A Mathematical Description of Nonlinear Systems and Its Application to Quantized Control Problems

Watanabe, Kajiro / SHIMIZU, Hatsuo

(出版者 / Publisher)
法政大学工学部

(雑誌名 / Journal or Publication Title)
法政大学工学部研究集報 / 法政大学工学部研究集報

(巻 / Volume)
15

(開始ページ / Start Page)
97

(終了ページ / End Page)
105

(発行年 / Year)
1979-03

(URL)
https://doi.org/10.15002/00004150

A Mathematical Description of Nonlinear Systems and Its Application to Quantized Control Problems

Hatsuo SHIMIZU* and Kajiro WATANABE**

Abstract

A nonlinear model (NBL model), based on a linearized model of the given nonlinear system, has been presented with an application to quantized control problems. In this model a system nonlinearity is regarded as a kind of disturbance to the linearized model. A high-speed and simple procedure for computing transient responses of the NBL model can be derived by use of the above thinking and the trapezoidal rule. From this treatment and disposition, a high-speed computational algorithm for determining quantized controls can be constructed by the steepest descent method. Finally, some examples were computed by the presented algorithm.

1. Introduction

For obtaining optimal controls a linearized mathematical model is usually adopted at first even if the actual system is described by the nonlinear differential equation. State variables of the linearized model, however, behave frequently beyond linearization assumptions under the optimizing computation. In this case the computed results are unreliable, so that the linearized model must be modified by recovering nonlinearities of the system and its optimal controls also must be computed all over again from the first.

Several approximation methods have been reported for determining suboptimal control laws for nonlinear systems ; e. g., the method of instantaneous linearization [1], the perturbation method [2], and methods of parameter optimization [3]-[5]. In addition, for solving the quantized control problems Havira *et al.* presented a computational method by use of a piecewise differential dynamic programming algorithm [6]. However, these methods are not always suitable for the above approach and are poor in high-order system problems. For solving this trouble a nonlinear model based on the linearized model (abbreviated NBL model) has been presented with an application to the optimal control problem [7].

In this paper a quantized control problem is formulated under the NBL model to construct an optimizing algorithm based on the steepest descent method. Here a one-dimensional minimization technique is also considered to improve the numerical convergence of the quantized control computation. Finally, some quantized controls are computed by the presented algorithm.

2. Mathematical description of nonlinear systems

The system with nonlinear elements (e. g., softening function, hardening function,

* Graduate Student Department of Instrument and Control Engineering

** Associate Prof. Department of Instrument and Control Engineering

saturation, deadzone, backlash, hysteresis, and relay) is described by the nonlinear differential equation

$$\dot{X}(t)=f(X,t)+Bu(t) \qquad (1)$$

where $X(t)$ is the n-dimensional state vector, $u(t)$ is the m-dimensional control vector, $f(X,t)$ is an n-dimensional nonlinear function, and B is the $n\times m$ control matrix.

The linearization by the Taylor-series expansion of (1) around an operating point $X_0(t)$ yields

$$\dot{x}(t)=Ax(t)+Bu(t) \qquad (2)$$

where $x(t)$ is the n-dimensional state vector newly-defined, and A is the $n\times n$ system matrix. This linearized model (2) is used as the basic model for the NBL model.

Consider now a construction of the NBL model in mathematical description. Each of output signals $r^j(x,t)$ $(j=1,\cdots,l)$ from nonlinear elements in (1) is determined by the magnitude and past history of each input signal $z^j(x,t)$. Around $X_0(t)$ the input-output relation of the jth nonlinear element is written by

$$r^j(x,t)=G_n{}^j z^j(x,t) \qquad (3)$$

where the jth nonlinear gain $G_n{}^j$ is normalized as a whole in order to approach to the linear gain 1. Such a nonlinear gain $G_n{}^j$ can be divided into the linear gain 1 and a deviated nonlinear gain $G_d{}^j$ as follows:

$$G_n{}^j=1+G_d{}^j \qquad (4)$$

Substitution of (4) into (3) gives

$$r^j(x,t)=z^j(x,t)+G_d{}^j z^j(x,t) \qquad (5)$$

If the deviated term $G_d{}^j z^j(x,t)$ in (5), the nonlinearity, is regarded as a kind of disturbance to the linearized model (2), then the nonlinearity can be recovered as in Fig. 1 by adding the equivalent signal $u_e{}^j(t)$, which is equal to $G_d{}^j z^j(x,t)$, to the point where the jth nonlinear element was located.

