy    root sum square (RSS) of a two component vector sample
% rwalk    generation of a random walk process
% statup   computation of the running mean, standard deviation and root mean
%          square for a sample
% vep      vertical error probable (VEP) computation
%-----
% xcepvep  main program used to compute CEP or VEP
% xgenrn   main program generating random numbers with normal (Gaussian) distribution
% xgmp1    main program generating first order Gauss-Markov sequence
% xgmp2    main program generating second order Gauss-Markov sequence
% xrwalk   main program generating random walk process sequence
% xstat    main program testing the following modules: rms, rss, rssxy, and statup
% xstatc   main program determining mean, standard deviation, and root mean square
%          (rms) of the elements of a specified column of the input array
%=====

```

Specialized Plotting Programs

```

% xpbar    main program plotting a bar graph for selected columns of the input array
% xyp1     main program plotting the x-y graph (with manual scaling) for selected
%          columns of the input array
% xyp1s    main program plotting the x-y graph (with manual scaling) for selected
%          columns of the input array; statistics (mean, standard deviation, and rms)
%          is incorporated
% xyp2w    main program plotting two x-y graphs (with manual scaling) for selected
%          columns of the input array, in two windows/subplots; statistics (mean,
%          standard deviation, and rms) is incorporated
% xyp3w    main program plotting three x-y graphs (with manual scaling) for selected
%          columns of the input array, in three windows/subplots; statistics (mean,
%          standard deviation, and rms) is incorporated
% xypc2    main program plotting a x-y graph of the difference between two data
%          columns from different data files; statistics (mean, standard deviation,
%          and rms) is incorporated
% xypc2rss main program plotting a x-y graph for root sum square (rss) of three
%          specified columns from the difference of two input data files; statistics
%          (mean, standard deviation, and rms) is incorporated
% xypm     main program plotting a x-y graph (with manual scaling) for the selected
%          columns of the input data file
% xyprss   main program plotting a x-y graph (with manual scaling) for root mean
%          square (rss) of three specified columns from an input data file;
%          statistics (mean, standard deviation, and rms) is incorporated
% xyprss2w main program plotting two x-y graphs (with manual scaling) for root mean
%          square (rss) of two sets of three specified columns from an input data
%          file in two windows/subplots; statistics (mean, standard deviation, and
%          rms) is incorporated
% xypvstd  main program plotting a x-y graph (with manual scaling) for a selected
%          column and the associated envelope (standard deviation) column from the
%          specified input file; statistics (mean, standard deviation, and rms) is
%          incorporated
%=====

```

General Purpose and Conventional Kalman Filter Functions

```

% gobsd    generation of observed data (measurements) for a linear time-invariant
%          model; general form including control vector term is included
% gobsd    generation of observed data (measurements) for a linear time-invariant
%          model; the control term and process noise multiplier matrix are not
%          included
% kfcov    covariance matrix analysis for a time-invariant model by using the
%          conventional formulation
% kfcov1   covariance matrix analysis for a time-invariant model by using the

```

