International Journal of Scientific and Innovative Mathematical Research(IJSIMR)
*Volume 11, Issue 2, 2023, PP 17-31*
*ISSN No. (Online) 2347-3142*
*DOI: https://doi.org/10.20431/2347-3142.1102002*
*www.arcjournals.org*

# Nonlinear Robust Stochastic Inverse Model Based Optimum Control

**Endre Nagy***
*Member, SICE, JAPAN*

***Corresponding Author:** **Endre Nagy,** *Member, SICE, JAPAN*

**Abstract:** *: The paper presents a method for solution of the robust nonlinear stochastic control / tracking problem by way of error feedback through the inverse plant model. Model structure, parameter estimation, state estimation, inverse model computation and optimization problems are also treated. An example shows how to design robust nonlinear stochastic control with the nonlinear model equations. Another example compares the results achieved with one – stage optimization and two – stage optimization, respectively. The solutions are obtained in analytical forms.*

**Keywords:** *Robust adaptive control, Stochastic system identification, Recursive identification, Nonlinear adaptive control, Estimation and filtering, Optimal control theory, Inverse model based control.*

## 1. Introduction

Nonlinear problems may be solved with the nonlinear model equations or through linearization. State and parameter estimations in the paper are based on a nonlinear filtering algorithm, which applies a recently developed nonlinear prediction method. With this predictor both one – stage and multi – stage ahead predictions are achievable, giving way to nonlinear model predictive control (MPC) and nonlinear stochastic tracking. To solve the robust nonlinear stochastic control problem, first optimum control is computed with the plant model. If there are no constraints imposed on the control, the problem may be solved with a single step control through output fitting. However, if there are constraints to take into consideration, multi – step optimum control or receding horizon control gives the best result. For solution of the optimum stochastic control problem, the optimization method "optimized stochastic trajectory / output sequence tracking" (OSTT, [1]) may be used, which is based on two – stage optimizations. However, output of the plant differs from that of the model. To make the control robust, the error is fed back through the negative inverse of the model to the input of the plant, forcing the plant output nearer to the model output. The presentation is made for discrete time control; however, for continuous time control the continuous process may be discretized and the corresponding discrete time problem may be solved. The optimal continuous control can be approximated from the optimal discrete control signal sequence.

## 2. Control Schemfe For Robust Control

A plant in series connection with its inverse is equivalent to the identity system, i.e. inputs and outputs are the same, if no disturbances act on it. The influence of external disturbances and parameter changes may somewhat be compensated with feedback. Control with a good inverse model usually has large stability margin, assuring robust stability. Increasing the loop gain for error reduction, however, the tendency for oscillations usually grows. The suitable loop gain may be obtained e.g. through the "trial and error" method, or iterations. Further improvement can be achieved for plants with time varying parameters through real time estimation of parameters, and real time re-computation of the model, resulting in robust adaptive control with robust performance. However, in general an additional adaptive filter is required for plant modelling, since behaviour of plant might cause instability in the adaptation process [2]. If there were no disturbances, in case of error – free inverse model this control would be stable for all inputs, even without feedback, The parameters of the inverse model may be determined directly, too; however, even if stability is achieved, the parameter estimation would be biased in case of additive plant disturbance [3]. At the start rough estimation of the plant model is necessary to initialize the controller parameters and avoid instability. However, a disturbed plant may not follow the desired output well enough, resulting in limited applicability. In case of such a control the inverse model serves as disturbance canceler and plant dynamics controller at the same time. The control may be improved if these two functions are divided [3], as shown in Fig. 1:

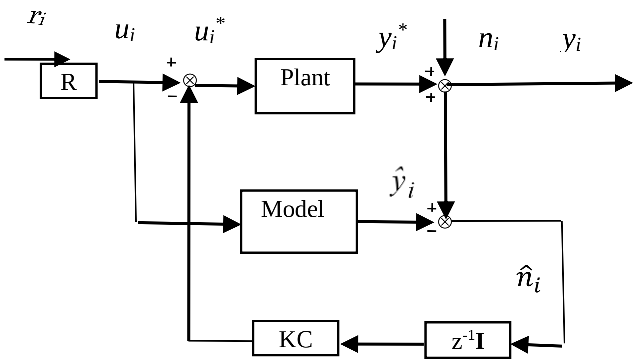


**Fig1.** *Control scheme for robust control*

On the Figure *R* – feedforward controller, *C* – inverse model compensator (disturbance controller), *K* - additional feedback gain, $r_i = r(i)$ - reference, $y_i^*$ - plant output, $n_i$ – plant disturbance, $y_i$ – disturbed plant outpt, $\hat{y}_i$ – estimated plant output, $u_i$ – feedforward controller output, $u_i^*$ - plant input, $\hat{n}_i$ – estimated disturbance. Optimum tracking is computed with the *R* feedforward controller; however, the plant output will differ from the model output. To force the plant output nearer to the model output, the error is fed back to the plant input through the negative inverse of the model, decreasing the plant input, if the plant output is greater than the model output and increasing it, if the model output is greater than the plant output. Simulations show that an additional feedback gain, determined through trials, may improve the control. Accuracy of tracking depends on accuracy of the model. The input for model parameter estimation is taken from the output of the feedforward controller to avoid bias. For the optimal model

$$\hat{y}(i) = E(y(i)/u^*(i)), \tag{1}$$

**i.e. the optimum model output is conditional expectation of plant output. It can be seen that**

$$\mathrm{E}(y_i / u_i^*) = \mathrm{E}(y_i^* / u_i^* + n_i / u_i^*) = \mathrm{E}(y_i^* / u_i^*)$$

$$+\,\mathrm{E}(n_i / u_i^*) = y_i^*(u_i^*) + \mathrm{E}(n_i / u_i^*). \tag{2}$$

(2) shows that the plant model is biased by the disturbance without disturbance cancellation, which may be carried out only with the plant input signal through feeding back the plant disturbance in such a way that it works against the disturbance on the plant output, as shown in Fig.1.

