AT&P JOURNAL plus 1 2010

Systems of automatic control

reviewed slovak professional magazine for scientific and engineering issues

Systémy automatického riadenia

recenzované periodikum vedeckých a inžinierskych publikácií

ISSN 1336-5010

Systémy automatického riadenia

Systems of automatic control

Odborný garant

prof. Ing. Ladislav Jurišica, PhD.
Slovenská technická univerzita v Bratislave
Fakulta elektrotechniky a informatiky
Ústav riadenia a priemyselnej informatiky
Oddelenie robotiky a umelej inteligencie
Ilkovičova 3, 812 19 Bratislava, Slovensko
Tel.: +421 2 60 29 13 51
Fax: +421 2 65 42 95 21
e-mail: ladislav.jurisica@stuba.sk

Technical guarantee

prof. Ing. Ladislav Jurišica, PhD.
Slovak University of Technology in Bratislava
Faculty of Electrical Engineering and Information Technology
Institute of Control and Industrial Informatics
Department of Robotics and Artificial Intelligence
Ilkovičova 3, 812 19 Bratislava, Slovak Republic
Tel.: +421 2 60 29 13 51
Fax: +421 2 65 42 95 21
e-mail: ladislav.jurisica@stuba.sk

Vydavateľ Publisher

HMH s.r.o.
Tavarikova osada 39
841 02 Bratislava 42
IČO: 31356273

Spoluzakladateľ Co-founder

Katedra ASR, EF STU
Katedra automatizácie a regulácie, EF STU
Katedra automatizácie, ChtF STU
PPA CONTROLL, a.s.

Partnerské organizácie AT&P journalu
AT&P journal partnering organizations

IEEE Czechoslovakia Section www.ieee.cz

ISA

MESA INTERNATIONAL Driving Operations Excellence EUROPE

Redakcia Editors office

AT&P journal
Evidenčné číslo: EV 3242/09
Košická 37, 821 09 Bratislava 2
tel.: 02/5026 1752 – 55
fax: 02/5026 1757
e-mail: info@atpjournal.sk
http://www.atpjournal.sk

Ing. Anton Gérer
šéfredaktor – editor in chief
sefredaktor@atpjournal.sk

Ing. Ildikó Csölleová
vedúca redakcie – editorial office manager
podklady@atpjournal.sk

Bc. Zuzana Bakošová
marketingová manažérka – marketing manager
marketing@atpjournal.sk

Ing. Branislav Bložon
odborný redaktor – editor
redaktor@atpjournal.sk

Ing. Martin Karbovanec
odborný redaktor – editor
karbovanec@atpjournal.sk

Zuzana Pettingerová
technická redaktorka – DTP
podklady@atpjournal.sk

Mgr. Bronislava Chocholová
jazyková redaktorka – text corrector

Datum vydania: máj 2010

ISSN 1336-5010

Obsah

Teória riadenia

Aplikácie riadiacich systémov

Mechatronické systémy

Articles

Control theory

Applications of control systems

Mechatronic systems

Decentralized control: some aspects of stability and performance

Danica Rosinová

Abstract
The paper studies some basic aspects of decentralized control design concerning stability and performance. Control structure selection based on performance relative gain array (PRGA, Hovd, Skogestad, 1992) is used and its ability to evaluate the achievable performance is discussed. Robust stability condition for decentralized control is considered, which provides the upper level on subsystems, thus limiting the performance. The compromise between robust stability and performance is illustrated on example and provides material which can be used in teaching complex systems control.

Keywords: control system design, decentralized control, robust stability

Introduction

Several important aspects of decentralized control design are studied in this paper. The main aim is to design appropriate decentralized control, so that the overall system stability is kept and the required performance specifications are achieved.

Decentralized control design comprises several steps and tasks (Skogestad, 2005):

- study the plant – system to be controlled and formulate the control objective;

- find a plant model, simplify it if necessary;

- analyze the model properties, scale the variables;

- decide which variables are to be controlled and which variables are to be the manipulated ones;

- select the control configuration: for decentralized control structure it means to choose the input – output pairing;

- specify the performance requirements respective to the control objective;

- determine the type of controller and design its parameters;

- examine the resulting control system, if the specified requirements are not met, redesign;

- analyze simulation resluts, if necessary repeat the whole procedure;

- realize the designed controller.

