

TO LINEARIZE NONLINEAR SYSTEM MODEL USING SENSITIVITY FUNCTIONS

Dorina POPESCU

Department of Automation and Industrial Informatics, "Politehnica" University of Timișoara, Faculty of Automation and Computers, Bv.V. Pârvan nr.2, 1900 Timișoara, Romania

Phone: +40 0256 403231 , e-mail: dpopescu@aut.utt.ro

Abstract: *This paper contains a new manner to simplifying nonlinear system models. The problem is to find a linear model so that the square error between the output of the nonlinear model and the output of the linear model is minimum for all considered time. To solve this problem, in the present paper, is suggested to use the potential capabilities of the parameter deviations and the sensitivity functions to compute the square error. Using sensitivity equations (models) with respect to the linearize parameters, it is possible, generally speaking, to obtain the sensitivity functions with respect to linearize parameters. Then, using a space-parameter diagram and an grapho-analytic method the parameters working point are funded. A particular application of the proposed solution is analyzed in later sections. It is about a mechanical well-known system. It is also suggested how this method can be combined with certain aspects of control design.*

Keywords: *to linearize nonlinear system model, sensitivity functions with respect to linearize parameters, linear model of sensitivity.*

1. INTRODUCTION

Nonlinear systems are more likely to be used as models for real-life systems. The linear models of nonlinear plants, around the stationary point, have a wide variety of applications (for example to analyze some properties of the stationary point or to design a control system for a nonlinear plant).

The main goals of the present paper are:

- (i) to use sensitivities to develop a theoretical method to find an optimal stationary point (in a certain least-squares sense). This

stationary point will be used to linearize a nonlinear system model.

- (ii) to apply this theoretical procedure to a class of affine models (a single-input plant with a cvasicanonical form in state-space model).

To achieve the goals, section 2 describes mathematical details of two blocks of the structure presented in fig. 1.

Section 3 develops a representation of the output block such that a quadratic index of performance is minimized.

Section 4 shows how to obtain all the sensitivity functions with respect to parameters linearization.

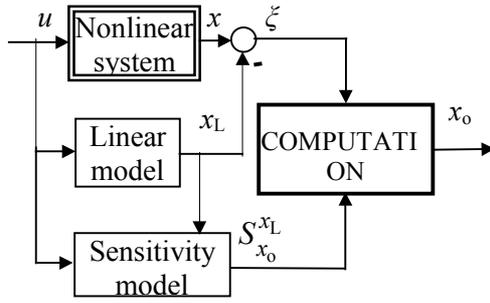


Fig.1. The block structure to compute the values of the linearization point parameters.

The manner to apply the theory above by a concrete case study will be explained in section 5.

In section 6, conclusions and future directions are summarized.

2. NONLINEAR AND TANGENT-LINEAR MODELS

This section presents the nonlinear system model, the tangent-linear model linearized around the stationary point. Also, a performance index will be defined.

2.1. Nonlinear system model

Consider a system described by the nonlinear state-space equation

$$\dot{x}(t) = f(x(t), u(t)), \quad x(t_0) = x_0 \quad (1)$$

where $x(t)$ is an n -vector representing the state of the system at time t , $u(t)$ is an m -vector representing the control input and the n -vector function f depends upon u and x . Suppose that $\Lambda_0(u_0, x_0)$ is the stationary point (at this point the system will not move). It is obvious that

$$f(x_0, u_0) = 0, \quad x_0 = x(t_0), \quad u_0 = u(t_0). \quad (2)$$

A particular case of a stationary point, ($m=n=1$ and output is x) is illustrated in fig.2. The input $u(t)$ is constrained to have values in a certain set $u(t) \in [u_{\min}, u_{\max}]$ for all $t \in [t_0, t_f]$. The state of the system is constrained in the same manner $x(t) \in [x_{\min}, x_{\max}]$, see [1].

Next, the main idea is to approximate a nonlinear system by a tangent-linear one in the neighborhood of Λ_0 point.