## 3. Computational Methods For The Solution

### 3.1 Inverse model computation

Consider a nonlinear system given with the equations

$$\boldsymbol{x}(\boldsymbol{i}+\mathbf{1}) = \boldsymbol{f}(\boldsymbol{x}(\boldsymbol{i}), \quad \boldsymbol{u}(\boldsymbol{i})) + \boldsymbol{w}(\boldsymbol{i}), \tag{3}$$

$$\boldsymbol{y}(\boldsymbol{i}) = \boldsymbol{g}(\boldsymbol{x}(\boldsymbol{i}),\boldsymbol{u}(\boldsymbol{i})) + \boldsymbol{n}(\boldsymbol{i}), \quad \boldsymbol{x} \in \boldsymbol{R}^{\boldsymbol{n}},\ \boldsymbol{u} \in \boldsymbol{R}^{\boldsymbol{p}},\ \boldsymbol{y} \in \boldsymbol{R}^{\boldsymbol{p}}. \tag{4}$$

In (3), (4) $\boldsymbol{x}(i)$ stands for the state vector, $\boldsymbol{u}(i)$ for the input vector, $\boldsymbol{y}(i)$ for the output vector, $\boldsymbol{w}(i)$ for white process noise and $\boldsymbol{n}(i)$ for white output disturbance. The right inverse of (3), (4) is such a discrete time nonlinear system that when the original system is connected in series with its right inverse, the outputs of the original system are equal to the inputs of the inverse system. On the other hand, when the inverse is connected in series to the original system, the output of the left inverse is the input of the original system. However, such an inversion rarely can be done. The forward time shift right inverse is of practical interest, which uses the γ- step forward time shift $\delta^{\gamma}$ operator on the original system output. The smallest γ, necessary for invertibility of (3), (4) has practical significance. For example, let the original system equation be

$$x(i+1)=ax^2(i)+bu(i), \tag{5}$$

$$y(i+1)=x(i+1). \tag{6}$$

In (5), (6) $a$, $b$ are parameters. Now we express $u(i)$ from (5) and (6):

$$u(i)=(y(i+1)-ax^2(i))/b. \tag{7}$$

Since (7) is a noncausal expression, $y(i+1)$ is forwarded with one step; the forwarded output of the original system is the $u_R(i)$ input of the inverse,

$$u_R(i)=y(i), \tag{8}$$

and the input of the original system is the $y_R(i)$ output of the inverse:

$$y_R(i)=(u_R(i)-ax^2(i))/b. \tag{9}$$

Invertibility of a given system is a basic question. The accurate process model, if known, is not necessarily invertible; however, for modeling purposes an invertible structure may be used. If the I/O map of a system is surjective, then inverse system exists. Right invertibility and surjectivity are equivalent [4]. However, for a unique inverse bijective mapping is needed.

### 3.2 State Estimation

For state estimation, "modified extended Kalman filter" (MOD – EKF) may be used. MOD – EKF uses the $\boldsymbol{M}$ – operator [5], [6] for predictions, and the EKF algorithm [7], [8] for updates. The $\boldsymbol{M}$ – operator has been developed on assumption of virtual sampling instants between real sampling time points and has the form

$$\boldsymbol{\Delta\hat{x}(i+1)} = \int_{\hat{x}^*(i-1)}^{\hat{x}^*(i)} \boldsymbol{f'}_{\hat{x}^*}(\hat{x}^*)d\hat{x}^* = \boldsymbol{M}(\boldsymbol{\Delta\hat{x}(i)},\ \boldsymbol{\Delta u(i)}). \tag{10}$$

In (10) $\boldsymbol{f}'_{\hat{x}^*}$ is derivative of the state vector and and $\hat{\boldsymbol{x}}^*$ is the modified (extended) state vector,

$$\hat{\boldsymbol{x}}^* = [\hat{\boldsymbol{x}}_1,\ \hat{\boldsymbol{x}}_2,\ \ldots\hat{\boldsymbol{x}}_n, \boldsymbol{u}_1, \boldsymbol{u}_2, \ldots \boldsymbol{u}_p]^{\mathrm{T}}. \tag{11}$$

The EKF algorithm computes the variance and gain matrices and updates with the linearized model, but the original nonlinear equations are used to state propagation. MOD – EKF replaces the standard state propagation of EKF with computation with the $\boldsymbol{M}$ – operator, resulting in better prediction and better overall estimation. The state prediction equation for discrete time EKF has the form [7]

$$\hat{\boldsymbol{x}}(i/i-1) = \boldsymbol{f}(\hat{\boldsymbol{x}}(i-1/i-1), \boldsymbol{u}(i-1)). \tag{12}$$

However, the $\boldsymbol{M}$ – operator uses infinite virtual sampling points to estimate the state evolution [5], [6] and the result is better prediction, especially for greater disturbances. The MOD – EKF algorithm uses the linearized model derived from (3), (4):

$$\Delta\boldsymbol{x}(i+1) = \boldsymbol{A}(i)\Delta\boldsymbol{x}(i) + \boldsymbol{B}(i)\Delta\boldsymbol{u}(i) + \Delta\boldsymbol{w}(i), \tag{13}$$

$$\Delta\boldsymbol{y}(i) = \boldsymbol{C}(i)\Delta\boldsymbol{x}(i) + \Delta\boldsymbol{n}(i), \tag{14}$$

$$\boldsymbol{\Delta x(i) = x(i) - x(i-1),\ \Delta u(i) = u(i) - u(i-1),}$$
$$\boldsymbol{\Delta y(i) = y(i) - y(i-1).}$$

In (13), (14) $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ are appropriate matrices. The algorithm may be summarized in the following manner:

**State propagation:**

$$\hat{\boldsymbol{x}}(i/i-1) = \hat{\boldsymbol{x}}(i-1/i-1) + \boldsymbol{M}(\Delta\hat{\boldsymbol{x}}(i), \Delta\boldsymbol{u}(i)) \tag{15}$$

**Covariance prediction:**

$$\boldsymbol{P}(i/i-1) = \boldsymbol{A}(\hat{\boldsymbol{x}}(i/i-1))\boldsymbol{P}(i-1/i-1)\boldsymbol{A}^{\mathrm{T}}(\hat{\boldsymbol{x}}(i/i-1)) + \boldsymbol{R}_w(i-1) \tag{16}$$

