'90 KACC 1990. 10. 26~27

Development of a Modern Control Simulation Package : KERICON (I)

* Jung Woo Park, Jae Duck Lee, Kook Hun Kim, and Yo Hee Kim.

Control Applications Laboratory
Korea Electrotechnology Research Institute
P.O. Box 20, Chang Won, Korea.

ABSTRACT

Even though the concept "Adaptive" was introduced in the late 50's, the main contribution to adaptive and/or self-tuning control has been made since late 70's. This paper describes the feature of adaptive control simulation package KERICON(I) developed in KERI.

Informations on hardware environments, install and testing of a new algorithm and user interfacings are also summarized. The package is written in C language and currently being updated for expert-type adaptive control package (KERICON II).

1. Introduction

During 1980's, adaptive control was one of the most prevalent control theory and so many scholars and engineers have contributed for adaptive control theory and its application [10]-[17]. But some debate on the robustness and practicality of adaptive control still remains [6]. The purpose of this work is to develope an adaptive control simulation package which meets one's need and desire. Even though developing such a control simulation program-package was initiated from research interest, SIMNON[1] has been updated for longer than a decade and developed into a semi-commercial package[8]. FAUST is another well known simulation package programmed especially for GPC (generalized predictive control) and currently being updated for hardware interface [9].

Adaptive control simulation package developed by the authors is KERICON(I) and currently being updated for expert-type adaptive control simulation package (KERICON II) and hardware interface. The scope of this paper is restricted to KERICON(I). Installing and /or developing hardware envirenments are described in section 2. The features of KERICON(I) is summerized in section 3. A test run of the program is demonstrated in section 4.

2 Hardware envirenments

IBM-PC(AT/XT) compatible system is enough for the KERICON(I). Details of the necessary system and specifications would be

- main memory more than 640 KB
- graphics card one of VGA, EGA, CGA, or HERCULES
- A/D converter card for external experiments
- co-processor for high speed execution
- color graphics monitor for alphanumeric and graphical output in run time mode

3 General features of KERICON(I)

3.1 How to start-up ?

By the command KERICON(I) on DOS prompt mode, the package is excited. Eight-course main menu is displayed on the screen. After a suitable setting of the parameters, simulation is executed. The start-up procedure is shown in Fig 1. This start-up procedure consists of setting the 8-course main-menu. The 8 items is shown in Fig 2.

3.2 How to follow the items

3.2.1 Procedure 1 : Plant Definition Mode

This is a mode to define an imitative plant in case of making experiments without actual plant. If this package is used for external experiments, this mode can be omitted. For a internal simulation purpose, a simulation model is necessary. At first the order of the transfer function(consisted of numerator, denominator and delay) is chosen, then the corresponding parameters are invoked to be set. Not only discrete but also continuous plant can be used in this stage. The plant definition mode is shown in Fig 2.

3.2.2 Procedure 2 : Model Definition Mode

The purpose of this mode is to choose the order of controller and/or estimator freely. By choosing model order different for procedure 1, reduced or over-parameterized estimation and control is possible. Using this mode, the effects of unmodelled dynamics can be tested.

3.2.3 Procedure 3 :The Initialization Mode for Estimator

This allows estimator to be installed with modeling order and time delay and initial value within the each estimation algorithm [1]-[5]. Whether the system to be estimated is real plant or imitative plant to be set in procedure 1 is decided according to internal or external simulation whose mode is set in procedure 7. estimation algorithms embeded in KERICON(I) is displayed in Fig 3. One important view is that the identification is a function not only of chosen algorithm but also of initial value. Another important elements for a nice estimation is input-output data. But in case of closed loop control, the control designer has no choice with relation to test system.

3.2.4 Procedure 4:The Initialization Mode for Controller

At this mode, the user can choose one of the 4 controller structure. The 4 controllers[5][13] are

- MRAC(model reference adaptive controller)
- PAC(pole assignment controller)
- GPC(generalized predictive controller)
- PID(proportional-integral and derivative controller)

For any one of the above controller, suitable parameters can be invoked for setting of the controller.

3.2.5 Procedure 5 : The Configuration for Input/Output

This mode would be visited only in the case of external experiments. As actual outputs should be scaled in the control algorithm, the package needs some program for identify of actual data and numerical value in the controller. Also setting of maximum/mimimum control input is possible in this mode.

3.2.6 Procedure 6 : Job scheduling

Job scheduling is implemented in this mode. When KERICON(I) is in run time mode, the operation is implemented according to job scheduling. Job scheduling is made with simple commands which include a set of english based mnemonics into actions of the control scheme. E.g., EST,CON,and SET named after estimation, controller,and setting. This mode lets the controller have a soft control , and the estimator prevent from busting. A scheduling is shown in Fig 4.