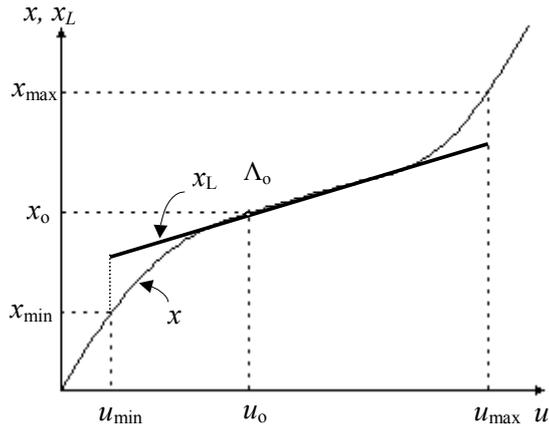


Fig.2. The stationary point $\Lambda_0(u_0, x_0)$ in a single input-output representation.

2.2. Tangent-linear model

A linearized model around the stationary point $\Lambda_0(u_0, x_0)$ is obtained applying the tangent linearization technique to expression (1). Then we can write the first form of the linearized model

$$\dot{x}_L(t; x_0) = \left. \frac{\partial f}{\partial x} \right|_{\Lambda_0} (x_L(t; x_0) - x_0) + \left. \frac{\partial f}{\partial u} \right|_{\Lambda_0} (u(t) - u_0) \quad (3)$$

$$x_L(t_0; x_0) = x_0, \quad u(t_0) = u_0. \quad (4)$$

This model (3) can be represented in the form

$$\dot{x}_L(t; x_0) = A(x_0) \cdot (x_L(t; x_0) - x_0) + B(x_0) \cdot (u(t) - u_0) \quad (5)$$

where

$$A(x_0) = \left. \frac{\partial f}{\partial x} \right|_{\Lambda_0}; \quad B(x_0) = \left. \frac{\partial f}{\partial u} \right|_{\Lambda_0}. \quad (6)$$

It is relevant to remark that $(x_0)^T$ can be regarded as a parameter-vector for the linear model (5). It is required to find the best nominal value for x_0 . This problem can be approached in many different ways. In the case under consideration the desired value of x_0 can be determined in such a manner that the performance index to be minimized.

3. THE COMPUTATION ALGORITHM

The computation algorithm of x_0 from fig.1 is based on a performance index minimization method.

3.1. The performance index of the quadratic errors

It is convenient to call the error between the nonlinear and the linear models, the difference

$$\zeta(t; x_0) = x_L(t; x_0) - x(t). \quad (7)$$

Both model (1) and model (5) have the same initial conditions. Then

$$\xi(t_0; x_0) = x_L(t_0; x_0) - x(t_0) = 0. \quad (8)$$

Consider now the performance index of the quadratic errors

$$J = \int_{t_0}^{t_f} \|\xi(\tau; x_0)\|^2 d\tau = \int_{t_0}^{t_f} (\xi(\tau; x_0))^T \cdot \xi(\tau; x_0) d\tau. \quad (9)$$

Note that (9) is in the form of a linear functional and [2-3] can therefore consider it as the square of the norm in the Hilbert space $L_2(t_0, t_f)$ or the inner product (dot product) between two identical vectors, $\langle \xi, \xi \rangle$.

We shall seek for a connection between the performance index and the sensitivity functions.

3.2. The minimum necessary condition

The ideal goal is to minimize the performance index (9) with respect to the parameter vector x_0 .

