Some Computational Results Obtained by Panuska's Method of Stochastic Approximations for Identification of Discrete Time Systems

Gustavsson, Ivar; Valis, Jaroslav

1969

Document Version:
Publisher's PDF, also known as Version of record

SOME COMPUTATIONAL RESULTS OBTAINED BY PANUŠKA'S METHOD OF STOCHASTIC APPROXIMATIONS FOR IDENTIFICATION OF DISCRETE TIME SYSTEMS

JAROSLAV VALIS
IVAR GUSTAVSSON

REPORT 6915 JUNE 1969
LUND INSTITUTE OF TECHNOLOGY
DIVISION OF AUTOMATIC CONTROL

SOME COMPUTATIONAL RESULTS OBTAINED BY PANUŠKA'S METHOD OF STOCHASTIC APPROXIMATIONS FOR IDENTIFICATION OF DISCRETE TIME SYSTEMS. †

J. Valis and I. Gustavsson

ABSTRACT

In this paper different variants of Panuška's method of stochastic approximations for system identification are used to estimate the discrete transfer function coefficients from simulated second order system input/output data sequences of finite length. Using Monte Carlo simulation mean values and variances of the estimates are computed and compared with those obtained using the method of maximum likelihood. Results confirm the superiority of M.L. estimates in this case.

† This work has been supported by the Swedish Board for Technical Development under Contract 68-336-f

1. A BRIEF OUTLINE OF PANUŠKA'S METHOD

In his papers [2], [3] V Panuška proposes a very simple method (perhaps the most simple method which could ever be used) for numerical identification of linear dynamic discrete time system of finite order with constant coefficients from normal operating records. Description of such a system can always be reduced to Åström's canonical form

$$y(k) + \sum_{i=1}^{n} a_i\, y(k-i) = \sum_{i=0}^{n} b_i\, u(k-i) + \lambda\Big(e(k) + \sum_{i=1}^{n} c_i\, e(k-i)\Big) \qquad (1)$$

where $\{y(k)\}$ is the sequence of values of output signal
$\{u(k)\}$ is the sequence of values of input signal
$\{e(k)\}$ is a sequence of normal independent random variables with zero mean and variance 1 ($N(0,1)$)
n is the order of the system

Panuška supposes, that finite length records of sequences

$$\{u(t), y(t);\ t = 1, 2, \ldots, N\}$$

are given and that the order of the system n is known. Then he uses the following formulas for stochastic approximation of estimates of $\hat{a}_i$, $\hat{b}_i$, $\hat{c}_i$ and $\hat{\lambda}$.

$$k = 1 + (t - 1) \bmod N \qquad (2a)$$

$$\hat{\varepsilon}(k) = y(k) + \sum_{i=1}^{n} \hat{a}_i(k)\, y(k-i) - \sum_{i=0}^{n} \hat{b}_i(k)\, u(k-i) - \sum_{i=1}^{n} \hat{c}_i(k)\, \hat{\varepsilon}(k-i) \qquad (2b)$$

$$G(t) = \alpha/t;\ \alpha \text{ suitable positive constant} \qquad (2c)$$

$$\hat{a}_r(t+1) = \hat{a}_r(t) - G(t)\cdot \hat{\varepsilon}(k)\cdot y(k-r) \qquad r = 1, \ldots, n \qquad (2d)$$

$$\hat{b}_r(t+1) = \hat{b}_r(t) + G(t)\cdot \hat{\varepsilon}(k)\cdot u(k-r) \qquad r = 0, 1, \ldots, n \qquad (2e)$$

$$\hat{c}_r(t+1) = \hat{c}_r(t) + G(t)\cdot \hat{\varepsilon}(k)\cdot \hat{\varepsilon}(k-r) \qquad r = 1, 2, \ldots, n \qquad (2f)$$

$$\hat{\lambda}^2 = \frac{1}{N} \sum_{k=1}^{N} \hat{\varepsilon}^2(k) \qquad (2g)$$

In [2] Panuška states without complete proof that the sequences $\hat{a}_r(t)$, $\hat{b}_r(t)$ and $\hat{c}_r(t)$ converge in the mean square sense to the true values of a_r, b_r and c_r respectively. He also gives some results of identification of artificially simulated data. Unfortunately, the paper does not contain complete answers to some important questions concerning variance, speed of convergence, stability of the computation, etc.

2. DISCUSSION OF THE ALGORITHM AND VARIANTS

Panuška's algorithm may have the following interpretation [Fig. 1].

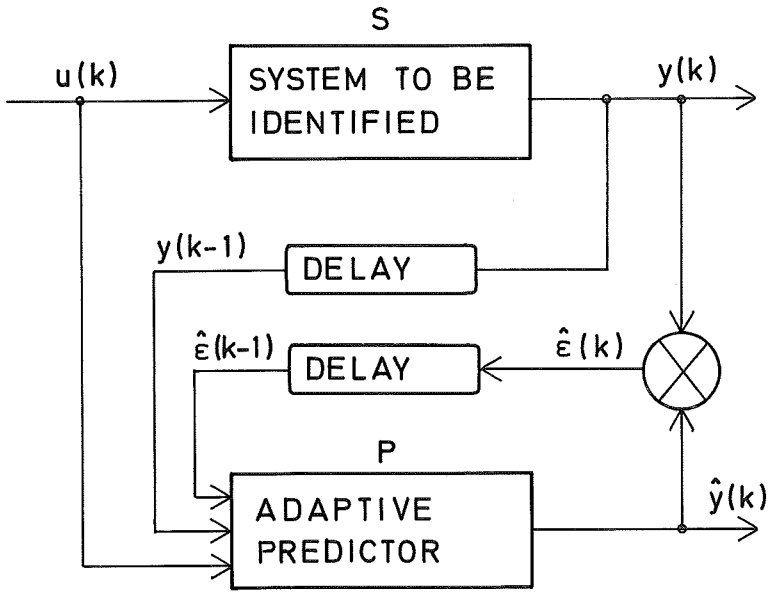

Fig. 1. Interpretation of Panuška's algorithm.

Let us assume, that the system S may be described by eq. (1). We want to adjust the coefficients of the adaptive predictor P, which predicts the value of output $\hat{y}(k)$ based on past values of input $u(k)$, $u(k-1)$, $u(k-2)$, ..., $u(k-n)$, past values of output $y(k-1)$, $y(k-2)$, ..., $y(k-n)$, and past prediction errors $\hat{\varepsilon}(k-1)$, $\hat{\varepsilon}(k-2)$, ..., $\hat{\varepsilon}(k-n)$ in such a way, that the prediction errors $\hat{\varepsilon}(k)$, defined by

$$\hat{y}(k) = \sum_{i=0}^{n} \hat{b}_i(k)\, u(k-i) - \sum_{i=1}^{n} \hat{a}_i(k)\, y(k-i) + \sum_{i=1}^{n} \hat{c}_i(k)\, \hat{\varepsilon}(k-i) \qquad (3)$$

$$\hat{\varepsilon}(k) = y(k) - \hat{y}(k) \qquad (4)$$

are „as small as possible".

Exactly, it means that the $E\hat{\varepsilon}^2(k)$ should be minimal and the $\{\hat{\varepsilon}(k)\}$ sequence should be white.

Our „adjust strategy" will be simply to minimize at each time instant k the square of instant prediction error $\hat{\varepsilon}(k)$ using gradient method (neglecting the whole past history). It means

$$\hat{\varepsilon}(k) = y(k) - \hat{y}(k) = y(k) + \sum_{i=1}^{n} \hat{a}_i(k)\, y(k-i) - \sum_{i=0}^{n} \hat{b}_i\, u(k-i) -$$

$$- \sum_{i=1}^{n} \hat{c}_i(k)\hat{\varepsilon}(k-i) \qquad (5)$$

and

$$\frac{\partial \hat{\varepsilon}^2(k)}{\partial \hat{a}_r(k)} = 2\,\hat{\varepsilon}(k)\cdot y(k-r) \qquad r = 1,2,\ldots,n \tag{6a}$$

$$\frac{\partial \hat{\varepsilon}^2(k)}{\partial \hat{b}_r(k)} = -2\,\hat{\varepsilon}(k)\cdot u(k-r) \qquad r = 0,1,\ldots,n \tag{6b}$$

$$\frac{\partial \hat{\varepsilon}^2(k)}{\partial \hat{c}_r(k)} = -2\,\hat{\varepsilon}(k)\,\hat{\varepsilon}(k-r) \qquad r = 1,2,\ldots,n \tag{6c}$$

Putting (5) and (6a, b, c) into the general formula for the gradient minimization method, we get (2b - g).