3.2.7 Procedure 7 : Run

This mode performs checking of entry for the relevant submenus before allowing run time mode to commence. If the setting is not proper, the flow is returned to main menu. If the entry is proper, this allows the package KERICON(I) to enter run time mode. in run time mode, output of estimator, input of control and output of system are displayed in the graphic mode. By pressing RETURN key, the graphics mode returns to initial state, i.e., prompt mode.

3.3.1 Data Entry Facilities

The selection of a main menu option invokes a sub menu option. Data entry in KERICON(I) is simplified by the following features.

1. All data entry is preceded by information which gives the current state of the variables to be modified.

2. Entry of a ENTER key only with no other input will leave the variable unchanged and continue to the next prompt for input.

3. Numeric data is checked for correctness of entry and errors result in a prompt for reentry of the information

4. Entry of numbers is in free_format,ie.a number can be entered in integer, real,or scientific format regardless of the type of the program variable.

3.3.2 The Structure of S/W

KERICON(I) is divided into four modules by its function these are modules connected with estimator and controller, with matrix and polynomial handling, man-machin interface and with settings for user to install main menu.

1) EST ,CON : The modules to embed estimator and controller.

This includes estimation algorithms to find the the system parameters for the plant defined in menu [0], control algorithms correspond to the operation of a perticular self tuning control.

2) MMI : The modules to interface Man_Machine

This involves machine dependent modules and this make use of machine specific coding to cover analogue input/output, data-logging, and terminal manipulation.

3) SMC : The mathematical modules for control

This includes numerical recipies to embody control algorithms for polynomial and matrix handling.

4) NFS : The neutral file related with settings in commisioning mode.

This is a data base made of configuration of control.

3.3.3 Features of KERICON(I)

KERICON(I) has the following useful features.

- Menu-driven commissioning mode.
- Alpha-numeric and graphical real-time display in run time mode.
- Batch job processing
- Internal and external control modes for simulation or real-time control without modification of the package
- Environment save and restore facility for rapid experiment set-up.
- Free-format input of data from Key-board.

4. Simulation

Using the adaptive control package KERICON(I), the simulation was implemented. Some of outputs is shown in [Fig 5] and [Fig 6].

[Sim. 1] MRAC

when a plant is

$$\frac{B(q-1)}{A(q-1)} = \frac{q-1\,(1 - 0.4\,q-1)}{(1 - 0.1\,q-1)\,(1 - 0.9\,q-1)}$$

MRAC was implemented with a model

$$\frac{B(q-1)}{A(q-1)} = \frac{q-1\,(1 - 0.2\,q-1)}{(1 - 0.3\,q-1)} \text{ ; (Gain = 10)}$$

It is shown in Fig 5 that output of plant is following output of model.

[Sim 2] PAC

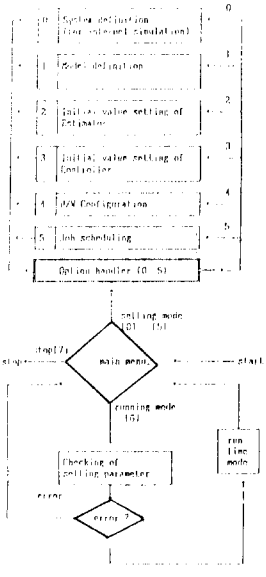
In case that a unstable plant is

$$\frac{B(q-1)}{A(q-1)} = \frac{q-1\,(0.5 + 1.0\,q-1)}{(1 - 0.9\,q-1)\,(1 - 1.1\,q-1)}$$

the simulation with PAC was given for poles of closed-loop system to be assigned at 0.1 , -0.1, and 0.7. The output is shown in Fig 6.

5. Conclusion

KERICON(I) which is a s imulation package for modern control theories is developed. It includes various features with flexibility and provides easiness to use, and test algorithms.