The minimum condition for (9) is

$$\frac{\partial J}{\partial x_0} = 0. \quad (10)$$

It follows from (9) and (6) that

$$J = \int_{t_0}^{t_f} (x_L(t; x_0) - x(t))^T (x_L(t; x_0) - x(t)) dt. \quad (11)$$

In this case equation (10) becomes

$$\underbrace{\int_{t_0}^{t_f} \left[\frac{\partial x_L(\tau; x_0)}{\partial x_0} \right]^T x(\tau) d\tau}_{I_1(x_0)} = \underbrace{\int_{t_0}^{t_f} \left[\frac{\partial x_L(\tau; x_0)}{\partial x_0} \right]^T x_L(\tau; x_0) d\tau}_{I_2(x_0)}. \quad (12)$$

Because x_0 is an n -vector

$$(x_0)^T = [x_{0,1}, x_{0,2}, \dots, x_{0,n-1}, x_{0,n}]^T \quad (13)$$

It is difficult to solve the equation (11) for x_0 . This means that a more careful approach is needed. These operations are discussed in more detail in the particular case illustrated in fig.2.

We denote by x_0^* the point at which the functional $J(x_0)$ has a minimum.

In order to achieve this, it is necessary to evaluate the definite integrals $I_1(x_0)$, $I_2(x_0)$ of the both sides of equation (12), at each given point x_0 from an infinite interval $x_0 \in [x_{\min}, \dots, x_{\max}]$. To make the calculation easier, it is convenient to replace this interval by an ordinate fixed set of k possible stationary (nominal) points, closely spaced

$$x_0 \in \{x_{\min} = x_0^1, x_0^2, x_0^3, \dots, x_0^k = x_{\max}\} \quad (14)$$

and then to use numerical computation. After all integrals have been evaluated we are now in a position to determine the solution of the equation (12). If it exists, this is the value x_0^* for which the two functions $I_1(x_0)$, $I_2(x_0)$ coincide (as in fig.5 from the study case).

3.3. Summary of sequence of calculations

The sequence of computations for determining x_0^* are summarized in the following steps:

1. Describe the nonlinear system model (1) and the stationary point (literal).
2. Compute (6) to find the linearize system model (5).
3. Settle the possible k stationary points (14). Set $j = 1$ and fix ε . The value of ε is a default tolerance.
4. Determine the sensitivity models with respect to all the linearization parameters (see section 4).
5. Simulate nonlinear model, linear model and sensitivity models for $x_0 = x_0^j$.
6. Compute $I_1(x_0^j)$ and $I_2(x_0^j)$, then $j = j + 1$.
7. If $j \leq k$, then repeat 5 and 6, else continue with 8.
8. For $j = 1, k$

$$d = | I_1(x_0^j) - I_2(x_0^j) |;$$
 if $d \leq \varepsilon$ then the desired value of x_0 is $x_0^* = x_0^j$;
 else, we shall say that equation (12) has no solution.

4. SENSITIVITY FUNCTIONS AND MODELS WITH RESPECT TO LINEARIZATION PARAMETERS

This section presents the simplest manner to obtain the sensitivity functions needed to solve the equation (12).

4.1. The sensitivity functions matrix with respect to linearization parameters

Generally, it is relevant to remark that to solve x_0 from (12) it is necessary to compute

$$\left. \frac{\partial x_L(t, x_0)}{\partial x_0} \right|_{x_0=x_0^j} = S_{x_0}^{x_L}(t, x_0^j). \quad (15)$$

for all $j, j = \overline{1, k}$. In fact, (15) corresponds to the $n \times n$ matrix of the state-space sensitivity functions with respect to parameters of x_0 , with nominal values x_0^j . We shall call this matrix: *the sensitivity functions matrix with respect to linearization parameters* (at the nominal point x_0^j). The column i of (15) represents the sensitivity vector with respect to the i -component of the parameter vector x_0

$$\lambda_i(t, x_0^j) = \left[\frac{\partial x_L(t, x_0)}{\partial x_{0,i}} \right]_{x_0=x_0^j}. \quad (16)$$

It is obtained as state-variables (or even output-variables) of the i -th *linearization sensitivity model* at nominal point x_0^j . That will be shown as follows.