In this paper we concentrate on key aspects of decentralized control design: control structure selection with appropriate input-output pairing and resulting single loops design so that it guarantees stability as well as required performance of the overall system including interactions. Two forms of stability condition for decentralized control structure are used and discussed on the example: one based on small gain theorem and one for systems with no RHP (right half plane) zeros. Standard interaction measure used for control structure selection is the relative gain array (RGA), nevertheless, performance relative gain array PRGA (performance relative gain array - closely related to RGA) is used in this paper, since besides input-output pairing it enables to evaluate the achievable performance (Hovd, Skogestad, 1992). The use of these design tools is illustrated and analysed on case study (decentralized control design for quadruple tank model). The interesting part of simulation experiments is connected with non-minimum phase case, where the single loops are minimum phase, however, the interconnected system includes transmission RHP zero, which inherently limits the required performance. The obtained results show the limitations following from overall stability condition and the influence of interconnections on overall system stability and performance. Presented material provides simple illustration of stability versus performance relationship in decentralized control design and can be used in teaching complex systems control.

1. Problem formulation and preliminaries

Consider a multi-input multi-output plant described by linear MIMO system model

$$Y(s) = G(s)U(s) \qquad (1)$$

where complex vectors $Y(s), U(s)$ are Laplace images of output and input signal of dimensions p and m respectively, $G(s)$ is transfer function matrix of dimensions $p \times m$. In the following we assume the square system model, i.e. $p = m$ and stable plant G.

Our aim is to design appropriate decentralized control, so that the overall system stability is kept (including possible uncertainties) and the required performance is achieved. In this paper we focus on two most important steps in decentralized control design:

1. determining control configuration, which means to choose appropriate input-output pairing;

2. the respective single control loops design so that the overall requirements are kept.

After completing step 1, the inputs or outputs can be reordered, so that the respective transfer system matrix G with reordered columns or rows has the paired elements on the main diagonal. Then the decentralized controller can be represented by the diagonal matrix $K(s) = diag(k_{ij})$. To find $K(s)$ (step 2), the so called independent design is considered, where individual loops are designed "independently" (simultaneously). In other words, local controllers $k_{ij}(s)$ are designed so that they:

a) stabilize individual loops

b) satisfy the overall system stability condition

c) satisfy the bounds obtained from performance requirements.

Note that conditions in b) and c) may be contradictory (as illustrated later in the presented example).

In the following, sensitivity is denoted as $S(s) = (I + G(s)K(s))^{-1}$ and closed loop transfer function is denoted as $T(s) = G(s)K(s)(I + G(s)K(s))^{-1}$. Argument s is often omitted for better readability.

1.1 Control configuration (pairing) selection

To choose appropriate pairing, several interaction measures have been proposed in literature (RGA, dRGA, PRGA, etc.), more details can be found e.g. in (Schmidt, 2002). Relative gain array (RGA), frequently used in practice, is defined as

$$RGA(G) = G(s) \circ \left(G(s)^{-1}\right)^T \tag{2}$$

where $\circ$ denotes entrywise matrix product (Hadamard product).

However, RGA index provides no information e.g. for the system with one way interconnections (when the transfer function matrix is upper or lower triangular). To better evaluate system performance, PRGA index has been introduced and shown to provide performance limits for system with decentralized control (Hovd, Skogestad, 1992). PRGA is defined as

$$PRGA(G) = \Gamma = G_D(s)\, G(s)^{-1} \tag{3}$$

where $G_D(s) \approx diag(G(s))$ denotes the diagonal matrix having diagonal elements of $G(s)$ on its diagonal. The relationship between PRGA and closed loop system performance can be summarized in the following way. Let us specify the required closed loop performance by bounds on control error (offset) and disturbance

