20/82
ME
I-5428

①

dr. 827

UCID- 19551

MASTER

SATER/LLNL:
AN INTERACTIVE
IDENTIFICATION/ESTIMATION/CONTROL
PACKAGE

J. V. Candy
and
C. H. Journeay

September 1, 1982

This is an informal report intended primarily for internal or limited external distribution. The opinions and conclusions stated are those of the author and may or may not be those of the Laboratory.

Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore Laboratory under Contract W-7405-Eng-48.

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

UCID--19551

DE82 021009

ABSTRACT

This issue of the computer program newsletter focuses on SATER/LLNL--an interactive code for identification, estimation, and classical control. A brief description of the SATER/LLNL operation is included as well as a sample session. Attached is the fundamental reference that details more of the package.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

1.0 INTRODUCTION

In this newsletter we discuss SATER/LLNL--a system identification/parameter estimation package. SATER/LLNL is an interactive, menu-driven code on small computers which enables the user to identify or "fit" single-input/single-output (SISO), discrete, input-output models to noisy data. The package enables the user to interactively simulate, display, manipulate, identify, and analyze noisy data. It also offers the capability to analyze the performance of the various estimators.

SATER is described in [1]. There is an enormous amount of technology encompassed by this package. In this newsletter, we will concentrate on the identification/parameter estimation features, since that is the most powerful and unique portion of the package.

SATER/LLNL solves the identification problem which can simply be stated as (see Fig. 1):

GIVEN a set of noisy measurement and input data,
FIND a linear model that "fits the data in some
optimal fashion.

The solution to this problem comes from the field of statistics. More specifically, it is an estimation problem which can be formulated mathematically and solved using a variety of algorithms in SATER/LLNL. In the next series of sections, we describe the package and give a sample problem.

GIVEN a set of noisy measurements $\{y(t)\}$ and inputs $\{u(t)\}$, **FIND** the minimum (error) variance estimate $\hat{\theta}(t)$ of $\theta(t)$, a vector of unknown parameters of linear system.

Figure 1. The Identification Problem.

2.0 PURPOSE

SATER/LLNL is designed to interactively identify transfer function models from noisy measurements. The need to perform an identification occurs in control problems, where models are required prior to controller design. SATER/LLNL can be used for the design of filter/estimators to remove undesired noise from data, since the identification solution is basically the optimal filtered (Wiener) solution. The package contains several estimation options such as Recursive Least Squares (RLS), Generalized Least Squares (GLS), Extended Least Squares (ELS), Instrumental Variables (IV), Recursive Maximum Likelihood (RML), and the Tally Method (TM). It also contains some options for the analysis of classical control systems such as Bode plots, Nyquist plots, root locus, and stability tests. So, one can consider SATER a signal estimation tool to solve the system identification/parameter estimation problem, or a classical control system design package which can be used to obtain a pulse transfer function model for digital control.

3.0 AUTHORS AND EVOLUTION

SATER was obtained from the Eindhoven University of Technology, Eindhoven, The Netherlands. The package has been under development since 1975. It is a primary educational and research tool used by the Electrical Engineering Department at the University. The primary developers of the algorithm were A. J. Van den Boom and W. J. Lemmens [1,2], as well as P. Eykhoff [3] and A. W. Van den Enden [4]. SATER initially executed on a DEC PDP-11/20 computer with RT-11, 16K core and RK05 backup memory.

SATER was brought to Lawrence Livermore National Laboratory in 1981 by J. Candy, where it was modified and redesigned by C. Journeay to execute on a DEC PDP-11/70 operating under IAS and RSX-11M. SATER/LLNL is substantially different in structure from the original version; however, most of the initial coding has remained. The package is multitasked, since it has over 10,000 lines of FORTRAN coding. SATER/LLNL has been enhanced from the original SATER version with the inclusion of further identification algorithms, input-output features, estimator performance analysis, state space conversion options, discrete-to-continuous domain conversions, and diagnostic outputs, as well as communication with other LLNL signal processing algorithms such as ANALYZE [5]. All of the code development for SATER/LLNL has been supported by the Engineering Research Division of the Electrical Engineering Department. SATER/LLNL is currently executed on the EE/VAX-11/780 computer operating under VMS.