Remark 1:

The computation of $\hat{\varepsilon}(k)$ using formula (2b) may easily become unstable if the polynomial

$$\hat{C}_t(z^{-1}) = z^n\left(1 + \sum_{i=1}^{n} z^{-i}\hat{c}_i(k)\right) \tag{7}$$

has zeroes outside of the unit circle. Panuška avoids this difficulty by putting certain bounds on values of estimates $\hat{a}_r(k)$, $\hat{b}_r(k)$, $\hat{c}_r(k)$ and errors $\hat{\varepsilon}(k)$. That means (q is suitable positive constant) for a-coefficients

$$\hat{a}_r(t+1) = \begin{cases} +q & \text{if } |\tilde{a}_r(t+1)| \geq q \\ \tilde{a}_r(t+1) & \text{if } |\tilde{a}_r(t+1)| < q \\ -q & \text{if otherwise} \end{cases} \tag{8}$$

where $\tilde{a}_r(t+1) = \hat{a}_r(t) - G(t)\cdot\hat{\varepsilon}(t)\,y(t-r)$ and similar for $\hat{b}_r(t)$, $\hat{c}_r(t)$ and $\hat{\varepsilon}(t)$.

Another way to assure the stability of the computation of $\hat{\varepsilon}(k)$ is to change the values of $\hat{c}_r(t)$ so that polynomial $\hat{C}_t(z^{-1})$ remains stable. Then we have to check the stability of the polynomial (7) after each iteration step. This is relatively easy to do in first and second order system cases but for $n > 2$ we must use a suitable numerical method for stability check because of the large number of iteration steps.

Remark 2:

To obtain better estimates, Panuška uses several runs through the input-output records so that the values of estimates after the first pass are used as the starting values for the second pass and so forth. This is a very good idea, but in our opinion still better accuracy can be achieved when using

$$G(t) = \frac{\alpha}{t} \tag{9a}$$

or

$$G(t) = \frac{\alpha}{\beta + k}\ , \quad \beta >> 1 \tag{9b}$$

instead of (2c) as a gain factor.

3. EXPERIMENTAL INVESTIGATION OF PANUŠKA'S METHOD

The papers [2], [3] contain results obtained by means of described method on identification of a 2nd order system

$$y(k) - 1.5y(k-1) + 0.7y(k-2) = u(k-1) + 0.5u(k-2) + \varepsilon(k) +$$

$$+ \varepsilon(k-1) + 0.2\varepsilon(k-2)$$

where $\{u(k)\}$ and $\{\varepsilon(k)\}$ are generated as independent pseudorandom sequences N(0,1). The results obtained from 500 input/output data pairs are very promising, but unfortunately the variance of these estimates cannot be found from only very few realisations.

To obtain better insight into properties of this method, we first wrote simple programs for simulation of a linear discrete-time system driven by pseudorandom normal signal and its identification from input/output records. We tried both proposed ways to assure the stability of computation of $\hat{\varepsilon}(k)$:

1) Original Panuška's with values of estimates of coefficients $\hat{a}$, $\hat{b}$, $\hat{c}$ bounded to ±5.0 and errors $\hat{\varepsilon}(k)$ bounded to 10.0. This algorithm was called ORIGPAN.

2) Checking the stability of $\hat{C}_t(z^{-1})$ polynomial at each iteration step - at unstable iterations we did not change the estimates. This algorithm was called STABPAN.

In both algorithms different gain sequences G(t) were used. In each example we computed mean values and variance of estimates by Monte Carlo method (from 20 different samples). Starting values for estimates and initial conditions were zero.

Numerical results are scheduled in Tab. 1. Surprising are large variances and bad estimates of c-coefficients (particularly c_2) in all cases when compared with result obtained by Åström's method of maximum likelihood (which can be considered as the most powerful method for identification at present time [1] , [4]).

Not surprising is the (relative) superiority of ORIGPAN when compared with STABPAN because even if the $\hat{C}$-polynomial is stable the values of $\hat{\varepsilon}$ may become very large and cause completely wrong changes of estimates.

For further comparison we chose the variant, which gave the most reasonable results (ORIGPAN, $\alpha = 5$, 5 passes denoted by f in Tab. 1) and tried it on two different samples of our 2nd order standard data. In this case the input signal was pseudorandom normal (0,1), length of sequences $N = 1000$ data pairs. Results, summarized in Tab. 2, seem not to be bad in comparison with M.L.E., but unfortunately the estimates obtained after 5th pass through the data cannot be considered as final because they show distinct trend to change further when we increase the number of passes as it is shown in Tab. 3.

The reason for this is quite obvious:

> According to (2d - 2f) the changes of estimates of coefficients shall continue while
>
> $$\frac{1}{t} \cdot \hat{\varepsilon}(k) \cdot \hat{\varepsilon}(k - r) \neq 0 \quad (r = 1, 2, \ldots, n)$$
>
> by other words the changes shall continue until the sequence $\{\hat{\varepsilon}(k)\}$ becomes such that the first n-values of its autocovariance function
>
> $$\hat{r}_{\varepsilon\varepsilon}(\tau) = \lim_{t\to\infty} \frac{1}{t} \sum_{i=1}^{t} \hat{\varepsilon}(i)\, \hat{\varepsilon}(i - \tau) \qquad \tau = 1, 2, \ldots, n$$
>
> are zero. In the case of input/output data of infinite length the sequences of estimates $\{\hat{a}(k)\}\{\hat{b}(k)\}$, $\{\hat{c}(k)\}$ would therefore converge to their true values. Apparently this may not be the case if we perform iterations by repeated passes through the same data of finite length N because this is equivalent to the case when the original sequence of (generated) errors $\{\varepsilon(k)\}$ is periodical with period N and the values of its autocovariance function
>
> $$r_{\varepsilon\varepsilon}(\tau) = \frac{1}{N} \sum_{t=1}^{N} \varepsilon(t)\, \varepsilon(t - \tau)$$
>
> may not be zero.

Actually, in our example of 2nd order standard data this sequence was not „very white" as it is shown in Tab. 4 and Fig. 2, where the estimates and covariances of the respective residuals are scheduled for different numbers of passes.

The algorithm ORIGPAN tries to make the values of $r^{\wedge\wedge}_{\varepsilon\varepsilon}(1)$ and $r^{\wedge\wedge}_{\varepsilon\varepsilon}(2)$ as small as possible (even if the original ones are not) without taking other values of $r^{\wedge\wedge}_{\varepsilon\varepsilon}$ into account while the autocovariance function of residuals of M.L. estimates has almost the same shape as the covariance of original noise.

To illustrate this we designed two more examples with different noise sequences and plotted the results in Fig. 3, 4 resp. Tab. 5. The „more white" sequence S_2 gave better estimate than the S_1 with relatively large values of $r_{\varepsilon\varepsilon}(1)$ and $r_{\varepsilon\varepsilon}(2)$.

4. CONCLUSIONS

It follows from the previous investigation that Panuška's method of stochastic approximations can be considered as interesting with the following advantages:

1. simplicity - at present time no more simple method is known
2. easy programmable - a computer subroutine contains a few statements only
3. suitable for real-time identification (see Appendix)

Unfortunately, the disadvantages are rather serious:

1. suitable gain sequence must be found experimentally
2. large numbers of passes through the data are needed to obtain more accurate results
3. the variances of estimates are greater than those obtained by Åström's M.L.E.
4. there is no measure of accuracy of estimates available.

BIBLIOGRAPHY:

[1] Gustavsson, I.: Parametric Identification of Multiple Input, Single Output Linear Dynamic Systems,Report 6907, July,1969, Lund Institute of Technology, Division of Automatic Control.

[2] Panuška, V.: A Stochastic Approximation Method for Identification of Linear Systems Using Adaptive Filtering, 1968, JACC.

[3] Panuška, V.: Some Numerical Results of Identification of Linear Systems by a Stochastic Approximation Technique, in „Advances in Control of Systems", 3rd UKAC Control Convention, University of Leicester, England, April, 1968.

[4] Åström, K.J., Bohlin, T.: Numerical Identification of Linear Dynamical Systems from Normal Operating Records, in „Theory of Self-Adaptive Systems, P.H. Hammond (ed.), Plenum Press, 1966.