**Innovation:**

$$\boldsymbol{e}(i) = \boldsymbol{y}(i) - \hat{\boldsymbol{y}}(i/i-1) \tag{17}$$

**Innovation covariance:**

$$\boldsymbol{R}_e(i) = \boldsymbol{C}(\hat{\boldsymbol{x}}(\boldsymbol{i}/\boldsymbol{i}-1))\boldsymbol{P}(\boldsymbol{i}/\boldsymbol{i}-1)\boldsymbol{C}^{\mathrm{T}}(\hat{\boldsymbol{x}}(i/i-1)) + \boldsymbol{R}_n(i) \tag{18}$$

**Kalman gain:**

$$\boldsymbol{K}(i) = \boldsymbol{P}(i/\boldsymbol{i}-1)\boldsymbol{C}^{\mathrm{T}}(\hat{\boldsymbol{x}}(i/i-1))\boldsymbol{R}_e^{-1}(i) \tag{19}$$

**State correction:**

$$\hat{x}(i/i) = \hat{x}(i/i-1) + \boldsymbol{K}(i)\boldsymbol{e}(i) \quad \textbf{(20)}$$

**Covariance correction:**

$$\boldsymbol{P}(i/i) = [\boldsymbol{I} - \boldsymbol{K}(i)\boldsymbol{C}(\hat{x}(i/i-1))]\boldsymbol{P}(i/i-1) \quad \textbf{(21)}$$

**Initial conditions:**

$\hat{\boldsymbol{x}}(\mathbf{0/0})$; $\boldsymbol{P}(\mathbf{0/0})$

**Assumed disturbances:**

$\boldsymbol{N}(0,\boldsymbol{R}_w(i)),\ \boldsymbol{N}(0,\boldsymbol{R}_n(i))$

### 3.3 Modeling

#### 3.3.1 Model structures

Process identification includes selection of the model structure and parameter estimation. Process model sometimes can be obtained from principles of physics. However, if no physical insight is available on the process, processing of measured inputs and outputs (black box modeling) has to be applied. The principle of modeling is to define a general nonlinear structure, like polynomial, spline [9], wavelet, neural network, fuzzy model [10], [11] etc., which can be formed to any concrete model through estimation of parameters. Black box modeling is based on the nonlinear mapping [12], [10]

$$\hat{\boldsymbol{y}}(\boldsymbol{t},\boldsymbol{\theta}) = \boldsymbol{g}(\boldsymbol{\varphi}(\boldsymbol{t}),\boldsymbol{\theta}), \quad \textbf{(22)}$$

where $t$ - time, $\hat{y}(t) \in \boldsymbol{R}^p$ - estimated output vector, $\boldsymbol{\varphi}(t) \in \boldsymbol{R}^d$- regression vector, $\boldsymbol{\theta}$ - parameter vector, $\boldsymbol{g} \in \boldsymbol{R}^p$ - suitable nonlinear function. The regression vector can be chosen in different ways. A basic possibility is to let it contain past inputs with past measured and predicted outputs, similarly to the linear case. Once $\boldsymbol{g}$ is selected and the number of unknown parameters is defined, parameter estimation may be achieved through minimization of the PI

$$\sum_{t=1}^{N} \|\boldsymbol{y}(t)\text{-}\boldsymbol{g}(\boldsymbol{\varphi}(t),\boldsymbol{\theta})\| \ . \quad (23)$$

The norm in (23) may be defined in different ways. The nonlinear function may be given for a SISO plant in the form of function expansion:

$$g(\boldsymbol{\varphi}(\mathrm{t}),\boldsymbol{\Theta})=\sum \theta_{\mathrm{k}}\, g_{\mathrm{k}}(\boldsymbol{\varphi}(\mathrm{t})). \quad (24)$$

In (24) $g_k$ is a basis function. The (24) expression with different basis functions and regression vectors gives the general structure of black box modeling. State space black box models usually relate the state variables to each other, to the observations and excitations [13]. With the $\boldsymbol{M}$ – operator we can get estimates for the states and outputs in function of the model parameters, and the parameter estimation may be done similarly. The state space black box model may have the form for a SISO plant [14], [15], [16], [17], [18], [19]

$$\begin{bmatrix} x_1(t+T) \\ x_2(t+T) \\ \vdots \\ x_n(t+T) \end{bmatrix} = \begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_n(t) \end{bmatrix} + T \begin{bmatrix} x_2(t) \\ x_3(t) \\ \vdots \\ \boldsymbol{\varphi}^T(t)\boldsymbol{\theta} \end{bmatrix}, \quad (25)$$

where $T$ is the sampling period, and

$$\boldsymbol{y}(\mathrm{t}) = [1,0,\ldots 0]\boldsymbol{x}(t). \quad (26)$$

### 3.3.2 A possible parameter estimation method

Assume that a structure has been selected for modeling of a given nonlinear system, but the parameters have to be estimated. Suppose that the process is ergodic and $\boldsymbol{f}$, $\boldsymbol{g}$ in (3), (4) are analytic functions. Parameter estimation may be achieved through minimization on an $N$ – step horizon:

$$\min_{\boldsymbol{x},\boldsymbol{\Theta}} \sum_{j=0}^{N-1} \sum_{k=1}^{p} \{y_k(i-j) - \hat{y}_k(i-j)\}^2, \tag{27}$$

where $\hat{y}_k$ is an estimated output component. (27) is minimization of the estimated output variance. If

$$E\{y_k(i-j) - \hat{y}_k(i-j)\} = 0, \quad k = 1,\dots p, \quad j = 0,\dots N-1, \tag{28}$$

(27) is approximation of the output estimation error variance, too. Global minimum of (27) has to be searched not only in function of parameters, but the states at the bottom of horizon, too. The other states may be estimated with the $\boldsymbol{M}$ – operator.