4.2. The linearization sensitivity model

Relation (13) implies that there are necessary n sensitivity models of the system (5) to compute all the n sensitivity vectors with respect to linearization parameter x_0 at nominal value x_0^j .

Since $j = \overline{1, k}$, then k set of n sensitivity models may be used to obtain all the $k \times n$ set of the sensitivity vectors, that means $k \times n \times n$ linearization sensitivity functions.

The simplest method to obtain a set of n sensitivity functions, presented in [4-6] is to obtain differential equations for the trajectory sensitivity functions, which may then be solved by computer simulation. The system of the differential equations for the trajectory sensitivity functions of the system (5) with respect to $x_{0,i}$ at nominal point x_0^j will be named the i -th *linearization sensitivity model* at nominal point x_0^j .

Differentiating (5) with respect to $x_{0,i}$ (assuming such derivatives exist), and interchanging the $x_{0,i}$ and t partials on the left-hand side (assuming this is permissible), we obtain

$$\begin{aligned} \dot{\lambda}_i(t, x_0^j) &= \left. \frac{\partial A(x_0)}{\partial x_{0,i}} \right|_{x_0=x_0^j} \cdot (x_L(t, x_0^j) - x_0^j) + \\ &+ A(x_0^j) \cdot (\lambda_i(t, x_0^j) - I_i) + \left. \frac{\partial B(x_0)}{\partial x_{0,i}} \right|_{x_0=x_0^j} \cdot \\ &\cdot (u(t) - u_0) + B(x_0^j) \cdot \left. \frac{\partial u_0}{\partial x_{0,i}} \right|_{x_0=x_0^j}. \end{aligned} \quad (17)$$

$$\lambda_i(t_0; x_0^j) = 0, \quad u(t_0) = u_0$$

where the term $x_L(t, x_0^j)$ is given by

$$\dot{x}_L(t, x_0^j) = A(x_0^j) \cdot (x_L(t, x_0^j) - x_0^j) + B(x_0^j) \cdot (u(t) - u_0) \quad (18)$$

$$x_L(t_0; x_0^j) = x_0^j, \quad u(t_0) = u_0. \quad (19)$$

To obtain the desired sensitivity functions it is necessary to simulate together models (17) and (18). Then $k \times n$ number of simulations are required to estimate in the best possible manner the value x_0^* of x_0 that solve (12).

4.3. The lowest number of the linearization sensitivity models

It is of considerable practical and theoretical interest to obtain the linearize sensitivity functions by using the lowest number of simulations possible.

In this section, we shall restrict our attention to the simplest form of nonlinear system model. This is the particular case of the single-input nonlinear system with a *cvasicanonical form* (later we explain this term) in state-space model

$$\begin{aligned} \dot{x}_1(t) &= x_2(t) \\ \dot{x}_2(t) &= x_3(t) \\ &\vdots \end{aligned} \quad (20)$$

$$\begin{aligned} \dot{x}_{n-1}(t) &= x_n(t) \\ \dot{x}_n(t) &= f_n(x_1(t), x_2(t), \dots, x_n(t), u(t)) \\ x_1(t_0) &= x_0, \quad x_2(t_0) = 0, \dots, x_n(t_0) = 0. \end{aligned} \quad (21)$$

In this case it is easy to remark that x_0 can be regarded as a scalar. Then, the stationary point Λ_0 has the coordinates $(u_0, x_0, 0, \dots, 0)$, where u_0 fulfils the condition

$$f_n(x_0, 0, \dots, 0, u_0) = 0, \quad u_0 = u(t_0). \quad (22)$$

For (20), around the stationary point Λ_0 , the linear model can be described by (5) with initial conditions (4), rewritten in the form

$$\begin{bmatrix} \dot{x}_{L,1}(t; x_0) \\ \dot{x}_{L,2}(t; x_0) \\ \vdots \\ \dot{x}_{L,n}(t; x_0) \end{bmatrix} = A(x_0) \cdot \begin{bmatrix} x_{L,1}(t; x_0) \\ x_{L,2}(t; x_0) \\ \vdots \\ x_{L,n}(t; x_0) \end{bmatrix} - \begin{bmatrix} x_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} + B(x_0) \cdot (u(t) - u_0) \quad (23)$$