By applying this idea to all of nonlinear elements in (1) the following nonlinear model based on (2) can be constructed:

$$\dot{x}(t)=Ax(t)+Bu(t)+B_e u_e(t) \qquad (6)$$

where B_e is an $n\times l$ coefficient matrix which identifies the location of nonlinear elements and $u_e(t)$ is an l-dimensional vector. Since $r(x,t)$ and $z(x,t)$, which are l-dimensional vectors, are obtained by the input-output information of nonlinear elements, $u_e(t)$ is computed by

$$u_e(t)=r(x,t)-z(x,t) \qquad (7)$$

with

$$z(x,t)=C_e x(t) \qquad (8)$$

where C_e is an $l\times n$ coefficient matrix. The above model (6) with (7) and (8) is the proposing NBL model.

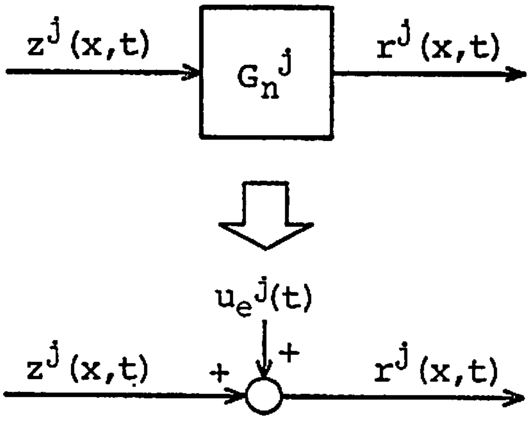


Fig. 1 Recovery of the nonlinearity

3. Transient response computation of the nonlinear system described by the NBL model

3.1 *Computational procedure*

Transient responses of the NBL model (6) can be computed directly by an explicit numerical method for solving the ordinary differential equation (e. g., the Runge-Kutta method). The stepwidth τ for the transient response computation by the usual explicit method is determined by considering the system highest frequency component, but a failure in its determination leads a qualitatively different solution from the theoretical one. The behaviores of the numerical solution (or the transient responses) are usually diverged (such a numerical trouble is called the numerical instability). In this case the stepwidth τ must be determined nallower than the necessary one to avoid the numerical instability. This disposition, however, increases the computational amount. For solving this numerical trouble the following method is considered.

Let x_k be the n-dimensional solution vector of (6) at the time point $k\tau$. Assume that $u(t)$ and $u_e(t)$ are held to the constant values u_k and u_{ek}, respectively, during the time interval $[k\tau,\ (k+1)\tau]$. This assumption is frequently used to transform the differential equation into the difference equation. Integration of (6) by the trapezoidal rule

$$\int_{k\tau}^{(k+1)\tau} \dot{x}(t)dt$$
$$=\int_{k\tau}^{(k+1)\tau}\left\{A\ (x_{k+1}-x_k)\frac{t}{\tau}+Ax_k+Bu_k+B_eu_{ek}\right\}dt \tag{9}$$

gives

$$x_{k+1}=Px_k+Qu_k+Q_eu_{ek} \tag{10}$$

where

$$P=I+A\tau(I-0.5\,A\tau)^{-1} \tag{11}$$
$$Q=\tau(I-0.5\,A\tau)^{-1}B \tag{12}$$
$$Q_e=\tau(I-0.5\,A\tau)^{-1}B_e \tag{13}$$

Furthermore, (7) and (8) become

$$u_{ek}=r_k-z_k \tag{14}$$
$$z_k=C_ex_k \tag{15}$$

at the time point $k\tau$, then u_{ek} is obtained by the above relations (14) and (15).

Therefore, a computational procedure for the NBL model is provided as follows:

0) compute P, Q, and Q_e
1) $z_k=C_ex_k$
2) $u_{ek}=r_k-z_k$
3) $x_{k+1}=Px_k+Qu_k+Q_eu_{ek}$
4) let $k+1$ be k and return to 1).

The matrix function (11) is one of the Padé expansions of $e^{A\tau}$ and (10) is the approximate discretization of (6) by use of the approximate shift operator $(1+0.5\,s\tau)/(1-0.5\,s\tau)$ which is the Padé 1/1 type expansion of $e^{s\tau}$. On the transient response computation of (2) this discretization always guarantees the numerical stability [8].