$$\left|e_i(j\varpi)/r_j(j\varpi)\right| = \left|S_{ij}(j\varpi)\right| < 1/\left|w_{ri}(j\varpi)\right| \quad \forall \varpi, i, j \tag{4a}$$

$$\left|e_i(j\varpi)/z_k(j\varpi)\right| = \left|[SG_z]_{ik}(j\varpi)\right| < 1/\left|w_{zi}(j\varpi)\right| \quad \forall \varpi, i, k \tag{4b}$$

where r_j denotes setpoint change, S_{ij} is the respective element of sensitivity function S, z_k is the expected disturbance and G_z its transfer function; w_{ri} and w_{zi} are scalar performance weights for control error and disturbance respectively.

For frequencies, where a feedback is effective ($\varpi < \varpi_B$, ϖ_B denotes bandwidth), it can be usually assumed $S = (I + GK)^{-1} \approx (GK)^{-1}$ yielding the following bounds for individual loops

$$\left|g_{ii}k_i(j\varpi)\right| > \left|\gamma_{ij}w_{ri}(j\varpi)\right| \quad \forall \varpi < \varpi_B, \forall i,j$$

γ_{ij} are elements of PRGA index Γ (5a)

$$\left|g_{ii}k_i(j\varpi)\right| > \left|\delta_{ik}w_{zi}(j\varpi)\right| \quad \forall \varpi < \varpi_B, \forall i,k$$

δ_{ik} are elements of ΓG_z (5b)

The above inequalities determine performance limits - lower bounds on single loop modules to achieve the required control error and disturbance attenuation and will be discussed in control design stage.

1.2 Stability condition for decentralized control

After the appropriate pairing has been determined, the decentralized control law is to be designed. There are various approaches to find decentralized control law. We adopt individual design as simple possibility to design single loops so that the overall stability and performance requirements are kept, i.e. that interactions do not introduce instability and do not significantly deteriorate performance. Let us consider stability condition for system with decentralized control. Matrix G can be split into its diagonal and off-diagonal parts: $G = G_D + G_M$. For a stable open loop system GK, the closed loop system stability condition based on small gain theorem is given in the next Lemma (Veselý, Harsányi, 2008).

Lemma 1

Consider a stable system G with a decentralized controller K. The respective closed loop system with transfer function $T(s)$ is stable if

$$\left\|G_D^{-1}W\right\|\left\|G_M\right\| < 1 \tag{6a}$$

or

$$\left\|G_D^{-1}W\right\| < \frac{1}{\left\|G_M\right\|} \tag{6b}$$

where matrix W is given by $R^{-1} + G_D = G_D W^{-1}$ (7)

Inequality (6b) can be reformulated into

$$\left\|G_D^{-1}T_D\right\| < M_0 = \frac{1}{\left\|G_M\right\|} \tag{8}$$

where $T_D = G_D K(I + G_D K)^{-1}$.

Condition (8) can be used for stable system without or with RHP zeros (both for minimum and non-minimum phase case). Note that the above condition can be rather limiting in low frequencies, where $\left\|T_D\right\| \approx 1$, for stable system with no RHP zeros this may be too restrictive. The less restrictive condition for this case can be found in (Skogestad, 2005).

Lemma 2

Consider a stable system G with a decentralized controller K. Assuming that neither G nor G_D has RHP zeros, the overall closed loop system is stable if and only if $\left(I - ES_D\right)^{-1}$ is stable, where

$$E = (G - G_D)G^{-1} = G_M G^{-1},\ S_D = (I + G_D K)^{-1}.$$

The above condition can be reformulated: $(I-ES_D)^{-1}$ stable means $det(I-ES_D)^{-1} \neq 0$. The sufficient stability condition is then $\|ES_D\| < 1$, or

$$\|G^{-1}S_D\| < M_0 = \frac{1}{\|G_M\|}. \qquad (9)$$

Note that whichever condition is used, (8) or (9), it must be satisfied for all frequencies, these two conditions cannot be mixed or combined.