Recursive nonlinear parameter estimation algorithms may be obtained if the procedure is repeated at the new sampling instant, and estimates of states and parameters on the preceding horizon are used at the bottom of new horizon. Define a vector

$$\boldsymbol{Q}(N) = [F(N), F(N-1), \dots F(N-n-m+1)]^T, \tag{29}$$

where $\hat{\boldsymbol{\Theta}} \in \boldsymbol{R}^m$ and

$$F(N,\hat{\boldsymbol{\Theta}}) = \sum_{j=0}^{N-1} \sum_{k=1}^{p} \{y_k(i-j) - g_k(\hat{\mathbf{x}}(i-j), \boldsymbol{u}(i-j), \hat{\boldsymbol{\Theta}})\}^2 = \sum_{j=0}^{N-1} \|\hat{\boldsymbol{\varepsilon}}_{i-j}\|^2. \tag{30}$$

In the definition of $\boldsymbol{Q}(N)$ and $F(N)$ some variations are possible, for example, the number of outputs taken into consideration for each element of $\boldsymbol{Q}(N)$ may differ in greater extent. Estimation on the new horizon may be improved with the

$$\alpha \boldsymbol{Q}(N) \approx -\boldsymbol{J}(\hat{\boldsymbol{x}}'(i-N)^{n+1} - \hat{\boldsymbol{x}}'(i-N)^{n}), \qquad 0 < \alpha \le 1 \tag{31}$$

relationship, which is based on the Newton iteration method. In (31)

$\hat{\boldsymbol{x}}'(i-N)^n$ - estimate of the modified state vector (state vector extended with the parameter vector after the $n$ – th iteration),

J - Jacobian matrix,

$$\boldsymbol{J} = \frac{\partial \boldsymbol{Q}}{\partial \hat{\boldsymbol{x}}'(i-N)}. \tag{32}$$

The derivatives in (32) may be evaluated numerically. From (31)

$$\hat{\boldsymbol{x}}'(i-N)^{n+1} \approx \hat{\boldsymbol{x}}'(i-N)^{n} - \alpha \boldsymbol{J}^{-1}\boldsymbol{Q}(N). \tag{33}$$

In practice $\boldsymbol{J}^{-1}$ is not computed but (31) is solved for $\hat{\boldsymbol{x}}'(i-\boldsymbol{N})^{n+1}$. The tuned filter gives the best performance. Tuning means not only good selection of filter parameters, but considerations for the form of PI and the used algorithm, too. If the parameters are time varying, influence of the old data may be decreased through use of forgetting factors [20], [21]:

$$F(N,\boldsymbol{\Theta}) = \sum_{j=0}^{N-1} \lambda^j \left\| \hat{\boldsymbol{\varepsilon}}_{i-j} \right\|^2, \qquad 0 < \lambda < 1. \tag{34}$$

A relatively simple computation may be used for state estimation through fixed point iteration [22]. For the ideal case

$$\begin{aligned} F(N) &= 0, \\ &\vdots \\ F(N-n-m+1) &= 0. \end{aligned} \tag{35}$$

The formula for iteration of the (35) system of equations is

$$\hat{\boldsymbol{x}}'(i-N)^{n+1} = \boldsymbol{Q}(N) + \alpha \hat{\boldsymbol{x}}'(i-N)^n, \tag{36}$$

where $0 < \alpha \leqq 1$ is the learning rate. If the (36) iteration is not convergent, selecting another initial point and / or smaller α, it may converge. With completion of the computations for several initial state values, the trap of finding the neighbourhood of a local minimum may be avoided. However, in case of recursive computations, this may be important only at the start of computations

### 3.3.3 Polynomial models

A convenient kind of models is the polynomial model. According the approximation theorem [23], if $f(x)$ is a continuous function in the interval $[a, b]$, then for any $\varepsilon > 0$ exists a polynomial $P_n(x)$ of degree at most $n$, $n=n(\varepsilon)$, such that

$$|f(x) - P_n(x)| < \varepsilon \tag{37}$$

for all $x$ in $[a, b]$. This means that arbitrarily close polynomial approximation of continuous functions is possible over a closed bounded interval. A particular interest is the problem of best approximation. The difference between $f((x)$ and $P_n(x)$ may be measured with the so – called maximum norm:

$$\left\| f(x) - P_n(x) \right\|_\infty = \max_{a \le x \le b} \left| f(x) - P_n(x) \right| = d(f(x), P_n(x)). \tag{38}$$

The polynom of best approximation minimizes (38). It can be proved [23] that for any continuous $f(x)$ function exists a $P_n(x)$ polynomial of degree at most $n$, which minimizes (38). The best approximation polynomial is unique [24]. However, computation of it is problematical in practice. A particular problem is that higher order polynomials tend to be oscillatory, causing numerical instability. For the discrete time case, the error between sampling time points may become too large. This problem can be solved with application of splines, which are numerically stable in general. Approximation techniques can be extended for the multivariable case [24].

### 3.3.4 Spline models

Splines (piecewise polynomials) don't have advantages comparing with polynomials if they are used for approximation of a well – behaved function (with the exception when the derivatives are also important, the measured data are sparse, or there are oscillatory problems (25)). The spline parameter computation in this paper is based on two – stage optimizations. At the start, either a decision is made on the degree of the spline, or the degree is finalized after some trial and error evaluations. The spline parameters on the first stage may either be selected or estimated from past data, and optimum control can be computed in function of the selected parameters and $\boldsymbol{u}(0)$ starting control signal. Assume that the system is at the $t_{i-1}$ sampling time point and at each sampling time point new spline parameters are computed, together with the new control signals. The unknowns can be computed from optimization on the $[t_{i-2}, t_i]$ interval. The necessary condition for two – stage minimum in function of the $a_1, \ldots a_m$ spline parameters and $u_1, \ldots u_p$ control signals is

$$\partial F_{i-1,i}/\partial a_1(i\text{-}2)=0,\ldots\ \partial F_{i-1,i}/\partial a_m(i\text{-}2)=0,\ \partial F_{i-1,i}/\partial u_1(i\text{-}2)=0,\ldots\partial F_{i-1,i}/\partial u_p(i\text{-}2)=0. \quad (39)$$