Therefore, it is expected that τ can be determined wider than any one of the usual explicit methods (i. e., the high-speed transient response computation can be expected).

3. 2 *Numerical stability*

Assume that all characteristic roots $a^j (j=1, \cdots, n)$ of (2) are located in the left half of the complex plane (abbreviated LHP). Consider the jth characteristic root deviated into $a^j+\varepsilon^j$ (located in the LHP) from a^j by the influence of system nonlinearities. Then the NBL model is written by

$$\dot{x}^{*j}(t)=a^j x^{*j}(t)+a^j u_e^{*j}(t) \tag{16}$$

with

$$u_e^{*j}(t)=\frac{\varepsilon^j}{a^j}x^{*j}(t) \tag{17}$$

By use of the Padé 1/1 type expansion, (16) is transformed into

$$x^{*j}_{k+1}=\frac{1+0.5a^j\tau}{1-0.5a^j\tau}x^{*j}_k+\frac{a^j\tau}{1-0.5a^j\tau}u_e{}^{*j}_k \tag{18}$$

with

$$u_e{}^{*j}_k=\frac{\varepsilon^j}{a^j}x^{*j}_k \tag{19}$$

Substitution of (19) into (18) yields

$$x^{*j}_{k+1}=P^j_k x^{*j}_k \tag{20}$$

where

$$P^j_k=\frac{1+0.5\,a^j\tau+\varepsilon^j\tau}{1-0.5\,a^j\tau} \tag{21}$$

Let

$$a^j\tau=-\alpha^j+i\beta^j,\ \ \varepsilon^j\tau=-\sigma^j+i\omega^j \quad (i=\sqrt{-1}) \tag{22}$$

and substitution of (22) into (21) gives

$$P^j_k=\frac{1+0.5(-\alpha^j+i\beta^j)+(-\sigma^j+i\omega^j)}{1-0.5(-\alpha^j+i\beta^j)} \tag{23}$$

In order to guarantee the stable computation of (20), the following condition must be satisfied

$$|P^j_k|<1.0 \tag{24}$$

Then the jth numerical stable region for (20) is obtained as follows:

$$\{W^j-(-1-0.5\,\alpha^j)\}^2+(Y^j-0.5\,\beta^j)^2 < (-1-0.5\,\alpha^j)^2+(0.5\,\beta^j)^2 \tag{25}$$

where $W^j=-(\alpha^j+\sigma^j)$ and $Y^j=\beta^j+\omega^j$, which is shown in Fig. 2.

Then the stable computation of (20) can be guaranteed while $\varepsilon^j\tau$ travels in the circle as in Fig. 2. Therefore, the numerical stability for the computation of (6) is clearly guaranteed while each of $\varepsilon^j\tau (j=1, \cdots, n)$ travels in each circle, respectively.

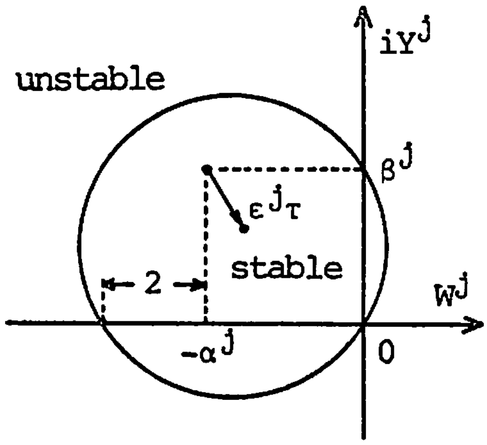


Fig. 2 Numerical stable region

4. Application to quantized control problems

The nonlinear system described by the NBL model

$$\dot{x}(t)=Ax(t)+Bu(t)+B_e u_e(t) \tag{6}$$

with

$$x(t_0)=x_0 \tag{26}$$

is considered. The problem is to choose the m-dimensional quantized control vector $u(t)$ to minimize

$$J=\frac{1}{2}\int_{t_0}^{t_f}\{x^T(t)Fx(t)+u^T(t)Ru(t)\}dt \tag{27}$$

where F is the $n\times n$ positive semi-definite weighting matrix, R is the $m\times m$ positive definite weighting matrix, t_0 and t_f are fixed, and $x(t_f)$ is free. Quantized levels of $u(t)$; $(0,\ \pm q_1,\ \pm q_2, \cdots,\ \pm q_M)$, are determined to minimize (27) and each of determined levels is held to the constant value during each time interval $[k\tau_q,\ (k+1)\tau_q]$. Here τ_q is the stepwidth for the quantized control computation.