2. Decentralized control design: case study

The decentralized control design approach based on stability condition (8) or, alternatively, (9) and performance bound (5a) is used to find decentralized control guaranteeing overall system stability and bounded control error (4a). The design procedure is illustrated on the case study - decentralized control of quadruple tank process: input variables are two input flows, output variables are water levels in lower two tanks. Detailed description of the plant can be found in (Johansson, 2000), (Rosinová and Markech, 2008). The controlled system is described by the linearized model

$$G(s) = \begin{bmatrix} \dfrac{2.69\,\gamma_1}{47.11s+1} & \dfrac{2.69(1-\gamma_2)}{(47.11s+1)(73.42s+1)} \\ \dfrac{4.39(1-\gamma_1)}{(87.81s+1)(76.05s+1)} & \dfrac{4.39\,\gamma_2}{87.81s+1} \end{bmatrix} \qquad (10)$$

parameters γ_1, γ_2 determine the input flow split between the lower and upper tank, $0 \le \gamma_1, \gamma_2 \le 1$. It is important to note that depending on values of γ_1, γ_2, two different plant configurations can be obtained:

A: minimum phase configuration for $1 < \gamma_1 + \gamma_2 < 2$ (in this case we consider γ_1=0.6, γ_2 =0.8) and

B: non-minimum phase configuration for $0 < \gamma_1 + \gamma_2 < 1$ (in this case we consider γ_1=0.2, γ_2 =0.3).

In the first step, the appropriate input-output pairing is determined using RGA, or PRGA index.

In steady state we have RGA index dependent only on the values of γ_1, γ_2

$$RGA = G(0) \circ \left[G(0)^{-1}\right]^T \cong \begin{bmatrix} \lambda & 1-\lambda \\ 1-\lambda & \lambda \end{bmatrix}, \quad \lambda = \frac{\gamma_1\gamma_2}{\gamma_1+\gamma_2-1} \qquad (11)$$

From (11), obviously the adequate pairings are: $u_1 - y_1$, $u_2 - y_2$ for configuration A, $u_1 - y_2$, $u_2 - y_1$ for configuration B (λ is negative). This choice is supported by frequency dependent RGA index shown in Fig.1a, 1b. In both cases, decentralized control is designed independently for both loops, decentralized PID controller is

$$K(s) = \begin{bmatrix} k_1(s) & 0 \\ 0 & k_2(s) \end{bmatrix} \qquad (12)$$

$$k_i(s) = P_i + I_i/s + D_i s \qquad i=1, 2 \qquad (13)$$

We propose the following decentralized control design strategy.

When designing parameters of $k_i(s)$, in the first step we consider stability criterion (8) or, alternatively, (9) as a bound on loops responses, next step is to shape the loops responses within stability bounds to achieve the required performance specifications "measured" by performance bound (5a).

The application of this strategy on the case study illustrates also the case, when the individual loops design follows only loops criteria without considering overall stability.

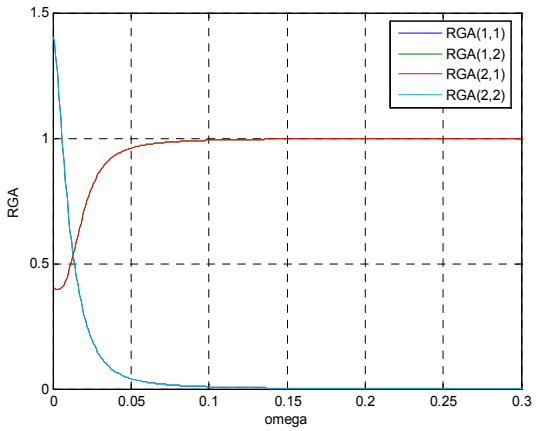


Fig.1a RGA for minimum phase case (γ_1=0.6, γ_2=0.8)