[Fig 1] Main Flow Chart

```
[0]. Plant Definition Mode
[1]. Model Definition Mode
[2]. The Initialization Mode For Estimator
[3]. The Initialization Mode For Controller
[4]. The Configuration Mode For Input/Output
[5]. Job Scheduling
[6]. Run
[7]. Quit
```

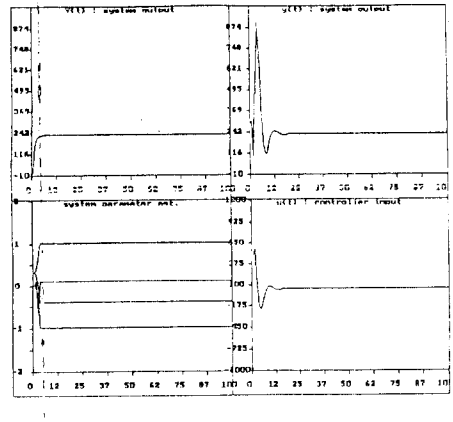
[Fig 2] Main Menu

```
[0]. Orthogonalized projection algorithm
[1]. Least squares with selective data weighting
[2]. Least squares with exponential data weighting
[3]. Least squares with covariance resetting
[4]. Least squares with covariance modification
[5]. QUIT
```

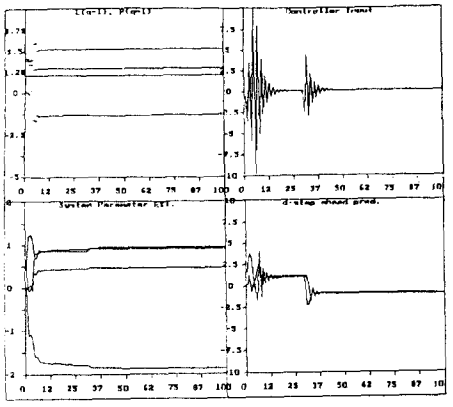
[Fig 3] The Initialization Mode For Estimator

0 STEP 10			
1 EST ON	CON OFF		
2 EST ON	CON ON	SET 1.0	
3 EST ON	CON ON	SET -1.0	
4 EST OFF	CON ON	SET -1.0	
5 EST OFF	CON OFF		

[Fig 4] Job Scheduling



[Fig 5] Model Reference Adaptive Controller



[Fig 6] Pole Assignment Controller

5.0 REFERENCES

1) K.J. Astrom and B.Wittenmark, "Computer Controled Systems", Englewood Cliffs, N.J.,Prentice-Hall,1984.

2) K.J. Astrom and B.Wittenmark, "Adaptive Control" ,Addison Wesley,1989.

3) L. Lung and T. Soderstrom, "Theory and Practice of Recursive Identification" ,The MIT Press,1983.

4) S. Sastry and M. Bodson, "Adaptive Control", Prentice Hall,1989.

5) G.C. Goodwin and K.S. Sin, "Adaptive Filtering Prediction and Control", Prentice Hall,1984.

6) C.E. Rohrs, "Adaptive Control in The Presence of Unmodelled Dynamics", Ph.D. Thesis,1982.

7) Automatic Control Group of S.N.U., "Adaptive Control Strategies", 1987.

8) H. Elmqvist, K.J. Astrom, and T. Schonthal, "SIMNON User's Guide for MS-DOS Computers", Lund, Technology, 1986.

9) P.S. Tuffs and D.W.Clarke,"Faust : A Software Package For Self-tuning Control",IEE Conf., 1985.

10) N.Rao Sripada and D. Grant Fisher, "Improved Leas Squares Identification for Aaptive Controllers", Proc. of the American Control Conference, Vol 3, PP.2027-2037,1987.

11) R.Kulhavy, "Restricted Exponential Forgetting i Real-Time Identification", IFAC, YORK,U.K., PP.1143-1148,1985.

12) K.J.Astrom and B.Wittenmark, "On Self Tuning Regulators", Automatica,Vol.9, pp.185-199,1973.

13) F.Cameron and D.E.Seborg,"A Self-tuning Controller with a PID Controller",con. pp. 401-417,1982.

14) D.W.Clark, C.Mohtadi and P.S.Tuffs, "Generalized Predictive Control - part I. The Basic Algorithm", automatica, pp.137-148, 1987.

15) P. Stoica, A. Nehorai, and S.m. Kay , "Statistical Analysis of the Least Squares Autoregressive Estimator in the Presence of Noise", IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. ASSp-35, No. 9, pp. 1273-1281,1987.

16) K.C. Shet and B.V. Rao, "Modified Fast Kalman Algorithm", IEEE Trans. on Acoustics, Speech, and Signal Processing,VOL.ASSP-35,NO.7, pp.1072-1075,1987.

17) M.S. Armed, "Fast GLS Algorithms for parameter Estimation", Automatica,Vol.20, No 2, pp.231-236,1984.

18) E. Karlsson and M.H. Hayes, "Least squares ARMA Modeling of Linear Time-Varying Systems : Lattice Filter Structures and Fast RLS Algorithms",IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. ASSP-35, NO. 7, pp. 994-1014,1987.