This quantized control problem can be solved by the steepest descent method which is one of the gradient methods [9], [10]. The Hamiltonian $H(t)$ can be defined as

$$H(t)=[Ax(t)+Bu(t)+B_e u_e(t)]^T\psi(t)+\frac{1}{2}\{x^T(t)Fx(t)+u^T(t)Ru(t)\} \tag{28}$$

where

$$\dot{\psi}(t)=-\left[A+B_e\frac{\partial u_e(t)}{\partial z^T(x,t)}C_e\right]^T\psi(t)-Fx(t) \tag{29}$$

with

$$\psi(t_f)=0 \tag{30}$$

Then the gradient $g(t)$ is

$$g(t)=\frac{\partial H(t)}{\partial u(t)}=B^T\psi(t)+Ru(t) \tag{31}$$

When the Lth approximate quantized control $u_L(t)$ is given, the Lth gradient $g_L(t)$ is obtained by solving (6) forward with (26) under $u(t)=u_L(t)$, solving (29) backward with (30) and then computing $g_L(t)$ from (31). Here (6) is solved by use of the procedure presented in section 3 and $u(t)$ is not interpolated because of its quantized form. For solving (29) the following formula is also presented :

$$\psi_{k+1}=D_k\psi_k+E_k(x_k+x_{k+1})/2 \tag{32}$$

with

$$\psi_0=0 \tag{33}$$

where

$$D_k=I+[M(k\tau)]^T\tau(I-0.5[M(k\tau)]^T\tau)^{-1} \tag{34}$$

$$E_k=\tau(I-0.5[M(k\tau)]^T\tau)^{-1}F \tag{35}$$

$$M(k\tau)=A+B_e\frac{\partial u_e(k\tau)}{\partial z^T(x,k\tau)}C_e \tag{36}$$

and $(x_k+x_{k+1})/2$ is the result of the linear interpolation of $x(t)$ during the time interval $[k\tau,\ (k+1)\tau]$, which improves the numerical accuracy.

Therefore, the algorithm for computing quantized controls is designed as follows :

0) set $u_0(t)=0$

1) compute $g_L(t)$

2) modify $g_L(t)$ to $[g_L(t)]_{mod}$ as in Fig. 3

3) set $s_L(t)=-[g_L(t)]_{mod}$

4) choose $v_L(>0)$ to minimize $J\{[u_L(t)+vs_L(t)]_{quan}\}$ where the value of J is computed by Simpson's rule and the notation $[*]_{quan}$ means the numerical quantization processing of $[*]$ by use of the quantizer as in Fig. 4