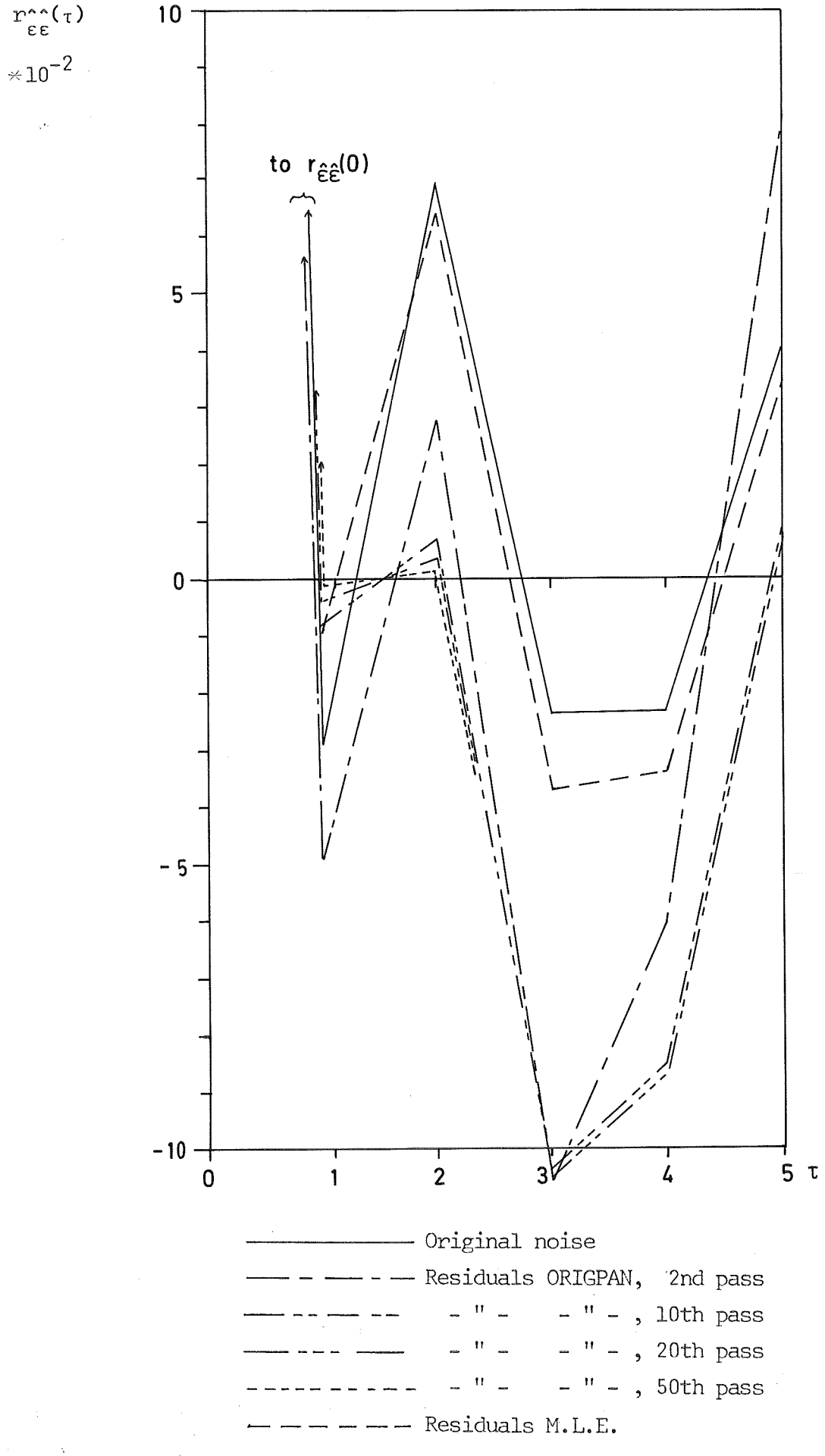

Fig. 2 Covariance of Original Noise and Residuals
2nd order standard data

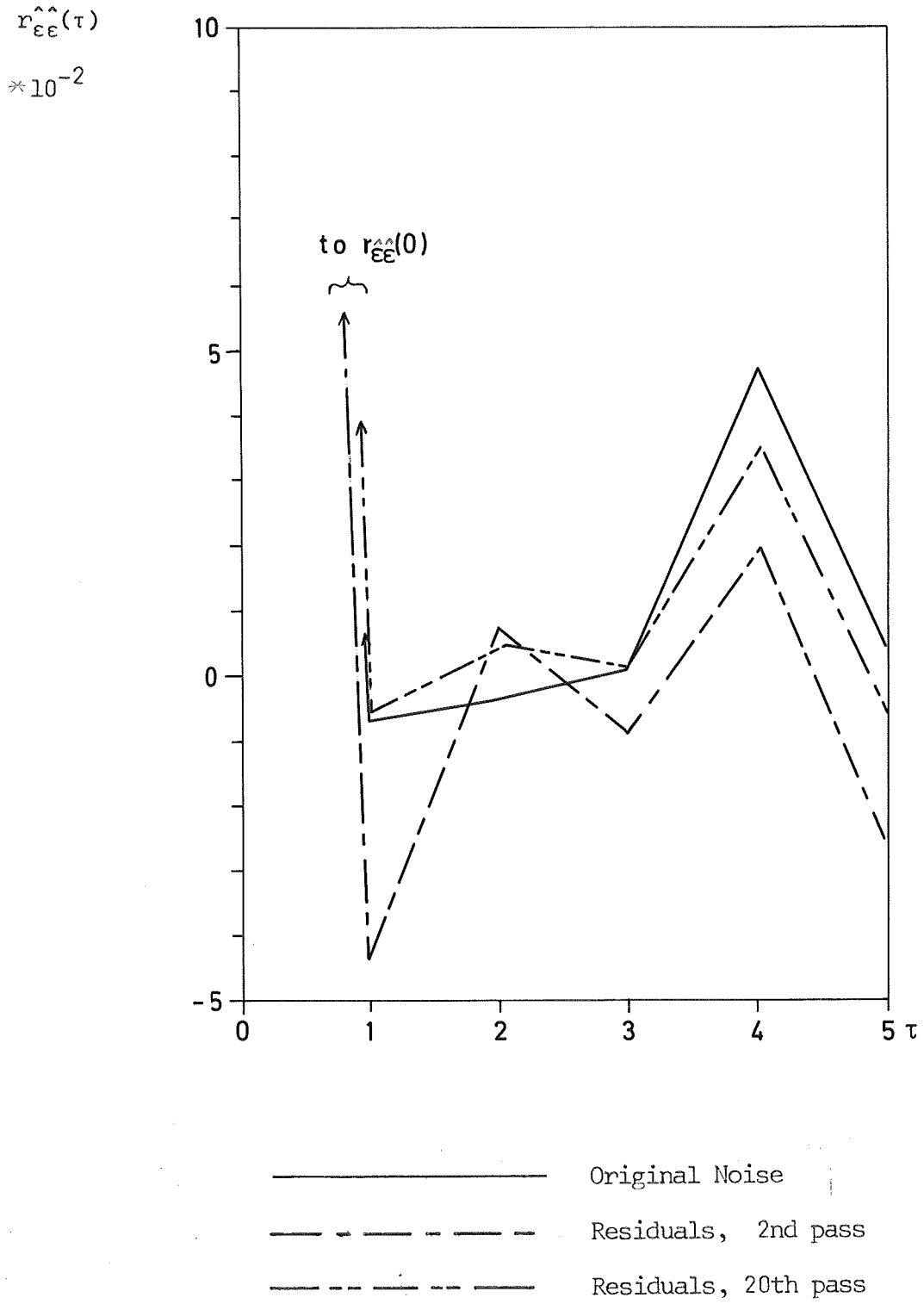

Fig.3 2nd Order Standard System Sample S_2
Covariance of Original Noise and Residuals