In (39) $F_{i-1,i}$ is the two – stage PI, computed with the $\boldsymbol{M}$ – operator for stochastic systems, the indices relate to the output horizon. If the known values of $a_1(i\text{-}2),\ldots a_m(i\text{-}2)$, $u_1(i\text{-}2),\ldots u_p(i\text{-}2)$ are substituted to (39), a system of equtions is obtained with the $a_1(i\text{-}1),\ldots a_m(i\text{-}1)$, $u_1(i\text{-}1),\ldots u_p(i\text{-}1)$ unknowns. From the solution the $\boldsymbol{u}(i\text{-}1)$ control vector is obtained, together with the new spline parameters. At $t_i$ the states are updated and the procedure is repeated on the $[t_{i-1}, t_{i+1}]$ interval. If there is no unique solution of the system of equations, some constraints can be

Another possibility is search for minimum of the two – stae PI through iterations.

There are some variations of the computed splines. For example, fitting of estimated outputs to specified references may be done not only at real sampling instants, but at intersample virtual sampling instants, too. The intrasample estimation can be done with MISLINPRED [6] or the $\boldsymbol{M}$ – operator ("$\boldsymbol{M}$" is from the first letter of MISLINPRED). This can be advantageous especially if the available data are sparse.

### 3.4 Optimization

It is well known that in general the minimum of a PI on a finite horizon is not equal with the sum of one – stage minima, but can be computed through a sequence of two – stage optimizations. The consecutive two – stage sections overlap each other on one stage, the second stage of the preceding section and the first stage of the new section is common [26] . Consequently, solution of the multi – step optimum control problem may be obtained through a sequence of two – stage optimizations. The practical working out has led to the optimization method "optimized trajectory tracking" (OTT) for deterministic systems and "optimized stochastic trajectory / output sequence tracking" [1] for stochastic systems. The finite horizon state space and I/O control problem can be solved with these optimization methods through a sequence of optimizations on a section of two stages. For optimum linear state space control, solution of the optimum infinite horizon control problem (computation of the steady state optimum feedback gain) can be derived through limit value calculation of the result of two – stage optimization [1]. Solution of the stochastic control problem is based on the principle "direct stochastic optimum tracking". The principle states that an estimated optimum stochastic trajectory (or output sequence) can be obtained with step – by – step optimum extension of a part of the estimated optimum trajectory (output sequence). With these principles and methods solution of the linear optimum stochastic state space and I/O, finite and infinite horizon control problems can be obtained [1], [27]. For the nonlinear case, assume a nonlinear process described with the state and output equations (3), (4), and with Gaussian zero mean disturbances and with time invariant parameters. The control problem may be solved through either linearization or with the original nonlinear equations. The linearized model for linearizable plants may be given in the form of (13), (14). This model behaves like a linear system with time varying parameters. However, if parameters of the nonlinear plant are constant and known, parameters of the linearized model can be computed and thus they can be considered as known. The PI for increments in a MIMO finite horizon nonlinear stochastic optimum tracking problem may have the form

$$\begin{aligned} F_{1,N} &= E\{\sum_{i=1}^{N}\{\Delta\boldsymbol{y}(\Delta\boldsymbol{x}(i))-\Delta\boldsymbol{r}(i)\}^T \boldsymbol{Q}\{\Delta\boldsymbol{y}(\Delta\boldsymbol{x}(\mathrm{i}))-\Delta\boldsymbol{r}(i)\} \\ &+\sum_{i=1}^{N}\Delta\boldsymbol{u}(i-1)^T \boldsymbol{R}\Delta\boldsymbol{u}(i-1)\}, \end{aligned} \tag{40}$$

where $\boldsymbol{Q}$ and $\boldsymbol{R}$ are appropriate weighting matrices. In (40) indices of $F$ refer to the output horizon. $E\{\boldsymbol{x}(0)\}$ is supposed to be known. The optimization problem may be solved similarly to the linear case, taking into consideration the computed operating point dependent values of parameters at each sampling time instant. It can be shown that the feedback solution [1], [27] can be given in the form

$$\Delta\boldsymbol{u}(i) = -\boldsymbol{K}(i)\Delta\hat{\boldsymbol{x}}(i) + \boldsymbol{v}\{\Delta\boldsymbol{r}(i+1), \Delta\boldsymbol{r}(i+2)\}. \tag{41}$$

In (41) $\boldsymbol{K}(i) = \boldsymbol{K}(\Delta\hat{\boldsymbol{x}}(i),\Delta\hat{\boldsymbol{x}}(i+2))$ is the time varying optimum feedback gain and $\boldsymbol{v}\{\Delta\boldsymbol{r}(i+1), \Delta\boldsymbol{r}(i+2)\}$ is the command input. (41) follows from minimization of the two – stage PI and arrangement of the terms, which are all linear, obtained through elimination of $\Delta\hat{\boldsymbol{x}}(i+1)$ and $\Delta\boldsymbol{u}(i+1)$ [1]. The control signal increment and the $\boldsymbol{K}(i)$ optimum local feedback gain are the same as in the corresponding deterministic system, if the disturbances are white with zero mean and appear in separated terms in the model, and the estimated actual and preceding operating point values are the same in both systems. The solution can be interpreted as the best choice with respect to a two – stage PI, if the system is in a certain state and has a certain past. To get the (41) solution, future states and outputs are estimated with the $\boldsymbol{M}$ – operator predictor. Following each update of states, update of state increments and parameters is also necessary.