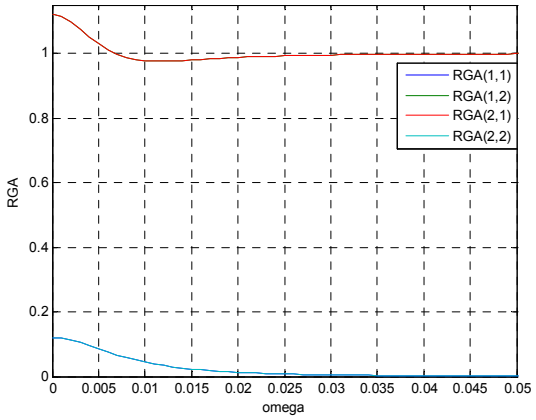


Fig.1b RGA for non-minimum phase case (γ_1=0.2, γ_2=0.3)

In decentralized control design below, the results are evaluated according to the respective plots. The overall stability condition (8) or (9) is examined from the plots of left hand side and right hand side modules: red line denotes the upper bound (right hand side of the inequality), to satisfy the inequality (8) or (9), the respective blue or green lines should be for all frequencies below the red one.

The performance criterion is checked through the plots respective to w_{ri} from (5a); recall that $1/w_{ri}$ provides the upper bound on control error (within the frequency range, where the feedback is effective), e.g. w_{ri} = 20 corresponds to control error less than 5%.

2.1 Decentralized control for a minimum phase configuration A

The results for PI controller parameters designed for individual loops: P_1=2.917, I_1=0.0619; P_2=2.50, I_2=0.0285; are shown in Fig.2a, 2b and 3. Fig. 2a shows that stability condition (8) is not satisfied in this case – blue line is for low frequencies above the red one, however, the overall stability is guaranteed by (for this case) the less restrictive condition

(9), which is satisfied since green line is below the red one for all frequencies. The respective step responses show minor differences between individual loops (lower plots) and the overall system (upper plots) performance.

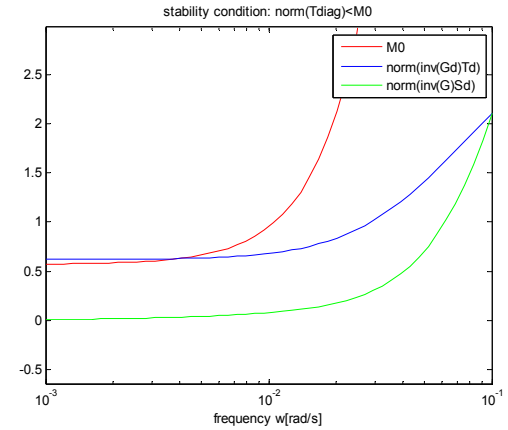


Fig.2a Decentralized control design: stability condition (satisfied)

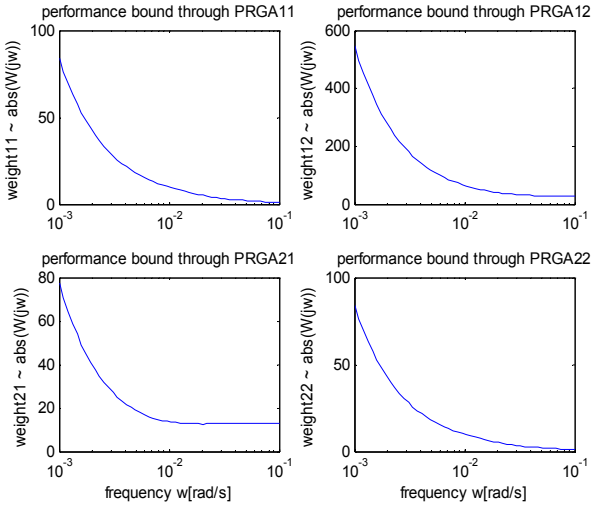


Fig.2b Decentralized control design: performance bound

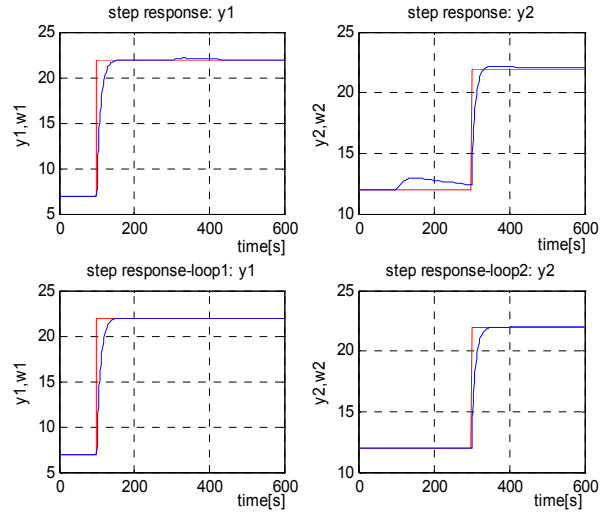


Fig.3 Decentralized step responses: Interconnected system – upper plots, individual loops – lower plots