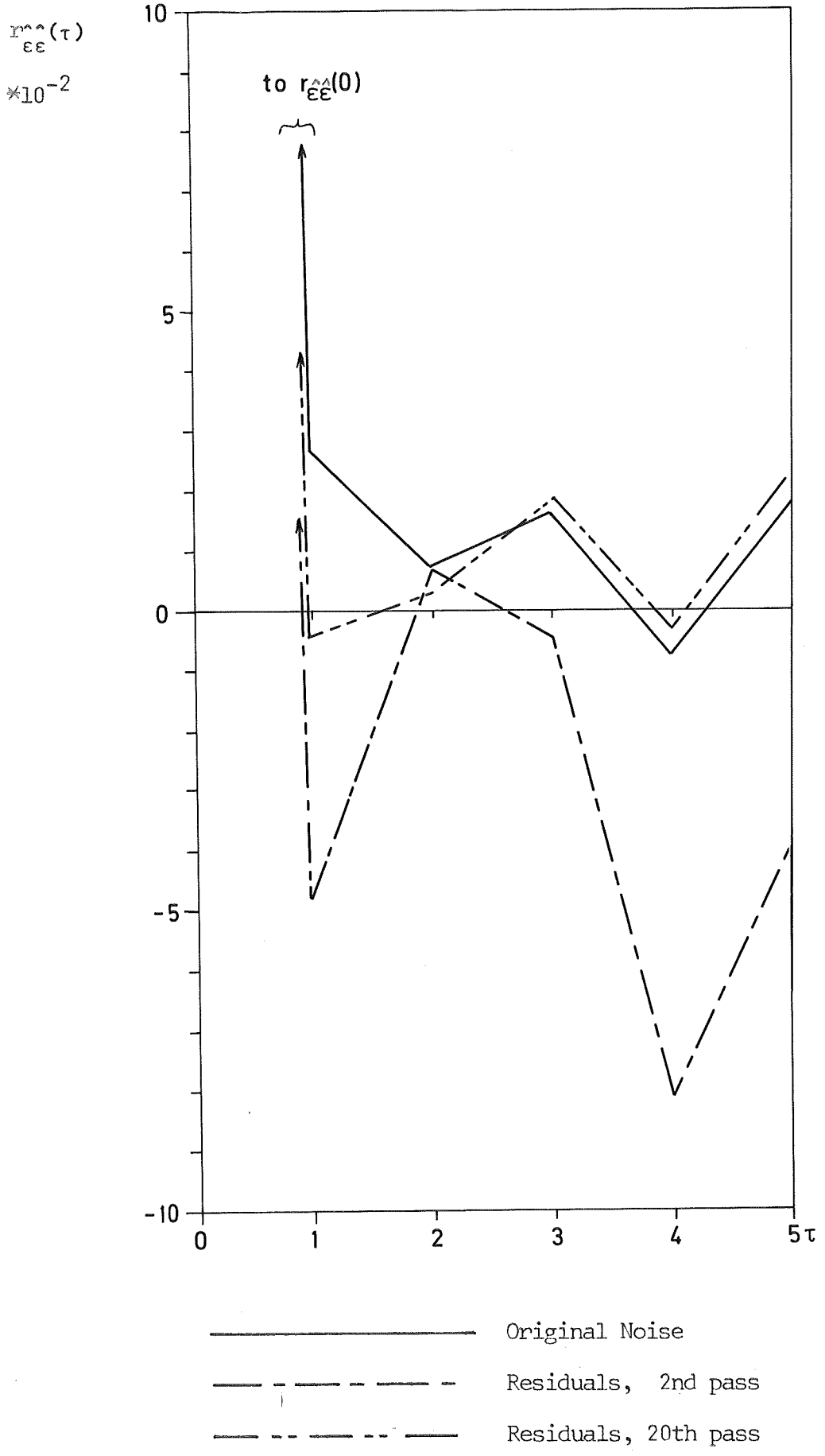

Fig. 4 2nd Order Standard System, Sample S_1
Covariance of Original Noise and Residuals

Tab. 1.

Input Signal Pseudorandom Gaussian, (0,1) generated by RANSS with starting value 1

Noise: Pseudorandom Gaussian, generated by RANSS with starting value 3

Length of record N = 500

Bounds: Coeff ± 5.0

Errors ±10.0

Example mark	Variant of algorithm		Gain seq.				Results - all scheduled values are sample means from 20 different realisations.											
	ORIGPAN	STABPAN	$G_t=\frac{a}{k}$	$G_t=\frac{a}{t}$	a	Number of passes	b_1		b_2		a_1		a_2		c_1		c_2	
							1.000		1.000		-1.500		0.700		-1.000		0.200	
							$\hat{b}_1$	±δ	$\hat{b}_2$	±δ	$\hat{a}_1$	±δ	$\hat{a}_2$	±δ	$\hat{c}_1$	±δ	$\hat{c}_2$	±δ
a	X		X		2.5	1	0.737	.19	1.116	.30	-1.422	.08	0.643	.078	-0.543	.12	-0.034	.07
b	X		X		2.5	5	0.668	.23	1.205	.27	-1.407	.08	0.626	.086	-0.533	.14	-0.156	.10
c	X		X		5.0	1	0.631	.39	1.197	.32	-1.258	.30	0.653	.35	-0.408	.25	0.035	.18
d	X		X		5.0	5	0.650	.49	1.175	.41	-1.252	.22	0.655	.40	-0.336	1.26	0.004	.20
e	X			X	2.5	5	0.895	.05	1.069	.075	-1.487	.02	0.682	.018	-0.825	.045	0.062	.06
f	X			X	5.0	5	0.873	.04	1.060	.074	-1.481	.024	0.667	.02	-0.785	.043	0.040	.06
g		X	X		0.5	5	0.725	.37	1.305	.37	-1.736	1.7	0.258	1.35	-0.164	.49	0.119	.21
h		X	X		1.0	5	0.735	.62	1.421	.98	-1.384	1.22	0.615	.29	-0.220	.67	0.119	.31
i		X	X		2.5	5	$7*10^3$	$3*10^4$	1.005	1.8	$-5.5*10^3$	$2*10^4$	0.425	3.8	-0.619	.47	0.163	.30
j		X	X		5.0	5	N O N S E N S E											
k		X		X	2.5	5	1.036	.65	1.026	.22	-1.474	.05	0.674	.04	-0.797	.08	0.048	.05
l		X		X	5.0	5	1.947	1.34	0.935	.43	-2.268	3.5	0.569	0.4	-0.785	.2	0.053	.16
m	X		$G_t=\frac{a}{2000+K}$		5.0	5	0.860	.04	1.073	0.05	-1.479	.02	0.668	.006	-0.776	.012	0.035	.02

Tab. 2

Input Signal: Pseudorandom Gaussian (0,1) generated by RANSS
Noise: Pseudorandom Gaussian (0,1) generated by RANSS
Length of record N = 1000

Sample q on cards
Sample r on tape 18

Sample	Used method	b_1		b_2		a_1		a_2		c_1		c_2		Computing time in S
		1.000		0.500		-1.500		0.700		-1.000		0.200		
		$\hat{b}_1$	$\pm\delta$	$\hat{b}_2$	$\pm\delta$	$\hat{a}_1$	$\pm\delta$	$\hat{a}_2$	$\pm\delta$	$\hat{c}_1$	$\pm\delta$	$\hat{c}_2$	$\pm\delta$	
q	ORIGPAN 1 pass	1.043		0.644		-1.296		0.591		-0.593		0.075		-
	ORIGPAN 5 pass	1.064		0.498		-1.459		0.678		-0.948		0.226		11
	M.L.E.	1.018	.032	0.476	.042	-1.508	.009	0.708	.008	-1.022	.031	0.233	.03	≈30
r	ORIGPAN 1 pass	1.002		0.646		-1.494		0.673		-0.729		0.032		-
	ORIGPAN 5 pass	0.999		0.534		-1.491		0.704		-0.945		0.219		17
	M.L.E.	1.008	.03	0.518	.039	-1.503	.009	0.703	.008	-1.006	.031	0.231	.031	≈30

Tab. 3

Original Panuška with bounds
N = 1000 2nd order standard data sample q
BE = 10.00 BC = 5.00
ALFA = 5.00

PASS NO. 1 COEF 1.0431+000	6.4419-001	-1.2961+000	5.9090-001	-5.9337-001	7.5014-002
PASS NO. 2 COEF 1.0670+000	5.5026-001	-1.3950+000	6.4703-001	-8.0823-001	1.5418-001
PASS NO. 3 COEF 1.0691+000	5.1778-001	-1.4285+000	6.6109-001	-8.8998-001	1.8526-001
PASS NO. 4 COEF 1.0671+000	5.0428-001	-1.4468+000	6.7027-001	-9.2766-001	2.0888-001
PASS NO. 5 COEF 1.0639+000	4.9775-001	-1.4586+000	6.7771-001	-9.4832-001	2.2567-001
TRUE: $b_1 = 1.000$	$b_2 = 0.500$	$a_1 = -1.500$	$a_2 = 0.700$	$c_1 = -1.000$	$c_2 = 0.200$