where

$$A(x_0) = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & 0 & 1 \\ \frac{\partial f_n}{\partial x_1} & \frac{\partial f_n}{\partial x_2} & \dots & \frac{\partial f_n}{\partial x_{n-1}} & \frac{\partial f_n}{\partial x_n} \end{bmatrix}_{\Lambda = \Lambda_0} \quad (24)$$

$$B(x_0) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \frac{\partial f_n}{\partial u} \Big|_{\Lambda = \Lambda_0} \end{bmatrix}$$

The companion canonic form of the matrix $A(x_0)$ explains the term *cvasicanonical form* previously used. For convenience, let us introduce the (28) notation.

$$a_i(x_0) = - \frac{\partial f_n}{\partial x_{i-1}} \Big|_{\Lambda = \Lambda_0}, \quad i = \overline{1, n} \quad (25)$$

$$b(x_0) = \frac{\partial f_n}{\partial u} \Big|_{\Lambda = \Lambda_0}$$

Then relations (24) become

$$A(x_0) = \begin{bmatrix} 0 & 1 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \dots \\ -a_0(x_0) & -a_1(x_0) & \dots & -a_{n-1}(x_0) \end{bmatrix} \quad (26)$$

$$B(x_0) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ b(x_0) \end{bmatrix}$$

The initial conditions for (23) are given by

$$\begin{bmatrix} x_{L,1}(t_0; x_0) \\ x_{L,2}(t_0; x_0) \\ \vdots \\ x_{L,n}(t_0; x_0) \end{bmatrix} = \begin{bmatrix} x_0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (27)$$

Following the above reasoning, only a single linearization sensitivity model has to be obtained for each nominal value x_0^j

$$\begin{aligned} \dot{\lambda}(t; x_0^j) &= \frac{\partial A(x_0)}{\partial x_0} \Big|_{x_0 = x_0^j} \cdot (x_L(t; x_0^j) - x_0^j \cdot I_1) + \\ &+ A(x_0^j) \cdot (\lambda(t; x_0^j) - I_1) + \frac{\partial B(x_0)}{\partial x_0} \Big|_{x_0 = x_0^j} \cdot (u(t) - \\ &- u_0) + B(x_0) \cdot \frac{\partial u_0}{\partial x_0} \Big|_{x_0 = x_0^j}, \quad (28) \\ \lambda(t_0; x_0^j) &= 0, \quad u(t_0) = u_0 \end{aligned}$$

where I_1 is a $n \times 1$ -vector and

$$\frac{\partial A(x_0)}{\partial x_0} \Big|_{x_0 = x_0^j} = \begin{bmatrix} 0 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 0 \\ \frac{\partial a_0}{\partial x_0} & \frac{\partial a_1}{\partial x_0} & \dots & \frac{\partial a_{n-1}}{\partial x_0} \end{bmatrix}_{x_0 = x_0^j}$$

$$\frac{\partial B(x_0)}{\partial x_0} \Big|_{x_0 = x_0^j} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \frac{\partial b(x_0)}{\partial x_0} \end{bmatrix}_{x_0 = x_0^j}; \quad I_1 = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}. \quad (29)$$

The term $x_L(t; x_0^j)$ needed as input in (28) may be obtained from (23), and (24) making $x_0 = x_0^j$.

It is interesting to observe that in the case under consideration there are necessary k simulations to obtain all the $k \times n$ sensitivity functions required to solve equation (12).

It is necessary an example to appreciate the quantitative aspect of the theoretical approach described above.

5. CASE STUDY

In this section, a numerical example will be presented in detail.