5) set $u_{L+1}(t)=[u_L(t)+v_Ls_L(t)]_{quan}$

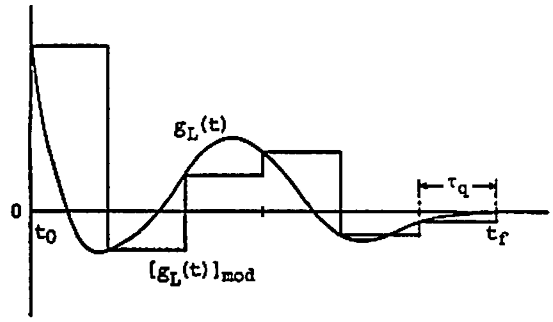


Fig. 3 A modification for $g_L(t)$

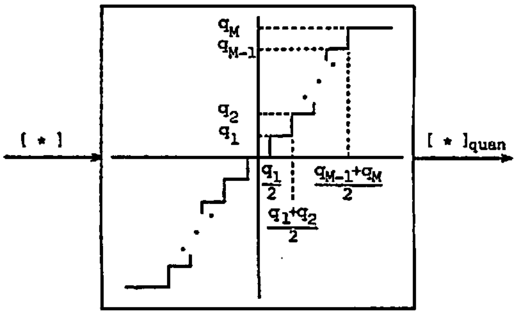
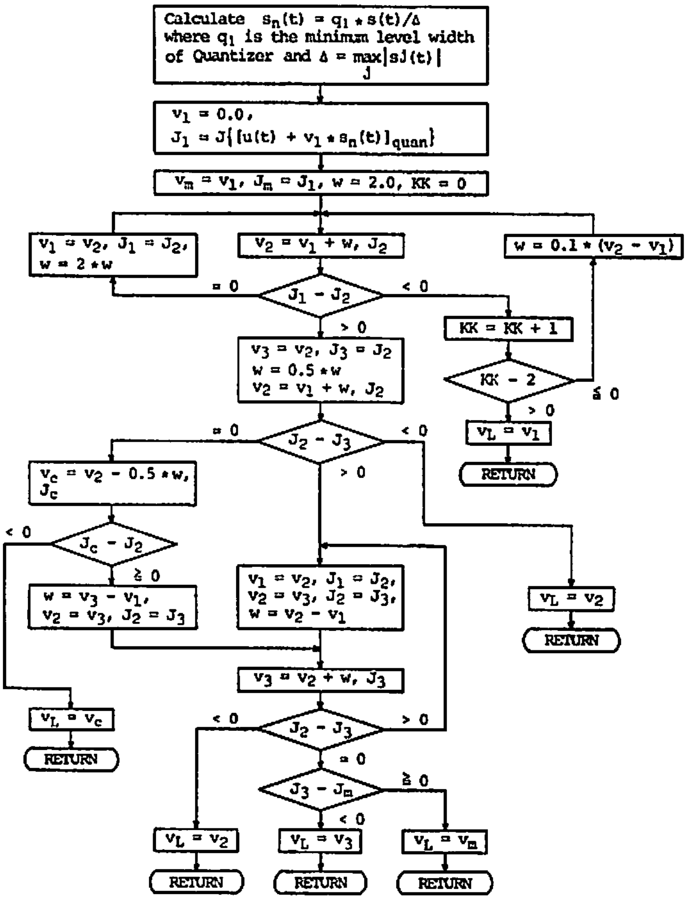


Fig. 4 Quantizer ; (0, $\pm q_1$, $\pm q_2$, ..., $\pm q_M$)



Calculate $s_n(t) = q_1 * s(t)/\Delta$ where q_1 is the minimum level width of Quantizer and $\Delta = \max_J |sJ(t)|$
$v_1 = 0.0$, $J_1 = J\{[u(t) + v_1 * s_n(t)]_{quan}\}$
$v_m = v_1$, $J_m = J_1$, $w = 2.0$, $KK = 0$
$v_1 = v_2$, $J_1 = J_2$, $w = 2 * w$
$v_2 = v_1 + w$, J_2
$w = 0.1 * (v_2 - v_1)$
= 0
$J_1 - J_2$
< 0
> 0
$KK = KK + 1$
$v_3 = v_2$, $J_3 = J_2$, $w = 0.5 * w$, $v_2 = v_1 + w$, J_2
$KK - 2$
≦ 0
> 0
$v_L = v_1$
RETURN
= 0
$J_2 - J_3$
< 0
> 0
$v_c = v_2 - 0.5 * w$, J_c
< 0
$J_c - J_2$
≧ 0
$w = v_3 - v_1$, $v_2 = v_3$, $J_2 = J_3$
$v_1 = v_2$, $J_1 = J_2$, $v_2 = v_3$, $J_2 = J_3$, $w = v_2 - v_1$
$v_L = v_2$
RETURN
$v_3 = v_2 + w$, J_3
$v_L = v_c$
RETURN
< 0
$J_2 - J_3$
> 0
= 0
$J_3 - J_m$
≧ 0
< 0
$v_L = v_2$
$v_L = v_3$
$v_L = v_m$
RETURN
RETURN
RETURN

Fig. 5 A procedure of one-dimensional minimization

6) let $L+1$ be L and return to 1).

The computation of v_L in 4) is important from the point of the numerical convergence (or computational amount). By the simple quadratic [cubic] interpolation method for one-dimensional minimization [9], [10], good results could not always be obtained for the quantized control problems. Here v_L is determined by a procedure as in Fig. 5 which is designed to have a good numerical convergence.