4.0 DESCRIPTION

SATER/LLNL solves the identification problem for SISO discrete systems. It is based on a specific input-output model which is related to the pulse transfer function given by

$$H(z) = z^{-d} B(z^{-1}) / A(z^{-1}) \quad (1)$$

where d is the delay, A and B are polynomials in z or z^{-1} ,

$$A(z^{-1}) = 1 + a_1 z^{-1} + \dots + a_{Na} z^{-Na} \quad ,$$

$$B(z^{-1}) = b_0 + b_1 z^{-1} + \dots + b_{Nb} z^{-Nb} \quad .$$

If we consider the equivalent time domain representation, then we have a difference equation relating the output sequence $\{y(t)\}$ to the input sequence $\{u(t)\}$. Here we use the backward shift operator q with the property that $q^{-k} y(t) = y(t-k)$.

$$A(q^{-1}) y(t) = q^{-d} B(q^{-1}) u(t)$$

or

$$y(t) - a_1 y(t-1) - \dots - a_{Na} y(t-Na) = b_0 u(t-d) + b_1 u(t-d-1) + \dots + b_{Nb} u(t-d-Nb) \quad (2)$$

When the system is driven by random inputs, the models are given by the autoregressive-moving average model with exogenous inputs (ARMAX)

$$A(q^{-1}) y(t) = q^{-d} B(q^{-1}) u(t) + C(q^{-1}) / D(q^{-1}) e(t) \quad (3)$$

where A , B , C , D are polynomials, as before, and $\{e(t)\}$ is white.

SATER/LLNL identifies or estimates the coefficients of the ARMAX model from noisy data, either simulated internally within or from actual experimental measurements. With this in mind, we can more specifically define the identification/parameter estimation problem in terms of the ARMAX model as

GIVEN a set of noisy input $\{u(t)\}$, output $\{y(t)\}$ measurements from an unknown system, FIND the minimum (error) variance estimate of the unknown parameters, $\{a_i\}$, $\{b_i\}$, $\{c_i\}$, and $\{d_i\}$, which characterize the ARMAX model of (3) above.

A typical recursive parameter estimation algorithm using the ARMAX model with $D(q^{-1}) = 1$, is given in Table 1 below. The algorithm is the Recursive Maximum Likelihood (RML) which gives estimates of the corresponding parameters of the $A(q^{-1})$, $B(q^{-1})$, and $C(q^{-1})$ polynomials. Note that we are actually modeling the noise in the system as colored when we use the C , D polynomials and we are simultaneously estimating the corresponding coloring filter. These algorithms are called Recursive Prediction Error Methods (RPEM)(see Ljung [6] for details) because they minimize the one-step prediction error defined below in Table 1. Most of the algorithms in SATER/LLNL are special cases of the RML algorithm. For example, if we set $\psi(t) = \phi(t)$ and remove the filtering ($C = 1$), then we obtain the Recursive Extended Least Squares algorithm (RELS). If we do not estimate the $C(q^{-1})$ polynomial at all, then we have the Recursive Least Squares (RLS) algorithm. As shown in Ljung [6], all of the other algorithm are special cases of the RPEM, which for the ARMAX model is the RML algorithm shown in the table. In

Table 1. Recursive Maximum Likelihood (RML)

PREDICTION ERROR:

$$e(t) = y(t) - \hat{y}(t|t-1) = y(t) - \phi'(t)\hat{\theta}(t-1) \quad [\text{Prediction Error}]$$

$$R_e(t) = \psi'(t)P(t-1)\psi(t) + \lambda(t) \quad [\text{Prediction Error Variance}]$$

UPDATE:

$$\hat{\phi}(t) = \hat{\phi}(t-1) + \mu K(t) e(t) \quad [\text{Parameter Update}]$$

$$\bar{e}(t) = y(t) - \bar{\phi}'(t) \hat{\theta}(t) \quad [\text{Residual Update}]$$

$$P(t) = (I - K(t)\psi'(t)) P(t-1) / \lambda(t) \quad [\text{Covariance Update}]$$

$$\lambda(t) = \lambda_0 \lambda(t-1) + (1-\lambda_0) \quad [\text{Forgetting Factor Update}]$$

$$k(t) = \beta k(t-1) + (1-\beta) \quad [\text{Contraction Factor Update}]$$

FILTERING:

$$\tilde{y}(t,i) = y(t-i) - [\hat{C}(k(t)q^{-1}) - 1] \tilde{y}(t,i) \quad [\text{Measurement Filter}]$$

$$\tilde{u}(t,i) = u(t-i) - [\hat{C}(k(t)q^{-1}) - 1] \tilde{u}(t,i) \quad [\text{Input Filter}]$$

$$\tilde{\tilde{e}}(t,i) = \bar{e}(t-i) - [\hat{C}(k(t)q^{-1}) - 1] \tilde{\tilde{e}}(t,i) \quad [\text{Residual Filter}]$$