Tab. 4

2nd Order standard data, sample q, N = 1000												
	b_1	b_2	a_1	a_2	c_1	c_2	$r_{\varepsilon\varepsilon}(0)$	$r_{\varepsilon\varepsilon}(1)$	$r_{\varepsilon\varepsilon}(2)$	$r_{\varepsilon\varepsilon}(3)$	$r_{\varepsilon\varepsilon}(4)$	$r_{\varepsilon\varepsilon}(5)$
Original system	1.000	0.500	-1.500	0.700	-1.000	0.200	1.01320	-0.02981	0.06935	-0.02303	-0.02293	0.04137
ORIGPAN 2nd pass	1.067	0.550	-1.395	0.647	-0.808	0.154	1.49710	-0.04949	0.02798	-0.10575	-0.05912	0.08046
ORIGPAN 10th pass	1.051	0.490	-1.483	0.697	-0.986	0.262	1.0828	-0.00769	0.00701	-0.10308	-0.08477	0.00853
ORIGPAN 20th pass	1.040	0.489	-1.494	0.706	-1.004	0.280	1.0552	-0.00388	0.00354	-0.10464	-0.08607	0.00433
ORIGPAN 50th pass	1.032	0.489	-1.501	0.707	-1.015	0.293	1.0394	-0.00127	0.00164	-0.10480	-0.08493	0.00523
M.L.E.	1.018	0.476	-1.508	0.708	-1.022	0.233	1.02392	-0.01003	0.06479	-0.03748	-0.03399	0.03542

Tab. 5

Second order standard system

Input: Pseudorandom Gaussian (0,1) generated by RANSS with starting value 3 (both samples)

Noise: Pseudorandom Gaussian (0.1) generated by RANSS. Sample S_1 with starting value 5
Sample S_2 with starting value 1

Length: N = 1000

	b_1	b_2	a_1	a_2	c_1	c_2	$r_{\varepsilon\varepsilon}(0)$	$r_{\varepsilon\varepsilon}(1)$	$r_{\varepsilon\varepsilon}(2)$	$r_{\varepsilon\varepsilon}(3)$	$r_{\varepsilon\varepsilon}(4)$	$r_{\varepsilon\varepsilon}(5)$
Original sample S_1	1.000	0.500	-1.500	0.700	-1.000	0.200	0.94431	0.02674	0.00639	0.01609	-0.00793	0.01812
ORIGPAN 2nd pass	1.024	0.503	-1.500	0.681	-0.838	0.063	1.31524	-0.04784	0.00659	-0.00461	-0.08108	-0.03867
ORIGPAN 20th pass	0.982	0.459	-1.519	0.722	-0.945	0.161	1.00728	-0.00402	0.00313	0.01936	-0.00305	0.02352
Original sample S_2	1.000	0.500	-1.500	0.700	-1.000	0.200	0.98573	-0.00786	-0.00325	0.00065	0.03824	0.00485
ORIGPAN 2nd pass	0.912	0.552	-1.491	0.664	-0.781	0.032	1.37403	-0.04381	0.00759	-0.00827	0.01986	-0.02558
ORIGPAN 20th pass	1.024	0.446	-1.509	0.699	-0.999	0.196	1.01185	-0.00588	0.00492	0.00120	0.03534	-0.00489

Appendix A.

Programs, Subroutines and Functions used for Computation

A 1.	RANSS	Subroutine
A 2.	BOUND	Real function
A 3.	ORIGPAN	Preliminary versions used in early
A 4.	STABPAN	trials (for results in Tab. 1)
A 5.	ORIGPAN 2	
A 6.	RTPAN with subroutine	
	RETORPA	Real time version of ORIGPAN
A 7.	GACF	Subroutine

A 1. RANSS (KK, RR)

used for generation of noise, resp. input signal sequences.

Title	Random Floating Point Numbers in Normal Distribution (RANSS)
CO-OP Class/Index Code	G5 WISC RANSS
CO-OP Organ. Code	WISC (adapted from CODA RNDEV)
Program Language	CODAP-1/COMPASS
Computer	1604/3600
Programmer	RNDEV by Wm Silverman, adapted by Richard G. Wolfe for the University of Wisconsin Computing Center
Contributing Organ.	University of Wisconsin Computing Center (Formerly Numerical Analysis Laboratory)
Date	February 1963
UWCC ID Code	C0010-00/S0010-00

1.0 PURPOSE

To generate random floating point numbers distributed according to the Normal (Gauss) distribution with a mean of 0 and a variance of 1. The range of the generated distribution is between plus and minus 8.

2.0 USAGE

An odd integer between 0 and 2^{43} is assigned by the User prior to the first call to RANSS, and this number determines the sequence of deviates that will be generated. This odd integer is changed by the subroutine to prepare for the next call and generation.

2.1 Calling Sequence.

2.1.1 FORTRAN-63 Calling Sequence: CALL RANSS (KK,RR)

2.2 Arguments Or Parameters.

KK is the odd integer assigned as above.

RR will be the generated deviate.

2.4 Space Required.

1604	3600
27_8 locations	41_8 locations

RANSS - 2 of 2
29 Jan 65

2.12 Timing.

1604	3600
approximately	approximately
1400 microseconds	380 microseconds

3.0 METHOD

The number *KK* is multiplied by 5^{13} and reduced modulo 2^{43}. The process is repeated 16 times and the results summed. The numbers obtained after each multiplication and reduction have many of the characteristics of uniformly distributed random variables which take their values in the odd integers between 0 and 2^{43}. If we call the ith such variable X_i and put $S_n = \sum_1^n X_i$, then the following result is implied by the Central Limit Theorem:

$$\lim_{n\to\infty} P\left\{\frac{S_n - nM}{\sigma\sqrt{n}} \leq X\right\} = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \exp(-t^2/2)\,dt$$

where M is the mean of the uniform variable X, and $M = 2^{42}$, and σ^2 is the variance of X, and $\sigma^2 = \frac{1}{3}(2^{42}-1)^2$ and the right hand term is the distribution function of a normally distributed variable with mean 0 and variance 1. Asymptotic considerations show that for $n \geq 16$ the approximation is sufficiently accurate for single precision arithmetic. So $n = 16$ is used with $16M = 2^{46}$ and $\sigma\sqrt{16} = \frac{4(2^{42}-1)}{\sqrt{3}}$.

A 2. BOUND (X, B)

This real function is used to limit the value of estimates of coefficients and residuals to certain value B.

The value of this function is equal to

X when $|x| \leq B$

B • sign X when $|x| > B$

FTN5.48 18/02-09

```
      REAL FUNCTION BOUND(X,B)
      IF(ABSF(X).LE.B) GO TO 1
      IF(X) 2,1,3
    1 BOUND=X $ RETURN
    2 BOUND=-B $ RETURN
    3 BOUND=B $ RETURN
      END
```

A 3. ORIGPAN - used in early trials for results in Tab. 1

Requires: Subroutine RANSS
Real function BOUND

```
      PROGRAM ORIGPAN
      DIMENSION U(510),Y(510),E(510),A(10),B(10),C(10),EA(10),EB(10),
     1EC(10),PC(10),BAC(20,9),EBAC(9),VBAC(9)
      REAL BOUND
      BE=10. $ BC=5.
      READ 100,NN
      PRINT111,NN
  111 FORMAT(*SIMULATION*/*LENGTH   *I5)
      READ 101,N
      READ 102,(B(I),I=1,N)
      READ 102,(A(I),I=1,N)
      READ 102,(C(I),I=1,N)
      READ 101,NE
      PRINT 107,(B(I),I=1,N)
      PRINT109,(A(I),I=1,N)
      PRINT110,(C(I),I=1,N)
  100 FORMAT(I4)
  101 FORMAT(I3)
  102 FORMAT(10F4.1)
  103 FORMAT(F4.1)
      ICOUNT = 1
      ALFA = 2.5
      PRINT 112,ALFA
  112 FORMAT(*ALFA=*F6.2)
      IT1=1$IT2=3
      DO 38 IREA=1,20
      DO 21 I=1,NN
      CALL RANSS(IT1,U(I+10))
      CALL RANSS(IT2,E(I+10))
   21 CONTINUE
      DO 22 I=1,10
   22 U(I)=Y(I)=E(I)=0.
      DO 23 K1=1,NN
      K=K1+10$EE=E(K)
      DO 24 I=1,N
      J=K-I
   24 EE=EE+B(I)*U(J)+C(I)*E(J)-A(I)*Y(J)
      Y(K)=EE
   23 CONTINUE
      PRINT 113,IREA
  113 FORMAT(*IDENTIFICATION NO.*I4)
      DO 26 I=1,NE
   26 EB(I)=EA(I)=EC(I)=0.
      DO 45 KK=1,ICOUNT
      E(11)=0.
      DO 25 K1=2,NN
      K=K1+10 $ EE=Y(K)
      G=ALFA/((KK-1)*NN+K1)
      DO 27 I=1,NE
      J=K-I$ EE=EE+EA(I)*Y(J)-EB(I)*U(J)-EC(I)*E(J)
   27 CONTINUE
      E(K)=BOUND(EE,BE)
      G=G*E(K)
      DO 28 I=1,NE
      J=K-I
      EC(I)=BOUND(EC(I)+G*E(J),BC)
```