5.1. Analytical models

Consider the second order nonlinear system described in [7-8] by the following equation

$$m \cdot \ddot{y}(t) + K_a \cdot \dot{y}(t) + a \cdot (1 - e^{-b \cdot y(t)}) = F_e(t) \quad (30)$$

$$y(0) = x_0, \quad \dot{y}(0) = 0 \quad (31)$$

where $y(t)$ is the measured output variable, $y \in [1 \cdot 10^{-3} \text{ m } 3 \cdot 10^{-2} \text{ m}]$, $F_e(t)$ is the input function $F_e \in [300, 45 \text{ N } 3000 \text{ N}]$ and $\alpha = [m, K_a, a, b]^T = [300 \text{ kg}, 6 \cdot 10^4 \text{ N} \cdot \text{s/m}, 3157, 2 \text{ N}, 100 \text{ m}^{-1}]$ is the parameter vector, well known.

Define states x_1, x_2 such that

$$x_1(t) = y(t), \quad x_2(t) = \dot{y}(t). \quad (32)$$

So we have

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= -\frac{K_a}{m} \cdot x_2(t) - \frac{a}{m} \cdot (1 - e^{-b \cdot x_1(t)}) + \frac{1}{m} \cdot F_e(t) \end{aligned} \quad (33)$$

$$x_1(0) = x_0, \quad x_2(0) = 0. \quad (34)$$

The stationary point $\Lambda_0(F_{e,0}, x_0, 0)$, fulfils the condition

$$-\frac{a}{m} \cdot (1 - e^{-b \cdot x_0}) + \frac{1}{m} \cdot F_{e,0} = 0, \quad F_{e,0} = F_e(t_0) \quad (35)$$

then

$$F_{e,0} = a \cdot (1 - e^{-b \cdot x_0}). \quad (36)$$

The real nonlinear system (33), (34) may be approximated around the stationary point Λ_0 by the tangent-linear model

$$\begin{aligned} \begin{bmatrix} \dot{x}_{L,1}(t; x_0) \\ \dot{x}_{L,2}(t; x_0) \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -\frac{abe^{-bx_0}}{m} & -\frac{K_a}{m} \end{bmatrix} \cdot \begin{bmatrix} x_{L,1}(t; x_0) - x_0 \\ x_{L,2}(t; x_0) \end{bmatrix} + \\ &+ \begin{bmatrix} 0 \\ \frac{1}{m} \end{bmatrix} \cdot (F_e(t) - F_{e,0}) \end{aligned} \quad (37)$$

$$x_{L,1}(0; x_0) = x_0, \quad x_{L,2}(0; x_0) = 0. \quad (38)$$

Consider now that the linear model (37) has a single parameter, $x_0, x_0 \in X_0$,

$$\begin{aligned} X_0 &= \left\{ x_0^j \right\}_{j=1,6} = \\ &= \{0.001 \quad 0.005 \quad 0.01 \quad 0.015 \quad 0.02 \quad 0.025\}. \end{aligned} \quad (39)$$

The output responses, $y = x_1$ and $x_{L,1}$, for an input step function are shown in figure 3.

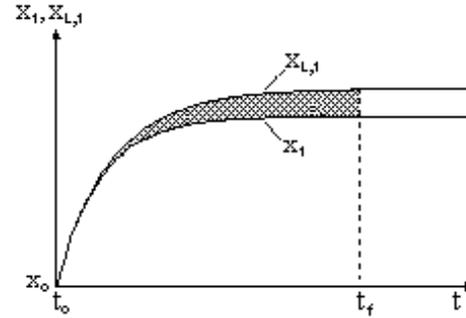


Fig.3. The nonlinear and linearized systems time responses.