5. Computed examples

The following quantized control problem was considered. Given the controlled system as in Fig. 6 with nonlinear elements as in Fig. 7. Minimize

$$J=J_x+J_u=\int_0^8\{x^T(t)x(t)+0.5u^2(t)\}dt$$

Here $x(0)=[0.0\ \ 0.0\ \ 0.0\ \ 1.0]^T$.

The quantized control $u(t)$ is generated by the following quantizers:

Quantizer 1; (0, ±0.2, ±0.4)

Quantizer 2; (0, ±0.1, ±0.2, ±0.3, ±0.4)

and then $x(8)$ is free.

Quantized controls of some cases were determined by the presented algorithm. Numerical results are summarized in Table 1. In this example the value of J was almost halved after a few iterations of each case. Typical trajectories (transient responses) of Cases 1 and 4 are plotted in Figs. 8 and 9, respectively. Furthermore, $u(t)$ for the linearized model of the given system was determined under the same conditions as Case 4 (Quantizer 2, $\tau_q=1.0$, $\tau=0.25$). The optimizing computation was stopped after 6 iterations. Transient responses, obtained by adding $u(t)$ to the NBL model, are also plotted in Fig. 10. In comparison of Fig. 9 with Fig. 10 it is clear that each of their differences for $x_1(t)$, $x_4(t)$, and J_x is little, respectively, but each effect of

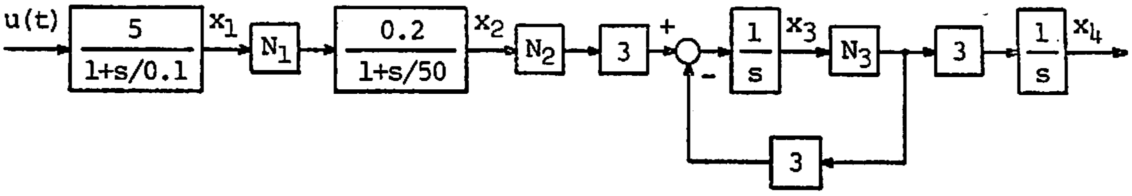


Fig. 6 Controlled system with nonlinear elements (N_1~N_3)

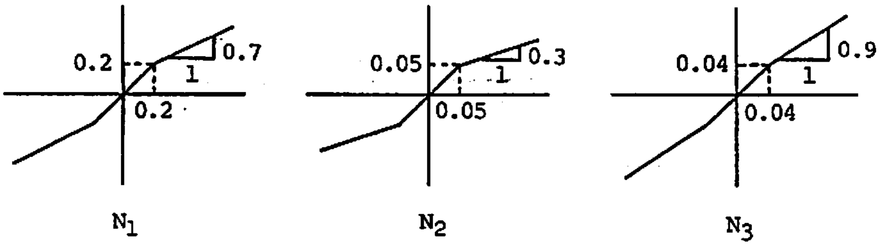


Fig. 7 Characteristics of nonlinear elements (N_1~N_3)

Table 1 Numerical results

Case	1	2	3	4	5	6
Quantizer	1	1	2	2	2	2
τ_q	1. 0	0. 2	1. 0	1. 0	1. 0	0. 5
τ	0. 25	0. 1	0. 5	0. 25	0. 025	0. 25
Number of iterations	4	5	3	3	3	4
Number of J computations	25	38	21	20	21	28
$J=J_x+J_u$	3. 849	3. 811	3. 796	3. 800	3. 805	3. 770
J_u	0. 100	0. 128	0. 110	0. 110	0. 110	0. 135
Computation time in seconds by FACOM 230-45S	5. 4	18. 4	2. 3	4. 2	41. 5	5. 8

$u(t)$ is greatly different.

Computed transient responses of $x(t)$ under the condition $\tau=0.25$ (Case 4) agree well with the results in $\tau=0.025$ (Case 5). Even $\tau=0.5$ (Case 3), the computation was stably under the presented procedure of the NBL model. However, the computation by the 4 th-order Runge-Kutta method is fallen into the numerical instability under the stepwidth condition $\tau=0.25$. In this case τ must be theoretically determined nallower than 0. 056 to guarantee the numerical stability, so that the Runge-Kutta