FTN5,4B 18/07-68

```
      EB(I)=BOUND(EB(I)+G*U(J),BC)
   28 EA(I)=BOUND(EA(I)-G*Y(J),BC)
   25 CONTINUE
   30 PRINT 104,KK
      PRINT 1000,((EB(I),I=1,NE),(EA(I),I=1,NE),(EC(I),I=1,NE))
 1000 FORMAT(*COEF*9E12.4)
   45 CONTINUE
      DO 39 I=1,NE
      I1=I+NE $ I2=I+2*NE
      BAC(IREA,I)=EB(I)
      BAC(IREA,I1)=EA(I)
   39 BAC(IREA,I2)=EC(I)
   38 CONTINUE
      ND=3*NE
      DO 41 I=1,ND
      SUM=0.
      DO 40 J=1,20
   40 SUM=SUM+BAC(J,I)
   41 EBAC(I)=SUM/20.
      PRINT 1001,(EBAC(I),I=1,ND)
 1001 FORMAT(/*EXP=*9E12.4)
      DO 42 I=1,ND
      DO 42 J=1,20
   42 BAC(J,I)=BAC(J,I)-EBAC(I)
      DO 43 I=1,ND
      SUM=0.
      DO 44 J=1,20
   44 SUM=SUM + BAC(J,I)*BAC(J,I)
   43 VBAC(I)=SUM/20.
      PRINT 1002,(VBAC(I),I=1,ND)
 1002 FORMAT(*VAR=*9E12.4)
  104 FORMAT(*PASS NO.*I3)
  107 FORMAT(*B=*(E12.5))
  109 FORMAT(*A=*(E12.5))
  110 FORMAT(*C=*(E12.5))
      CALL EXIT
      END
```

A 4. STABPAN - used in early trials for results in Tab. 1.

Requires: Subroutine RANSS

28/05-65

```
      PROGRAM STABPAN
      DIMENSION U(510),Y(510),E(510),A(10),B(10),C(10),EA(10),EB(10),
     1EC(10),PC(10),BAC(20,9),EBAC(9),VBAC(9)
      BE=10. $ BC=5.
      READ 100,NN
      PRINT 111,NN
  111 FORMAT(*SIMULATION*/*LENGTH   *I5)
      READ 101,N
      READ 102,(B(I),I=1,N)
      READ 102,(A(I),I=1,N)
      READ 102,(C(I),I=1,N)
      READ 101,NE
      PRINT 107,(B(I),I=1,N)
      PRINT 109,(A(I),I=1,N)
      PRINT 110,(C(I),I=1,N)
  100 FORMAT(I4)
  101 FORMAT(I3)
  102 FORMAT(10F4.1)
  103 FORMAT(F4.1)
      ICOUNT=5
      ALFA=.
      PRINT 112,ALFA
  112 FORMAT(*ALFA=*F6.2)
      IT1=1$IT2=3
      DO 38 IREA=1,20
      DO 21 I=1,NN
      CALL RANSS(IT1,U(I+10))
      CALL RANSS(IT2,E(I+10))
   21 CONTINUE
      DO 22 I=1,10
   22 U(I)=Y(I)=E(I)=0.
      DO 23 K1=1,NN
      K=K1+10$EE=E(K)
      DO 24 I=1,N
      J=K-I
   24 EE=EE+B(I)*U(J)+C(I)*E(J)-A(I)*Y(J)
      Y(K)=EE
   23 CONTINUE
      PRINT 113,IREA
  113 FORMAT(*IDENTIFICATION NO.*I4)
      DO 26 I=1,NE
   26 EB(I)=EA(I)=EC(I)=0.
      DO 45 KK=1,ICOUNT
      E(11)=0.
      DO 25 K1=2,NN
      K=K1+10 $ EE=Y(K)
      G=ALFA/((KK-1)*NN+K1)
      DO 27 I=1,NE
      J=K-I$ EE=EE+EA(I)*Y(J)-EB(I)*U(J)-EC(I)*E(J)
   27 CONTINUE
      E(K)=EE
      G=G*E(K)
      DO 46 I=1,NE
      J=K-I
   46 PC(I)=EC(I)+G*E(J)
      IF(ABS(PC(2)).LT.1..AND.ABS(PC(1)).LT.1.+PC(2)) 47,25
```

28/06-06

```
   47 DO 28 I=1,NE
      J=K-I
      EC(I)=PC(I)
      EB(I)=EB(I)+G*U(J)
   28 EA(I)=EA(I)-G*Y(J)
   25 CONTINUE
   30 PRINT 104,KK
      PRINT 1000,((EB(I),I=1,NE),(EA(I),I=1,NE),(EC(I),I=1,NE))
 1000 FORMAT(*COEF*9E12.4)
   45 CONTINUE
      DO 39 I=1,NE
      I1=I+NE $ I2=I+2*NE
      BAC(I,EA,I)=EB(I)
      BAC(I,EA,I1)=EA(I)
   39 BAC(I,EA,I2)=EC(I)
   38 CONTINUE
      ND=3*NE
      DO 41 I=1,ND
      SUM=0.
      DO 40 J=1,20
   40 SUM=SUM+BAC(J,I)
   41 EBAC(I)=SUM/20.
      PRINT 1001,(EBAC(I),I=1,ND)
 1001 FORMAT(/*EXP=*9E12.4)
      DO 42 I=1,ND
      DO 42 J=1,20
   42 BAC(J,I)=BAC(J,I)-EBAC(I)
      DO 43 I=1,ND
      SUM=0.
      DO 44 J=1,20
   44 SUM=SUM + BAC(J,I)*BAC(J,I)
   43 VBAC(I)=SUM/20.
      PRINT 1002,(VBAC(I),I=1,ND)
 1002 FORMAT(*VAR=*9E12.4)
  104 FORMAT(*PASS NO.*I3)
  107 FORMAT(*B=*(E12.5))
  109 FORMAT(*A=*(E12.5))
  110 FORMAT(*C=*(E12.5))
      CALL EXIT
      END

ARY DECK***
,1500, , ,
```

A 5. ORIGPAN 2

Complete program used for computation of Tab. 2 and Tab. 3.

Requires Subroutine GACF[1)]
Real function BOUND

[1)]Remark: The use of subroutine GACF for computation of covariance of residuals after each pass through the data is not vital and may be omitted by skipping the respective statements at the end of program, i.e.

```
      CALL GACF(...
      PRINT 888,(...
  888 FORMAT(...
```