GAIN:

$$K(t) = P(t-1)\psi(t)R_e^{-1}(t) \quad [\text{Gain or Weight}]$$

$$\bar{\phi}'(t) = [-y(t-1) \dots -y(t-Na) \mid u(t-d) \dots u(t-d-Nb) \mid \bar{e}(t) \dots \bar{e}(t-Nc)] \quad [\text{Phi-Update}]$$

$$\psi'(t) = [-\tilde{y}(t-1) \dots -\tilde{y}(t-Na) \mid \tilde{u}(t-d) \dots \tilde{u}(t-d-Nb) \mid \tilde{\tilde{e}}(t) \dots \tilde{\tilde{e}}(t-Nc)] \quad [\text{Gradient Update}]$$

these other algorithms different approximations of the gradient are used. A detailed discussion of the RPEM can be found in Ljung [6,7]. This completes the basic description of the identification/parameter estimation algorithms employed in SATER/LLNL.

SATER/LLNL includes many capabilities. The main menu of the package is shown in Fig. 2. We see the following major categories addressed with (option):

1. Identification [Order Estimation(13)].
2. Parameter Estimation [RELS(3), GLS(10), TM(11), IV(12), ML(17), RLS(25), RML(26)].
3. Estimator Performance Analysis [Response(20), Pole/Zero(2,18)].
4. Simulation [Process-Noise Parameters(7), Input-Noise(8), Simulator(4)].
5. Classical Control [Nyquist(1), Bode(5), Root Locus(16)].
6. Input/Output/Display [Data Input(14), Data Display(21), Data Output(ANALYZE)(6)].
7. Conversion [Discrete-Continuous(9), ARMAX-State Space(23), Continuous(19)].
8. Diagnostics [Internal Files(22)].

SATER/LLNL CAPABILITIES INCLUDE:

1. NYQUIST DIAGRAM
2. DISCRETE POLE/ZERO DIAGRAM
3. PARAMETER ESTIMATION EXTENDED MATRIX METHOD
4. SAMPLED SIMULATION OF OUTPUT SIGNAL
5. LOGARITHMIC CONVERSION TO BODE PLOT
6. WRITE SELECTED SATER SIGNALS TO FILES IN 'ANALYZE' FORMAT
7. MANUAL INPUT OF DISCRETE PROCESS PARAMETERS
8. MANUAL INPUT OF INPUT SIGNAL PARAMETERS
9. Z-S CONVERSION TO CONTINUOUS-TIME PROCESS PARAMETERS
10. PARAMETER ESTIMATION GENERALISED LEAST SQUARES
11. PARAMETER ESTIMATION TALLY (P.K.F)
12. PARAMETER ESTIMATION INSTRUMENTAL VARIABLE METHOD
13. ORDER ESTIMATION
14. READ EXTERNAL SAMPLED INPUT/OUTPUT SIGNALS
15. *** NOT USED ***
16. ROOT LOCUS DIAGRAM
17. PARAMETER ESTIMATION MAXIMUM LIKELIHOOD METHOD
18. CALCULATION OR MANUAL INPUT OF CONTINUOUS POLE/ZEROS
19. CALCULATION OR MANUAL INPUT OF CONTINUOUS PROCESS PARAMETERS
20. ESTIMATED RESPONSE SIMULATION WITH NOISE
21. PLOT OF SAMPLED INPUT/OUTPUT DATA
22. PRINTOUT OF INTERNAL SATER DATA FILES
23. CONVERT ARMAX MODEL TO STATE-SPACE MODEL
24. SELF-TUNING PREDICTOR/REGULATOR
25. RECURSIVE LEAST SQUARES ESTIMATION
26. RECURSIVE MAXIMUM LIKELIHOOD ESTIMATION

Figure 2. SATER/LLNL Main Menu.

The first three list elements are the Identification/Estimation/Analysis/Display options which enable the user to solve the identification problem, estimate coefficients of the ARMAX model and analyze/display the performance of the estimator. The fourth list element is the ARMAX simulator, and the fifth is the classical control options. The final set of options involve input/output and conversions.