```

%      conventional formulation (variant of kfcov, time propagation and
%      measurement incorporation steps are inverted)
% kfcov1a covariance matrix analysis for a time-invariant model by using the
%      alternate conventional formulation
% mdric1  steady state solution of the discrete matrix Riccati equation; covariance
%      matrix after measurement incorporation is determined
% mdric2  steady state solution of the discrete matrix Riccati equation; covariance
%      matrix before measurement incorporation is determined
% meas1cov covariance matrix measurement updating for one measurement by using
%      conventional Kalman formulation (with symmetrization)
% meas1jcov covariance matrix measurement updating for one measurement by using Joseph
%      stabilized Kalman formulation
% meascov  covariance matrix measurement updating for all measurements by using
%      classical Kalman formulation (with symmetrization)
% measjcov covariance matrix measurement updating for all measurements by using
%      Joseph stabilized Kalman formulation
% mndec   decorrelation of the measurement noise
% sdkf   suboptimal (constant gain) discrete Kalman filter by using conventional
%      formulation
% smcov  determination of smoothed covariance matrix based on Rautch-Tung-Striebel
%      algorithm when the model parameters are constant
% smcovps determination of smoothed covariance matrix and state based on Rautch-
%      Tung-Striebel algorithm when the model parameters are constant
%-----
% xgobsd  main program generating the observed data (measurements) for a linear
%      time-invariant model
% xgobsdr main program generating the observed data (measurements) for a simplified
%      linear time-invariant model
% xkfcov  main program executing the covariance analysis by using the conventional
%      or alternate conventional Kalman filter formulation
% xkfcovps main program executing the discrete Kalman filter (covariance and state
%      analysis) by using the conventional Kalman filter formulation
% xmdric  main program computing the steady-state solution of the discrete matrix
%      Riccati equation by using two different iterative methods
% xmndec  main program executing the decorrelation of the measurement noise
% xsdkf  main program computing the suboptimal (constant gain) discrete Kalman
%      filter by using conventional formulation
% xsmcov  main program executing the Rautch-Tung-Striebel smoothing for covariance
%      matrix, when model parameters are constant
% xsmcovps main program executing the Rautch-Tung-Striebel smoothing for covariance
%      matrix and state, when model parameters are constant
%=====
%-----
% Specialized U-D Kalman Filter Functions
% mcud   covariance matrix determination from its U-D factors
% mrlup  updating the U-D factors when a rank-one matrix modification is applied
% mreast measurement reasonableness test for a given scalar measurement
% mudd   U-D factorization of a real symmetric, positive (semi)definite matrix by
%      using modified Cholesky decomposition
% mudm   U-D measurement updating by using Bierman algorithm for one measurement,
%      when the measurement is the input
% mudm1  U-D measurement updating by using Bierman algorithm for one measurement,
%      when the measurement residual is the input
% mudst  standard deviations (sigmas) determination from the U-D factors
% mwgs1  U-D factors determination from the un-normalized W-DW factors (used in
%      the modified weighted Gram-Schmidt algorithm)
% tpudd  time propagation of U-D factors by using the direct method
% tpudgs time propagation of U-D factors by using the modified weighted Gram-
%      Schmidt method
% tpuds  time propagation of U-D factors by using the rank one matrix updating

```

```

%          method
%-----
%  xkfud   main program implementing the discrete U-D form Kalman filter for a
%          specified application. Several options related to the input/output data
%          and selection of variant to be used are available
%  xmuddu  main program executing the decomposition and reconstruction of a real
%          symmetric positive (semi)definite matrix into and from its U-D factors
%  xmudm   main program executing the discrete Kalman filter Bierma's U-D
%          measurement updating algorithm
%  xmudst  main program determining sigmas (standard deviations) of a covariance
%          matrix from its U-D factors
%  xtpud   main program executing time propagation of the U-D factors by using three
%          different methods (direct method, rank one matrix updating method, and
%          modified weighted Gram-Schmidt method)
%=====
%
%          Application Dependent Modules
%  hmat    measurement matrix computation
%  phimat  transition matrix computation
%  qmat    process noise matrix computation
%  rmat    measurement noise matrix computation
%=====
%
%          GPS application modules
%  eleva   elevation angle and the ECEF unit line-of-sight vector computation
%  svpalm  ECEF satellite position determination based on almanac data
%  tgdecef geodetic to ECEF coordinates transformation
%  uverv   unit vertical vector for a given ECEF position vector
%  vecefenu ECEF (Earth Centered Earth Fixed) to ENU (East, North, Up) transformation
%  wgs84con setting of most used WGS-84 constants
%-----
%  xgpsr5s main program performs covariance analysis for the 5-state GPS receiver
%          model (for near-stationary user)
%  xgpsr8s main program performing covariance analysis for the 8-state GPS receiver
%          model (for near-constant velocity user)
%=====
%  Copyright (C) 1992-2008 by L3NAV Systems. All Rights Reserved Worldwide.
%=====

```