5.48 18/02-09

```
      PROGRAM ORIGPAN 2
C     NUMERICAL IDENTIFICATION OF LINEAR DISCRETE TIME SYSTEM FROM NORMAL
C     OPERATING RECORDS USING PANUSKAS METHOD OF STOCHASTIC APPROXIMATIONS
C     (V.PANUSKA  A STOCHASTIC APPROXIMATION METHOD OF IDENTIFICATION OF
C     LINEAR SYSTEMS USING ADAPTIVE FILTERING, DEPT. OF ENG. SC. OXFORD
C     UNIVERSITY, OXFORD, ENGLAND   SUBMITTED TO 1968 JACC CONFERENCE)
C
C     MODIFIED AND PROGRAMMED BY JAROSLAV VALIS  12.6.1968 IN INSTITUT FOR
C     REGLERINGSTEKNIK TH LUND
C
C       BE   BOUND FOR ERRORS
C       BC   BOUND FOR COEFFICIENTS
C       NN   NUMBER OF DATA PAIRS (MAX 1000)
C       U    INPUT SEQUENCE
C       Y    OUTPUT SEQUENCE
C       NE   ASSUMED ORDER OF THE SYSTEM
C       ICOUNT  NUMBER OF PASSES THROUGH THE DATA
C
C     THIS PROGRAM REQUIRES THE REAL FUNCTION BOUND(X,B)
C
      DIMENSION U(1010),Y(1010),E(1010),EA(10),EB(10),EC(10)
       DIMENSION ACVF(21)
      REAL BOUND
      BE=10. $ BC=5.
      READ 100,NN
      PRINT 111,NN,BE,BC
  111 FORMAT(*ORIGINAL PANUSKA WITH BOUNDS*/*NN=*I5,/*  BE=*F5.2,
     1*  BC=*F5.2)
  100 FORMAT(I4)
       ICOUNT = 20
      ALFA = 5.
      PRINT 112,ALFA
  112 FORMAT(*ALFA=*F6.2)
      DO 22 I=1,10
   22 U(I)=Y(I)=E(I)=0.
      NNN=NN+10
      READ 999,(U(I),Y(I),I=11,NNN)
  999 FORMAT(10F8.3)
      READ 102, NE
  101 FORMAT(10F8.3)
  102 FORMAT(I3)
      DO 26 I=1,NE
   26 EB(I)=EA(I)=EC(I)=0.
      DO 45 KK=1,ICOUNT
      DO 25 K1=1,NN
      K=K1+10 $ EE=Y(K)
      G=ALFA/((KK-1)*NN+K1)
      DO 27 I=1,NE
      J=K-I $ EE=EE+EA(I)*Y(J)-EB(I)*U(J)-EC(I)*E(J)
   27 CONTINUE
      E(K)=BOUND(EE,BE)
      G=G*E(K)
      DO 28 I=1,NE
      J=K-I
      EC(I)=BOUND(EC(I)+G*E(J),BC)
      EB(I)=BOUND(EB(I)+G*U(J),BC)
```

```
      28 EA(I)=BOUND(EA(I)-G*Y(J),BC)
      25 CONTINUE
      30 PRINT 104,KK
     104 FORMAT(*PASS NO.*I3)
         PRINT 1000,((EB(I),I=1,NE),(EA(I),I=1,NE),(EC(I),I=1,NE))
          CALL GACF(E,1010,11,1010,ACVF,11,21)
          PRINT 888,(ACVF(I),I=1,11)
     888 FORMAT(/*AUTOCOVARIANCE FUNCTION OF RESIDUALS*/10E12.4)
    1000 FORMAT(*COEF*9E12.4)
      45 CONTINUE
         CALL EXIT
         END
```

A 6. RTPAN with RETIORPA

Complete example of use and results (Tab. 5, sample S_1, Fig. 4). RTPAN is the program used for computation of results shown in Tab. 5, Fig. 3 and Fig. 4.

Requires:
- Subroutine RETIORPA
- Real function BOUND
- Subroutine GSIMDATA [1)]
- Subroutine GACF
- Subroutine CLEAN [2)]
- Subroutine RANSS

} All included in the listing of the program

Remark: [1)] Subroutine GSIMDATA simulates run of a given linear stochastic system for given input sequence

[2)] Subroutine CLEAN is used to set all elements of a desired vector array to zero.

RETIORPA is universal subroutine which performs one iteration of Panuška's algorithm on latest data pair and therefore can be used for real time identification. It also computes the covariance of residuals.

```
      PROGRAM RTPAN
C     THIS IS AN EXAMPLE HOW TO USE SUBROUTINE RETIORPA FOR IDENTIFICATION
C     OF A SIMULATED SYSTEM OF SECOND ORDER
C     AUTHOR,JAROSLAV VALIS  07/03-69  LTH LUND  SWEDEN
C
C     SUBROUTINE REQUIRED
C        RETIORPA
C        BOUND       (REAL, FUNCT.)
C        RANSS       (PSEUDORANDOM NORMAL NUMBERS (0,1) GENERATOR)
C        GSIMDATA    (SYSTEM SIMULATOR)
C        GACF        (COMPUTES AUTOCOVARIANCE)
C        CLEAN
C
      DIMENSION U(1000),Y(1000),E(1000),A(2),B(2),C(2),REE(11),U1(2),
     1Y1(2),E1(11)
      A(1)=-1.5 $ A(2)=.7 $ B(1)=1. $ B(2)=.5 $ C(1)=-1. $ C(2)=.2
      N=2
      NN=1000
      IT1=3
      IT2=5
C
C     GENERATE INPUT SEQUENCE
      DO 1 K=1,NN
    1 CALL RANSS(IT1,U(K))
C
C     SIMULATE RUN OF GIVEN SYSTEM
      CALL GSIMDATA(N,A,B,C,1.,IT2,NN,U,Y,E)
C
C     COMPUTE COVARIANCE OF ORIGINAL NOISE
      CALL GACF(E,NN,1,NN,REE,11,11)
      PRINT 100,(REE(I),I=1, 11)
  100 FORMAT(*COV.OF ORIG.NOISE*/12F9.5)
C
C     SET ZERO INITIAL CONDITIONS
      CALL CLEAN(A,N)
      CALL CLEAN(B,N)
      CALL CLEAN(C,N)
      CALL CLEAN(E1,11)
      CALL CLEAN(U1,N)
      CALL CLEAN(Y1,N)
      CALL CLEAN(REE,11)
C
C     START IDENTIFICATION
      ALFA=5
      IPASS=20
C
C     REPEAT (IPASS) TIMES
      DO 3 IC=1,IPASS
      KK=(IC-1)*NN
      DO 3 K=1,NN
      G=ALFA/(KK+K)
    3 CALL RETIORPA(U(K),Y(K),U1,Y1,N,A,B,C,5.,10.,G,E1,REE,11)
C
C     PRINT OUT THE RESULTS
      PRINT 101,IC
  101 FORMAT(/*PASS NO. *I3)
```

```
      PRINT 102,(A(I),I=1,N)
  102 FORMAT(/*AA  1.0000*10F8.4)
      PRINT 103,(B(I),I=1,N)
  103 FORMAT(*BB        *10F8.4)
      PRINT 104,(C(I),I=1,N)
  104 FORMAT(*CC  1.0000*10F8.4)
      DO 4 I=1,11
    4 REE(I)=REE(I)/NN
      PRINT 105,(REE(I),I=1,11)
  105 FORMAT(*COV.OF RES.*/12F9.5)
      CALL CLEAN(REE,11)
    2 CALL CLEAN(E1,11)
      CALL EXIT
      END
```

```
      SUBROUTINE RETIORPA(UK,YK,U,Y,NE,A,B,C,BC,BE,GAIN,E,REE,L)
C     REAL TIME IDENTIFICATION OF LINEAR SYSTEM USING STOCHASTIC
C     APPROXIMATIONS
C     REFERENCE,PANUSKA,V.,A STOCHASTIC APPROXIMATION METHOD FOR IDENTIFICATION
C     OF LINEAR SYSTEMS USING ADAPTIVE FILTERING, JACC 1968
C     AUTHOR,JAROSLAV VALIS, 07/03-69
C
C     UK- LATEST VALUE OF INPUT
C     YK- LATEST VALUE OF OUTPUT
C     U - VECTOR OF LENGTH (NE) CONTAINING LAST NE VALUES OF INPUT
C     Y - VECTOR OF LENGTH (NE) CONTAINING LAST NE VALUES OF OUTPUT
C     NE- ORDER OF THE ESTIMATED MODEL
C     A - VECTOR OF LENGTH (NE) CONTAINING CURRENT ESTIMATES OF A-COEF.
C     B - VECTOR OF LENGTH (NE) CONTAINING CURRENT ESTIMATES OF B-COEF.
C     C - VECTOR OF LENGTH (NE) CONTAINING CURRENT ESTIMATES OF C-COEF.
C     BC- BOUND VALUE FOR COEFFICIENTS A,B,C
C     BE- BOUND VALUE FOR ERRORS E
C     GAIN - GAIN CONSTANT
C     E - VECTOR OF LENGTH (L) CONTAINING LAST L RESIDUALS
C     REE-VECTOR OF LENGTH (L) CONTAINING CURRENT COVARIANCE OF RESIDUALS E
C     L - LENGTH OF VECTORS E,REE
C
C     SUBROUTINE REQUIRED
C        BOUND
C
      DIMENSION A(NE),B(NE),C(NE),REE(L),E(L),U(NE),Y(NE)
C
C     COMPUTE THE LATEST RESIDUAL
      EE=YK
      DO 1 I=1,NE
    1 EE = EE+A(I)*Y(I)-B(I)*U(I)-C(I)*E(I)
      EE = BOUND(EE,BE)
C
C     UPDATE ESTIMATES
      G = GAIN*EE
      DO 2 I=1,NE
      C(I)=BOUND(C(I)+G*E(I),BC)
      B(I)=BOUND(B(I)+G*U(I),BC)
    2 A(I)=BOUND(A(I)-G*Y(I),BC)
C
C     UPDATE I/O DATA
      DO 3 I=2,NE
      I1=NE-I+1
      U(I1+1)=U(I1)
    3 Y(I1+1)=Y(I1)
      U(1)=UK
      Y(1)=YK
      DO 4 I=2,L
      I1=L-I+1
    4 E(I1+1)=E(I1)
      E(1)=EE
C
C     UPDATE COVARIANCES
      DO 5 I=1,L
    5 REE(I)=REE(I)+EE*E(I)
      RETURN
```



```
      END
```