5.0 LIMITATIONS

SATER was originally designed to identify a tenth-order (10th) system with 21 total parameters, and many of the options are limited to that order. SATER/LLNL was modified to identify a total of 100 parameters, but unfortunately at present all of the options have not been modified to reflect that order. Most of the parameter estimators and performance analysis options will operate on a 100 parameter total. SATER will run on small minicomputers with 32K of memory.

SATER is limited to linear systems.

6.0 AVAILABILITY

SATER is written in FORTRAN-IV and is available on DEC PDP-11/20, PDP-11/60, PDP-11/70 operating on IAS and DEC VAX-11/780 operating on VMS. SATER and SATER/LLNL are proprietary, but they are available for internal LLNL usage.

7.0 STORAGE REQUIREMENTS

SATER/LLNL requires 32K of addressable memory on the DEC PDP-11 minicomputer systems, but since all of the algorithms are recursive, most of the operations are written and read off disk. The SATER/LLNL is a multitasked, heavily overlaid package which calls in small subtasks to perform the necessary operations.

8.0 TIMING

Timing is very problem dependent, but since the algorithms are recursive, they do not require much CPU time; however, the interaction time is extensive because much of the time is spent doing input/output operations on a disk. It is important to the SATER/LLNL user that the computer system being utilized is either dedicated or allows a reasonable amount of user interaction. Thus, the system should not be heavily burdened with tasks that "capture" the CPU.

9.0 SAMPLE PROBLEM

In this section, we selected a sample identification problem that is a default in the SATER/LLNL package. We will go through all the steps, after we simulate the data set, that one would normally go through to perform an "identification."

In practice, identification is accomplished on data by:

1. Preprocessing the noisy measurements (e.g., filter, decimate, etc.).
2. Testing for the order (N_a) of the process and noise (N_d).
3. Performing the parameter estimation.
4. Validating the identified model (prediction error tests).
5. Validating the "model" on other independent measurements (ensemble).

In this sample problem, we will simulate the "noisy measurement" using the SATER/LLNL simulator (options 4, 7, 8). We chose the internal SATER example given by

$$(1 - 1.5q^{-1} + 0.7q^{-2})y(t) = (q^{-1} + 0.5q^{-2})u(t) + (1 + 0.3q^{-1})/(1 - 0.5q^{-1}) e(t)$$

(4)

where $u(t)$ and $e(t)$ are pseudo-random binary and uniform white noise respectively. The simulated data is shown in Fig. 3. The $u(t)$, a pseudo-random binary signal, is superimposed on top of the noisy measurement of the process.

A series of order tests were performed using the SATER/LLNL order testing options. Two such tests are shown in Fig. 4. In Fig. 4a, we see a nonparametric test based on calculating a particular loss function. The plot is loss function value versus model order. The objective is to find the portion of the curve that is a minimum or flatens out [8,9]. In Fig. 4a, we see the curve is flat at a third ($N_a=3$) and we even see a local minimum at order two (2). These results are not unreasonable, since this particular test has no way of differentiating between the process order (N_a) and coloring filter order (N_d). In our problem, $N_a + N_d = 3$. The second order test is shown in Fig. 4b. It is a parametric test that enables separate testing of process order (N_a) and noise order (N_d). Here the test is to search for pole (x) - zero (0) cancellations. When order is overestimated, the excess pole-zero pairs will cancel. Examining Figure 4b, we see that the process order is $N_a = 1$, since the excess pole-zero pairs in order 3 through 5 cancel. The coloring filter order (N_d) is also estimated as $N_d = 1$ (see [1]).

Once the order is estimated, the next step is to perform the parameter estimation. We choose the ELS algorithm, so that we can estimate the coloring filter (C, D-polynomials). We select $N_a = 2$, $N_b = 3$, $N_c = N_d = 1$ [1]. The ELS estimator was run and the estimated parameters are shown in Fig. 5a along with the parameter trajectories.

NUMBER 1 OUTPUT DATA
NUMBER 2 INPUT DATA

PRESS RETURN TO CONTINUE

SOURCE FILES:
INPUT - 22 OPTION 4 22
OUTPUT - 22 OPTION 4 22

FILE 'SIGN DAY.1' DATA
SAMPLE PERIOD - 0.10000E+01
AMPLITUDE FACTOR - 0.75000E+01

Figure 3. Simulated Noisy Measurement with Pseudo-Random Input.