The linearization sensitivity model (28) takes the form (after combining similar terms)

$$\begin{aligned} \begin{bmatrix} \dot{\lambda}_1(t; x_0^j) \\ \dot{\lambda}_2(t; x_0^j) \end{bmatrix} &= \begin{bmatrix} 0 & 0 \\ \frac{ab^2 e^{-bx_0^j}}{m} & 0 \end{bmatrix} \cdot \begin{bmatrix} x_{L,1}(t; x_0^j) - x_0^j \\ x_{L,2}(t; x_0^j) \end{bmatrix} + \\ &+ \begin{bmatrix} 0 & 1 \\ -\frac{abe^{-bx_0^j}}{m} & -\frac{K_a}{m} \end{bmatrix} \cdot \begin{bmatrix} \lambda_1(t; x_0^j) \\ \lambda_2(t; x_0^j) \end{bmatrix} \\ \begin{bmatrix} \lambda_1(0; x_0^j) \\ \lambda_2(0; x_0^j) \end{bmatrix} &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}. \end{aligned} \quad (40)$$

It is necessary to simulate together models (40) and (37), obviously for each $x_0 \in X_0$, to compute and plot the two functions $I_1(x_0), I_2(x_0)$ of the two sides of (12).

5.2. Simulations and results

As has been shown above, six number of simulations are required to estimate in the best possible manner the value $x_0 = x_0^*$ that solve (12). In fact this number is double. Two simulations for each x_0^j are necessary, to touch the lower limit x_{\min} and the upper limit x_{\max} of $x(t)$.

Those simulations can be performed with the aid of MATLAB - SIMULINK computer programs. All the simulations were performed with the step-input functions. The time interval of simulations $[t_0=0, t_f]$ is fixed in such a way that all (and only) the transient responses are included.

The numerical results obtained by simulations are presented in table 1 and the graphical results are shown in figures 4 and 5. Thus, figure 4 illustrates the variations of the two functions $I_1(x_0)$ and $I_2(x_0)$, with respect to x_0 and figure 5 gives the performance index $J(x_0)$.

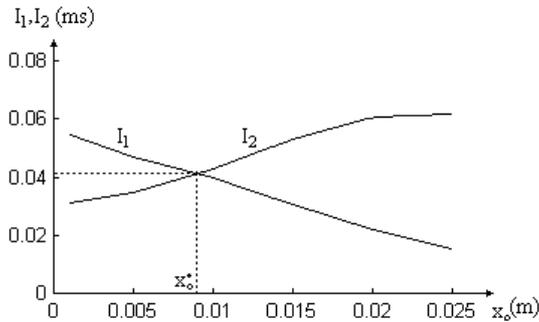


Fig.4. The I_1, I_2 dependencies with respect to x_0 .

Table 1. The computed values by simulations.

x_0 [m]	I_1 [ms]	I_2 [ms]	J [m ² s]
0.001	$5.4752 \cdot 10^{-2}$	$3.0852 \cdot 10^{-2}$	$2.5258 \cdot 10^{-4}$
0.005	$4.6860 \cdot 10^{-2}$	$3.4403 \cdot 10^{-2}$	$1.6302 \cdot 10^{-4}$
0.010	$3.9651 \cdot 10^{-2}$	$4.2642 \cdot 10^{-2}$	$3.9241 \cdot 10^{-4}$
0.015	$3.0799 \cdot 10^{-2}$	$5.3125 \cdot 10^{-2}$	$1.0218 \cdot 10^{-3}$
0.020	$2.1897 \cdot 10^{-2}$	$6.0808 \cdot 10^{-2}$	$1.7820 \cdot 10^{-3}$
0.025	$1.5120 \cdot 10^{-2}$	$6.1546 \cdot 10^{-2}$	$2.4746 \cdot 10^{-3}$

We are now in a position to determine the value of x_0^* . We observe that in this example, the equality $I_1(x_0) = I_2(x_0)$ is satisfied for x_0^* . Since both $I_1(x_0)$ and $I_2(x_0)$ are monotonic a faster interpolation was used to find x_0^* .

Therefore, the function $J(x_0)$ reaches its minimum at

$$x_0^* \approx 9.0318 \cdot 10^{-3} \text{ m}, J(x_0^*) \approx 3.2320 \cdot 10^{-4}. \quad (41)$$

In this case the solution was unique.