01/04-09

```
      REAL FUNCTION BOUND(X,B)
      IF(ABSF(X).LE.B) GO TO 1
      IF(X) 2,1,3
    1 BOUND=X $ RETURN
    2 BOUND=-B $ RETURN
    3 BOUND=B $ RETURN
      END
```

01/04-09

```
      SUBROUTINE CLEAN(A,N)
      DIMENSION A(N)
      DO 1 I=1,N
    1 A(I)=0.
      RETURN
      END
```

01/04-09

```
      SUBROUTINE GSIMDATA(N,A,B,C,LAMBDA,IT2,NN,U,Y,E)
      DIMENSION A(N),B(N),C(N),U(NN),Y(NN),E(NN)
      REAL LAMBDA
      DO 1 I=1,NN
      CALL RANSS(IT2,EE)
      E(I)=EE*LAMBDA
    1 CONTINUE
      DO 2 K=1,NN
      EE=E(K)
      DO 4 I=1,N
      J=K-I
      IF(J)2,2,4
    4 EE=EE+B(I)*U(J)-A(I)*Y(J)+C(I)*E(J)
    2 Y(K)=EE
C   END OF SIMULATION
      RETURN
      END
```

01/04-09

```
      SUBROUTINE GACF(X,NN,N1,N2,RXX,IT,N3)
      DIMENSION X(NN),RXX(N3)
      INTEGER TAU
      NT=N2-N1+2
      DO 1 TAU=1,IT
      XX=0.
      LB=N1+TAU-1
      DO 2 K=LB,N2
      J=K-TAU+1
    2 XX=XX+X(K)*X(J)
    1 RXX(TAU)=XX/(NT-TAU)
      RETURN
      END
```

```
RY DECK***
00,  ,1

ON STARTED AT   1507 -25
```

```
COV.OF ORIG.NOISE
  0.94431  0.02674  0.00639  0.01609 -0.00793  0.01812 -0.04917

PASS NO.   1

AA  1.0000 -1.3962  0.5021
BB          1.0275  0.5256
CC  1.0000 -0.6749  0.0731
COV.OF RES.
 28.58597 -1.46674 -1.74870  3.04233  8.31074 -1.78609  3.15400

PASS NO.   2

AA  1.0000 -1.4097  0.6813
BB          1.0236  0.5033
CC  1.0000 -0.8384  0.0630
COV.OF RES.
  1.31524 -0.04784  0.00659 -0.00461 -0.08108 -0.03867 -0.08550

PASS NO.   3

AA  1.0000 -1.5204  0.6997
BB          1.0117  0.4852
CC  1.0000 -0.8614  0.1000
COV.OF RES.
  1.18973 -0.01580  0.01996  0.02072 -0.02930  0.00551 -0.05179

PASS NO.   4

AA  1.0000 -1.5248  0.7083
BB          1.0040  0.4761
CC  1.0000 -0.8806  0.1168
COV.OF RES.
  1.12799 -0.01632  0.01337  0.02078 -0.02027  0.00999 -0.05050

PASS NO.   5

AA  1.0000 -1.5248  0.7129
BB          0.9990  0.4709
CC  1.0000 -0.8954  0.1258
COV.OF RES.
  1.09370 -0.01561  0.00938  0.01924 -0.01701  0.01168 -0.04993

PASS NO.   6

AA  1.0000 -1.5237  0.7157
BB          0.9956  0.4676
CC  1.0000 -0.9061  0.1339
COV.OF RES.
  1.07244 -0.01396  0.00774  0.01893 -0.01429  0.01367 -0.04797

PASS NO.   7

AA  1.0000 -1.5226  0.7177
BB          0.9931  0.4654
CC  1.0000 -0.9140  0.1369
COV.OF RES.
  1.05804 -0.01223  0.00691  0.01906 -0.01198  0.01557 -0.04629

PASS NO.   8
```

```
AA   1.0000 -1.5217   0.7190
BB            0.9911   0.4639
CC   1.0000 -0.9201   0.1405
COV.OF RES.
   1.04763 -0.01082   0.00634   0.01996 -0.01010   0.01720 -0.04474

PASS NO.    9

AA   1.0000 -1.5211   0.7199
BB            0.9895   0.4628
CC   1.0000 -0.9247   0.1441
COV.OF RES.
   1.03977 -0.00962   0.00587   0.01943 -0.00862   0.01853 -0.04349

PASS NO.   10

AA   1.0000 -1.5206   0.7205
BB            0.9882   0.4620
CC   1.0000 -0.9284   0.1469
COV.OF RES.
   1.03360 -0.00862   0.00547   0.01953 -0.00745   0.01959 -0.04247

PASS NO.   11

AA   1.0000 -1.5202   0.7210
BB            0.9871   0.4614
CC   1.0000 -0.9314   0.1493
COV.OF RES.
   1.02863 -0.00780   0.00511   0.01960 -0.00653   0.02043 -0.04184

PASS NO.   12

AA   1.0000 -1.5199   0.7213
BB            0.9861   0.4609
CC   1.0000 -0.9339   0.1513
COV.OF RES.
   1.02455 -0.00711   0.00479   0.01962 -0.00578   0.02111 -0.04097

PASS NO.   13

AA   1.0000 -1.5197   0.7215
BB            0.9853   0.4605
CC   1.0000 -0.9360   0.1531
COV.OF RES.
   1.02114 -0.00652   0.00450   0.01962 -0.00518   0.02169 -0.04043

PASS NO.   14

AA   1.0000 -1.5195   0.7217
BB            0.9846   0.4602
CC   1.0000 -0.9378   0.1546
COV.OF RES.
   1.01825 -0.00601   0.00424   0.01961 -0.00469   0.02219 -0.03996

PASS NO.   15

AA   1.0000 -1.5194   0.7218
BB            0.9840   0.4600
CC   1.0000 -0.9393   0.1560
COV.OF RES.
   1.01577 -0.00557   0.00401   0.01958 -0.00428   0.02246 -0.03962
```

PASS NO. 16

AA	1.0000	-1.5193	0.7219			
BB		0.9835	0.4598			
CC	1.0000	-0.9406	0.1573			
COV.OF RES.						
	1.01362	-0.00518	0.00380	0.01954	-0.00395	0.02276

PASS NO. 17

AA	1.0000	-1.5192	0.7219			
BB		0.9831	0.4596			
CC	1.0000	-0.9418	0.1584			
COV.OF RES.						
	1.01173	-0.00484	0.00361	0.01950	-0.00366	0.02300

PASS NO. 18

AA	1.0000	-1.5191	0.7220			
BB		0.9826	0.4595			
CC	1.0000	-0.9428	0.1593			
COV.OF RES.						
	1.01008	-0.00454	0.00344	0.01945	-0.00343	0.02321

PASS NO. 19

AA	1.0000	-1.5191	0.7220			
BB		0.9823	0.4594			
CC	1.0000	-0.9437	0.1602			
COV.OF RES.						
	1.00860	-0.00427	0.00328	0.01941	-0.00322	0.02338

PASS NO. 20

AA	1.0000	-1.5190	0.7220			
BB		0.9820	0.4593			
CC	1.0000	-0.9445	0.1611			
COV.OF RES.						
	1.00728	-0.00402	0.00313	0.01936	-0.00305	0.02352