(a)

PRESS RETURN TO CONTINUE

ERROR-FUNCTION 18061011 (WOODSIDE)

ORDER	VALUE	1ST DIFFERENCE	2ND DIFFERENCE
0	3.63810		
1	2.81630	-0.818799	-1.13238
2	0.865149	-1.95116	1.78886
3	0.702849	-0.162300	0.152377
4	0.692347	-0.970878E-02	-0.418597E-02
5	0.678858	-0.140887E-01

(b)

POLE-ZERO PLOT OF ESTIMATED PROCESS MODEL

PRESS RETURN TO CONTINUE

Figure 4. Order Testing: (a) Nonparametric Test, (b) Pole-Zero Test.

Figure 5. Parameter Trajectories (a) and Estimated Response (b).

After the parameters are estimated, the performance of the estimator is investigated. The "fit" is analyzed by first overlaying the estimated output found by driving the estimated process with the input u , i.e.,

$$\hat{y}(t) = [1 - \hat{A}(q^{-1})] \hat{y}(t) + q^{-d} \hat{B}(q^{-1}) u(t). \quad (5)$$

The results are shown in Fig. 5b. We see that the fit appears quite reasonable.

To validate the fit we must perform the prediction error tests, i.e., it has been shown that the prediction error is to be zero-mean and white, if the fit is adequate. The prediction error is shown in Fig. 6a and the corresponding whiteness test is shown in 6b. The whiteness test requires that 95% of the samples lie within the two-sigma bounds (dotted lines) for the sequence under test to be statistically white.

This completes the sample problem.

NOISE MEAN VALUE
NOISE STANDARD DEVIATION
PRESS RETURN TO CONTINUE

-0.154841E-01
0.142340

PLOT OF AUTOCORRELATION FUNCTION OF $X(k)$
DATA SET USED FROM OPERATION 3

STARTING SAMPLE 30

PERCENT OUT OF BOUNDS - 0.000001

Figure 6. Model Validation Prediction Error Tests:
(a) Errors, (b) Whiteness Test.

10.0 REFERENCES

- [1] A. van den Boom, "The Interactive Program Package SATER for Estimation and Control in Education and Research," Journal A, Vol. 20, 1979.
- [2] W. Lemmens and A. van den Boom, "Interactive Computer Programs for Education and Research: A Survey", Automatica, Vol. 15, 1979.
- [3] P. Eykhoff, System Identification, Parameter and State Estimation, London: Wiley, 1974.
- [4] A. van den Boom and A. van den Enden, "The Determination of the Orders of Process and Noise Dynamics," Automatica, Vol. 10, 1974.
- [5] S. Azevedo, "ANALY7E---Lecture Notes," LLNL course notes on State Space Lectures, CEC-2018H, 1982.
- [6] L. Ljung and T. Soderstrom, Theory and Practice of Recursive Identification, MIT Press: Boston, 1982.
- [7] L. Ljung, "Analysis of a General Recursive Prediction Error Identification Algorithm," Automatica, Vol. 17, 1982.
- [8] C. Woodside, "Estimation of the Order of Linear Systems," Automatica, Vol. 7, 1971.
- [9] H. Unbehauen and B. Gohring, "Tests for Determining Model Order in Parameter Estimation," Automatica, Vol. 10, 1974.

DISCLAIMER

This document was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor the University of California nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government thereof, and shall not be used for advertising or product endorsement purposes.

Printed in the United States of America
Available from
National Technical Information Service
U.S. Department of Commerce
5285 Port Royal Road
Springfield, VA 22161
Price: Printed Copy \$; Microfiche \$3.50

<u>Page Range</u>	<u>Domestic Price</u>	<u>Page Range</u>	<u>Domestic Price</u>
001-025	\$ 5.00	326-350	\$ 18.00
026-050	6.00	351-375	19.00
051-075	7.00	376-400	20.00
076-100	8.00	401-425	21.00
101-125	9.00	426-450	22.00
126-150	10.00	451-475	23.00
151-175	11.00	476-500	24.00
176-200	12.00	501-525	25.00
201-225	13.00	526-550	26.00
226-250	14.00	551-525	27.00
251-275	15.00	526-550	28.00
276-300	16.00	601-up ¹	
301-325	17.00		

¹ Add 2.00 for each additional 25 page increment from 601 pages up.

Technical Information Department • Lawrence Livermore Laboratory
University of California • Livermore, California 94550