For infinite horizon control the steady state optimum feedback gain can be obtained through limit value calculation [1], [27] as

$$\begin{aligned} \boldsymbol{K}_\infty(i) &= \lim_{i\to\infty} \boldsymbol{K}\{\Delta\hat{\boldsymbol{x}}(i),\Delta\hat{\boldsymbol{x}}(i+2)\} \\ &= \lim_{i\to\infty} \hat{\boldsymbol{K}}\{\frac{\Delta\hat{\boldsymbol{x}}_j(i+2)}{\Delta\hat{\boldsymbol{x}}_j(i)}\}, j=1...n. \end{aligned} \tag{42}$$

$\boldsymbol{K}(i)$ and $\boldsymbol{K}_\infty(i)$are operating point and reference dependent local gains. With (42), the optimum infinite horizon control law is

$$\Delta \boldsymbol{u}(i) = -\boldsymbol{K}_{\infty}(i)\Delta \hat{x}(i) + \boldsymbol{v}\{\Delta \boldsymbol{r}(i+1), \Delta \boldsymbol{r}(i+2)\}. \tag{43}$$

Before optimizing on the new two – stage section, the states are updated at the beginning of section with MOD – EKF. If updated values are used at the end of first stage, the optimization gives the feedforward solution [27]. The signals of the original system have to be reconstructed from the solution for increments. Through application of the two – stage optimization concept, optimum control of the nonlinear process respect to a suitable PI is possible not only around an operating point, but on a horizon, too.

If the original nonlinear equations are used for optimization and there are no constraints to take into consideration, the necessary control maybe computed from evaluation of

$$E\{\boldsymbol{g}(\boldsymbol{x}(i+1))/\boldsymbol{Y}(i)\} = \boldsymbol{r}(i+1). \ \mathrm{i} = 0,1,...N\text{-}1, \tag{44}$$

where $\boldsymbol{Y}(i)$ is the measurement vector. (44) may be solved through computation with the $\boldsymbol{M}$ – operator. However, if constraints are imposed on the control, the (44) fitting may not give the best result, the optimum solution may be computed through a sequence of two – stage optimizations. Assume a finite horizon control problem with the PI to be minimized

$$F_{1,N} = \sum_{i=0}^{N-1} f'(\hat{x}(i), u(i)) \tag{45}$$

and the constraints to be taken into consideration

$$\begin{aligned} &d_k(\hat{\boldsymbol{x}}(i), \boldsymbol{u}(i)) \geq 0, \quad k = 1,...K, \\ &l_j(\hat{\boldsymbol{x}}(i), \boldsymbol{u}(i)) = 0, \quad j = 1,...J, \quad i = 0,1,...N-1. \end{aligned} \tag{46}$$

$d_k$ and $l_j$ are, as a rule, nonlinear functions. The expectation of the initial state

$$E\{\boldsymbol{x}(0)\} = \hat{\boldsymbol{x}}_0$$

is assumed to be known. To the solution the PI can be modified to take into account the constraints. The problem can be solved in principle through a series of minimizations of the two – stage PI

$$F_{i+1,i+2} = \sum_{l=i}^{i+1} f'(\hat{x}(l), u(l)), \qquad l = 0, \ldots N-2. \tag{47}$$

At the beginning of computations estimation is made for the control signal on the first stage, e.g. with (44). However, if the references are known in advance on a finite horizon, the optimum starting control signal can be obtained from solution of the corresponding MPC problem [1]. On the two – stage sections for optimization, the state at the end of the first stage is estimated with MOD – EKF, and at the end of the second stage with the $\boldsymbol{M}$ – operator. The algorithm uses updated values everywhere when it is possible. $\boldsymbol{u}(1)$ and $\hat{\boldsymbol{y}}(2)$ can be obtained in knowledge of $\hat{x}(0), \hat{\boldsymbol{y}}(1)$ and $\boldsymbol{u}(0)$. In the next step optimization is made on the section $[\hat{\boldsymbol{y}}(1), \hat{\boldsymbol{y}}(3)]$ from which $\boldsymbol{u}(2)$ and $\hat{\boldsymbol{y}}(3)$ can be computed. Following this procedure, $\hat{\boldsymbol{y}}(N) \approx \boldsymbol{r}(N)$ is reached through $N$ - 1 two - section optimizations. The obtained output sequence, if the optimization problem has only one unique solution, is optimum in function of $\boldsymbol{u}(0)$, since on any $[\hat{\boldsymbol{y}}(i\text{-}1), \hat{\boldsymbol{y}}(i+1)]$section $\hat{\boldsymbol{y}}(i+1)$ is optimum for given $\hat{\boldsymbol{y}}(i-1)$, $\boldsymbol{u}(i\text{-}1)$, or with other words, $\hat{\boldsymbol{y}}(i)$ is optimum for given $[\hat{\boldsymbol{y}}(i\text{-}1), \hat{\boldsymbol{y}}(i+1)]$ with respect to the PI, and this is peculiarity of the optimum output sequence.

## 4. Examples

Example 1:

Consider a plant with the equations

$$x(i+1) = a\,exp(-x(i))u(i) + bu(i) + w(i), \tag{48}$$

$$y(i+1) = x(i+1) + n(i+1). \tag{49}$$

In (48) $a$, $b$ are parameters. It is assumed that the disturbances are of normal distribution and the expectations, parameter values and variances are

$$\begin{aligned} &E\{w(i)\} = 0;\ E\{n(i)\} = 0;\ a = 0.2; \\ &b = 0.5;\ \sigma^2_{\boldsymbol{w}(\boldsymbol{i})} = 0.0168;\ \sigma^2_{\boldsymbol{n}(\boldsymbol{i})} = 0.0675. \end{aligned} \tag{50}$$

Let the model structure to the design be

$$x(i+1) = a1x^2(i) + b1u(i) + w(i), \tag{51}$$

$$y(i+1) = x(i+1) + n(i+1). \tag{52}$$

In (51) $a1$, $b1$ are parameters. The motivation for selection of the (51), (52) model may be the easy manageability, and the fact that arbitrary accuracy can be achieved with polynomials, although high degree polynomials may become numerically unstable [22]. To the solution first the model parameters are identified; this has been done as shown in section 3.3.2. The control scheme is as of Fig.1. State estimation may be achieved with the MOD – EKF algorithm. Evaluating the $\boldsymbol{M}$ – operator with (10),

$$\begin{aligned} \boldsymbol{M}(\Delta\hat{\boldsymbol{x}}(i),\Delta\boldsymbol{u}(i)) = a1(\hat{\boldsymbol{x}}^2(i/i) - \hat{\boldsymbol{x}}^2(i-1/i-1)) \\ + b1(\boldsymbol{u}(i) - \boldsymbol{u}(i-1)). \end{aligned} \tag{53}$$