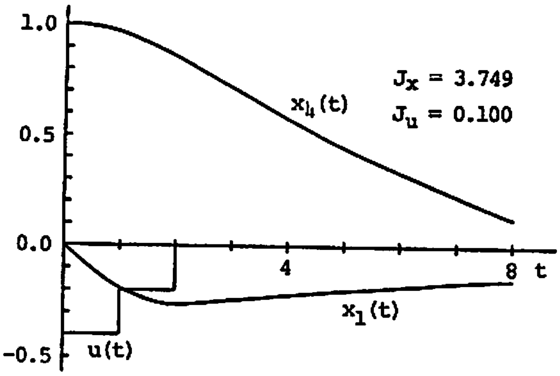


Fig. 8 Trajectories of Case 1

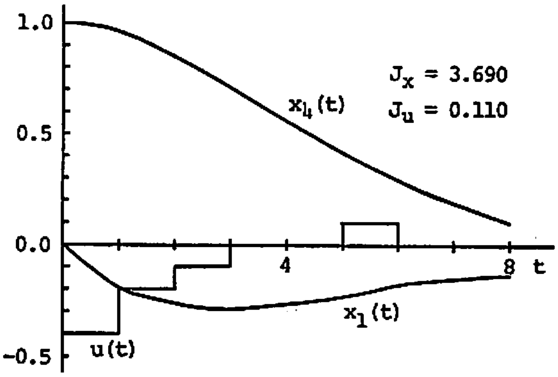


Fig. 9 Trajectories of Case 4

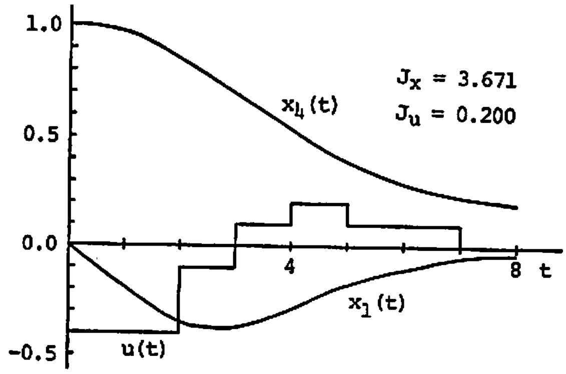


Fig. 10 Corresponding trajectories of Case 4 by the quantized control for the linearized model

method spoiles the high-speed optimizing computation.

6. Conclusion

This paper provided the NBL model, which describes the nonlinear system in the usual linear matrix notation, and derived its transient response computational method with a numerical stability discussion. In addition, an algorithm for computing quantized controls was presented under the concise formulation by use of the NBL model, which can also compute high-order system problems. Its computer program routines are constructed simply by adding the option routines to the linear model optimization codes (computer software systems). Some examples were actually computed by the presented algorithm and the optimization algorithm was effective judging from these computed results.

References

[1] J. D. Pearson, "Approximation Methods in Optimal Control," I. Sub-optimal Control, *J. Electronics and Control* 13, 453/469 (1962)

[2] W. L. Garrard, N. H. McClamroch & L. G. Clark, "An Approach to Sub-optimal Feedback Control of Non-linear Systems," *Int. J. of Control* 5, 425/435 (1967)

[3] R. C. Durbeck, "An Approximation Technique for Suboptimal Control," *IEEE Trans. Automatic Control* AC-10, 144/149 (1965)

[4] A. G. Longmuir & E. V. Bohn, "The Synthesis of Suboptimal Feed-back Control Laws," *ibid.* AC-12, 755/758 (1967)

[5] N. Sannomiya, Y. Nishikawa & H. Itakura, "Suboptimal Design of Nonlinear Regulators by Perturbation Method and Parameter Optimization Technique," *Trans. SICE (Japan)* 8, 355/361 (1972)

[6] R. M. Havira & J. B. Lewis "Computation of Quantized Controls Using Differential Dynamic Programming," *IEEE Trans. Automatic Control* AC-17, 191/196 (1972)

[7] K. Watanabe & H. Shimizu, "A Mathematical Expression of Nonlinear Systems Based on Linear Model and Its Application to Optimal Control Problems," *Trans. SICE (Japan)* 14, 44/49 (1978)

[8] K. Watanabe, "Approximated Calculation Method of the Transition Matrix Using Pade's Approximate Function of exp(x)," *ibid.* 10, 36/44 (1974)

[9] L. S. Lasdon, *Optimization Theory for Large Systems*, Macmillan (1970)

[10] J. Kowalik & M. R. Osborne, *Methods for Unconstrained Optimization Problems*, Elsevier (1968)