Accuracy is limited by the number of the initial data set X_0 and by the tolerance of computer calculations.

The values of J and the figure 5 have been made only to confirm the correctness of the procedure above and to verify that x_0^* is the desired solution.

It should be observed that a small number of simulations and computations are necessary to find x_0^* by using the procedure specified above in this paper then in an elementary manner from $J(x_0)$.

The most general conclusions and how the method described above can be extended are shown as follows.

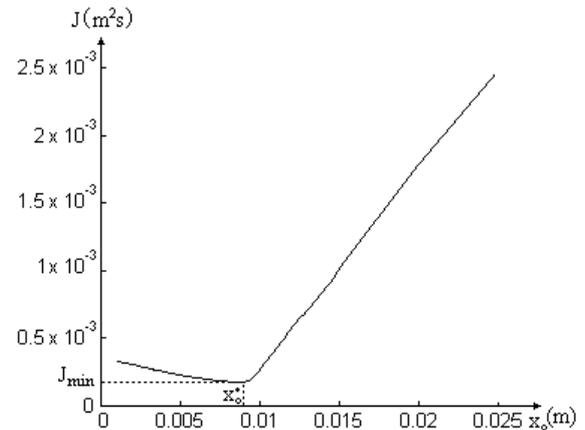


Fig.5. The performance index J with respect to x_0 .

6. CONCLUSIONS AND FUTURE DEVELOPEMENTS

The method developed provides a promising tool for a variety of problems, including linear or nonlinear systems optimization. An application to a composite system, whether it is linear or nonlinear, is straightforward. The effectiveness of the method is illustrated by examining a typical example. Since the procedure is completely systematic, it can efficiently be used in a computer-programmed computation.

MATLAB is the best interactive program to help us with numeric computation. As an extension of MATLAB, Simulink is a tool for modeling, analyzing and simulating mathematical system models, linear and nonlinear. Those are the reasons that recommend them to be used in this paper.

To appreciate the main advantages of the theory here developed, it suffices to observe that for a controllable single input linear model (system) described by the state equations (1) a nonsingular transformation in almost cases (always for linear systems) exists such that a cvasicanonical model (20) can be made. It is

clear that the theory is valid for the most general nonlinear systems of interest.

Several other linear functionals, very useful from the practical point of view, can be considered as well as performance index in further approach, in a way analogous to that.

7. REFERENCES

- [1] Sobral, M., - "Sensitivity in Optimal Control Systems", *Proceeding of the IEEE*, vol. 56, no.10, **1644-1651**, 1968.
- [2] Sarachic, P.E. - "Applications of Functional Analysis to Nonlinear Control Systems with Unknown Plants", *Feedback Systems*, 14, McGraw-Hill, Inc., **269-318**, 1972.
- [3] Newcomb, R.B. and Anderson, B.D.O. - "A Distributional Approach to Time-Varying Sensitivity", *SIAM J. Appl. Math.*, 15, **1001-1009**, 1967.
- [4] Wilkie, D. F. and Perkins, W. R., - "Essential Parameters in Sensitivity Analysis", *Automatica*, 5, **191-197**, 1969.
- [5] Frank, P.M. - "Empfindlichkeitsanalyse dynamischer Systeme", R. Oldenbourg Verlag, München, **24-29**, 1976.
- [6] Rosenwasser, E., Yusupov, R., - "Sensitivity of automatic control systems", CRC Press Boca Raton, Florida, **369-390**, 2000.
- [7] Popescu, D., - "An Approach to Solve an Inverse Sensitivity Problem", *Periodica Politechnica, Transactions on Automatic Control and Computer Science*, 47 (61), Timișoara, **107-110**, 2002.
- [8] Popescu, D. - "Two discrete time approaches to solve the inverse sensitivity problem", *Proceedings CSCS - 14*, (1) Bucuresti, **191-194**, 2003.