The state estimation algorithm with MOD – EKF can be given with the following equations:

$$x(i+1/i) = \hat{x}(i/i) + a1(\hat{x}^2(i/i) - \hat{x}^2(i-1/i-1) + b1u(i) - u(i-1)) \tag{54}$$

$$P(i+1/i) = (2a1\hat{x}(i+1/i))^2 P(i/i) + R_w(i) \tag{55}$$

$$e(i+1) = y(i+1) - \hat{y}(i+1/i) \tag{56}$$

$$R_e(i+1/i) = P(i+1/i) + R_n(i) \tag{57}$$

$$K(i+1) = P(i+1/i)/R_e(i+1) \tag{58}$$

$$\hat{x}(i+1/i+1) = \hat{x}(i+1/i) + K(i+1)e(i+1) \tag{59}$$

$$P(i+1/i+1) = (1 - K(i+1))P(i+1/i). \tag{60}$$

It can be seen from (53), (54), (51), (15) and (12) that prediction with the $\boldsymbol{M}$ – operator for the deterministic case gives back the EKF predictor. As for the linear case, it has been proved in [5] that prediction with the $\boldsymbol{M}$ – operator leads to the Kalman filter predictor.

Since the optimum solution is two – stage optimization based one, for optimization the PI

$$F_{i+1,i+2} = \sum_{j=i+1}^{i+2} [\{y(j) - r(j)\}^2 + \lambda u^2(j-1)] \tag{61}$$

is used. In (61) $r(j)$ is the reference and λ is the control weight. First optimum tracking is computed with the model, and the difference between the plant and the model output is fed back through the negative inverse of the model to the input of the plant, as in Fig. 1. From

$$\frac{\partial F_{i+1,i+2}}{\partial u(i)} = 0 \tag{62}$$

the next control signal can be computed in knowledge of $u(i)$. The initial control signal has been estimated from one – stage optimization. Completing the computations, we get

$$\begin{aligned} u(i+1) = &-(2(\hat{x}(i/i) + a1(\hat{x}(i/i)^2 - \hat{x}(i-/i-1)^2) \\ &+b1(u(i)\text{-}u(i-1))\text{-}r(i+1))b1 + 2\lambda u(i) \\ &+2(\text{DER}(i+1) + 2a1\hat{x}(i+1/i+1)\text{DER}(i+1) \\ &\text{-}b1)(\hat{x}(i+1/i+1) + a1(\hat{x}(i+1/i+1)^2 - \hat{x}(i/i)^2) \\ &\text{-}b1u(i)\text{-}r(i+2)))/(2(\text{DER}(i+1) \\ &+2a1\hat{x}(i+1/i+1)\text{DER}(i+1)\text{-}b1)b1), \\ &\text{DER}(i+1) = \frac{\partial \hat{x}(i+1/i+1)}{\partial u(i-1)}. \end{aligned} \tag{63}$$

DER($i$+1) was numerically approximated to the simulation. The correction on the input of the plant can be computed as

$$ucorr(i+1) = -K(y(\text{i}+1) - \hat{x}(i+1/i+1) - a1\hat{x}(i+1/i+1)^2)/b1. \tag{64}$$

To the simulation (63) and (64) were used. Various investigations have been done through simulations. For the process parameters of (51) the parameter estimator finds minimum at $a1$= 0.463 and $b1$= 0.563. If λ=0.5, the control is unstable without feedback. The control is unstable for $a1$=0.3 and $b1$=0.6. too, without feedback, but it becomes stable with feedback in both casas, achiving robust stability. However, deterioration of the process output is 45%, based on comparison of sum of error squares on a finite horizon. Accuracy of tracking may be improved for processes of time varying parameters if the model parameters are identified in real time, resulting in robust performance. Optimum value of the additional feedback gain is $K \approx 1.5$.

Example 2:

The control problem, parameters and other conditions are the same as in Example 1; however, now one – stage control is computed. The PI for one – stage control by application of the ***M*** – operator is

$$F_{i+1} = \{\hat{x}(i/i) + a1(\hat{x}^2(i/i) - \hat{x}^2(i-1/i-1)) \\ +b1(u(i) - u(i-1)) - r(i+1)\}^2 + \lambda u^2(i). \tag{65}$$

Necessary condition for minimum is

$$\frac{\partial F_{i+1}}{\partial u(i)} = 2\{\hat{x}(i/i) + a1(\hat{x}^2(i/i) - \hat{x}^2(i-1/i-1)) \\ + b1(u(i) - u(i-1)) - r(i+1)\} b1 + 2\lambda u(i) = 0. \tag{66}$$

From (66) the one – stage control law is

$$u(i) = -\{\hat{x}(i/i) + a1(\hat{x}^2(i/i) - \hat{x}^2(i-1/i-1)) \\ - b1(u(i-1) - r(i+1)\} b1/(\lambda + b_1^2). \tag{67}$$

Fig.2. shows the result of simulation for reference tracking with the output:

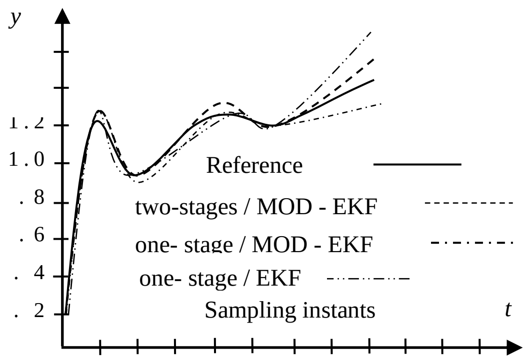


Fig.2. Reference tracking with one – stage and two – stage control.

The simulation shows that tracking accuracy for λ=0.5 and MOD-EKF estimation deteriorates with 22.6% on a 8 step horizon if one – stage control is used instead of two – stage one. However, if one – stage control with EKF estimation is used, the deterioration is 855%. All simulations were made with the same initial and other conditions. If disturbances are considered zero, λ=0 and the filtering algorithm is replaced with the expression for prediction, reference tracking is carried out with very small computational error, which becomes zero after a few steps, even with two – stage control.

## 5. Conclusions

The paper shows that optimum solution of the discrete time robust nonlinear stochastic control / tracking problem may be achieved through a sequence of two – stage stochastic optimizations, and that feedback of error through the inverse of the model may make the control robust. The solution may be extended for continuous time control through discretization of the plant and approximating the discrete time control signal with a continuous one; obtaining a suboptimal solution. It is also shown that the solution may be obtained either through linearization or with the nonlinear model, applying recently developed prediction and filtering methods. An example illustrates that the presented methods are applicable and analytical solutions may be achieved. The solution optimum for the model with the given PI and estimation methods, and is working effectively for the plant. Another example clearly demonstrates that the two – stage optimization based solution gives better result than the one – stage optimization based one, and estimation with the recently developed MOD – EKF algorithm assures higher tracking accuracy than what can be achieved with EKF.

## REFERENCES

[1] Nagy, E., Control Design: a New Approach, Proceedigs of International Conference on Computer, Communication and Control Technologies and The 9th International Conference on Information Systems, Analysis and Synthesis, **III**, Pp. 236-241, (2003).

[2] Klippel, W.J., Adaptive inverse control of weakly nonlinear systems, Proc. IEEE – ICASSP, Pp. 355-358, (1997).

[3] Nagy, E., Modeling and Control with Neural Networks through a New Learning Method, ICNPAA World Congress: 10th International Conference on Mathematical Problems in Engineering, Aerospace and Sciences, AIP Conf. Proc. 1637, 697, (2014).

[4] Ü. Kotta, *Inversion Method in the Discrete – time Nonlinear Control Systems Synthesis Problem*, Springer - Verlag London Limited, 1995.

[5] Nagy, E., Minimum Variance Nonlinear Estimation: State Evolution Approach, Science Direct, IFAC PapersOnLine 50-1 3786-3792, (2017a).

[6] Nagy, E., Model Predictive Control: a New Approach, AIP Conf. Proc. **1798**, 020105, doi: 10.1063/1.4972697, (2017b).

[7] J.V. Candy, *Signal processing*, McGraw – Hill Book Company, New York, 1987.

[8] R.F. Stengel, *Optimal control and estimation,* Dover Publications, Inc., New York, 1994.

[9] L.L. Shumaker, *Spline functions: Basic Theory,* Wiley, Chichester, 1981.

[10] L. Ljung, *System Identification Theory for the User,* Prentice Hall, Inc., Upper Saddle River, New Jersey, 1996.

[11] Ljung, L., Non-linear black box models in system identification, Proc. IFAC Symposium on Advanced Control of Chemical Processes, ADCHEM'97, Banff, **30**, Pp. 1-12, (1997).

[12] Sjöberg, J., Zhang, Q ., Ljung, L., Benveniste, A., Deylon, B., Glorennec, P-Y., Hjalmarsson, H. and Juditsky, A., Nonlinear Black – Box Modeling in System Identification: a Unified Overview, Automatica, **31**, 1691 – 1724, (2005).

[13] J.P. Norton, *An Introduction to Identification*, Academic Press, London and New York, 1986.

[14] Wigren, T., Recursive identification based on nonlinear state space models applied to drum – boiler dynamics with nonlinear output equations, Proc. of ACC 2005, Portland, Pp. 5066-5072, (2005a).

[15] Wigren, T., Scaling of the Sampling Period in Nonlinear System Identification, American Control Conference, Portland, Pp. 5058 – 5065, (2005b).

[16] Wigren, T., Recursive prediction error identification and scaling of nonlinear state space models using a restricted black box parametrization, Automatica, 42(1), 168-206 (2006).

[17] Brus, L. Noninear Identification of an Anaerobic Digestion Process, IEEE International Conference on

Control, Applications, Toronto, Pp. 137 – 142, (2005a)

[18] Brus, L., Nonlinear Identification of a Solar Heating System, IEEE International Conference on Control Applications, Toronto, Pp. 1491 – 1497, (2005b).

[19] Brus, L and D. Zambrano, Black box identification of solar collector dynamics with variant time delay, Control Engineering Practice 18(10), 1133-1146 (2010).

[20] K.J. Åström, and B. Wittenmark , *Computer Controlled Systems: Theory and Design*, Prentice – Hall, Inc., Englewood Cliffs, 1984.

[21] K.J. Åström and B. Wittenmark, *Adaptive Control*, Addison - Wesley Publishing Company, Inc., New York, 1995.

[22] J.R. Rice, *Numerical Methods, software, and analysis*, McGraw – Hill, Singapore, 1983.

[23] J.G. Burkill, *Lectures on Approximation by Polynomials*, Tata Institute of Fundamental Research, Bombay, 1959.

[24] E. Isaacson,.and H.B. Keller, *Analysis of Numerical Methods*, Dover Publications, Inc., New York,1966.

[25] R.W. Hamming, *Numerical Methods for Scientists and Engneers*, McGraw-Hill, New York, 1973.

[26] Nagy, E., Control of Distributed Parameter Systems: a Practical Approach, Sixth International Conference on Mathematical Problems in Engineering & Aerospace Sciences, Cambridge Scientific Publishers, Sivasundaram, S. (ed.), Pp. 527-539, (2007).

[27] Nagy, E., Control of Systems with Time Varying Parameters and Time Varying Expectation of Stochastic Disturbances, Seventh International Conference on Mathematical Problems in Engineering, Aerospace and Sciences, Cambridge Scientific Publishers, Sivasundaram, S. (ed.), Pp. 298-311, (2008).

**Citation:** *Endre Nagy, "Nonlinear Robust Stochastic Inverse Model Based Optimum Control", International Journal of Scientific and Innovative Mathematical Research (IJSIMR), vol. 11, no. 1, pp. 17-31, 2023. Available : DOI: https://doi.org/ 10.20431/2347-3142.1102002*