2.2 Decentralized control for non-minimum phase configuration B

This configuration is characterized by the existence of transient RHP zeros (while individual transfer functions have no RHP zeros), which complicates the decentralized controller design. We illustrate the impact of interactions on three different designs of control loops.

In the first case the overall stability condition is not satisfied, though the individual loops have satisfactory performance indices. PI controller parameters: P_1=0.9360, I_1=0.0156; P_2=2.7725, I_2=0.0210. As shown in Fig.4a, the overall system stability condition (8) is not satisfied (around the bandwidth frequency), therefore performance indices have no reason. Step responses in Fig.5 show the significant differences between individual loops (both are stable and damped) and the overall system, which is unstable.

The next case (P_1= 0.4680, I_1= 0.0052; P_2= 1.1090, I_2= 0.0084) shows that as soon as the condition (8) is satisfied (Fig.6a: blue line below the red one), the overall system responses are similar to single loop ones – Fig.7; performance indicators are still satisfactory for low frequencies (Fig.6b).

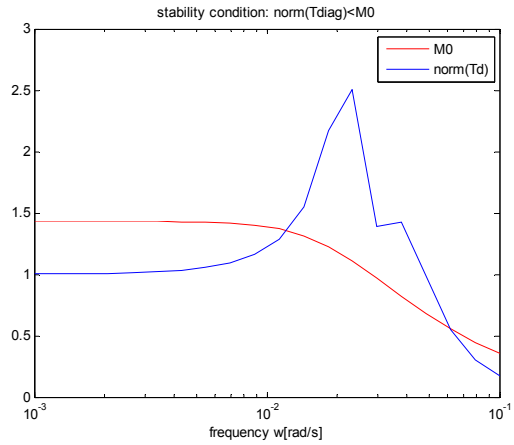


Fig.4a Decentralized control design: stability condition (*not satisfied*)

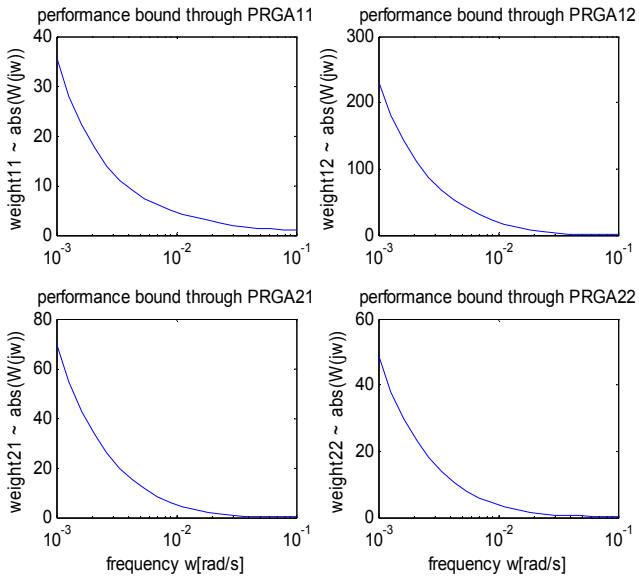


Fig.4b Decentralized control design: performance bound indicates satisfactory result

The third case shows results after the redesign of loop controllers' parameters, note that the respective performance bounds – Fig.8b are worse than in previous case – Fig.6b.

The corresponding step responses in Fig.9 are slower than in previous case.

PI controller parameters were designed for individual loops, considering the overall stability condition (8): P1=0.2847, I1=0.0025; P2=0.5308, I2=0.0063.

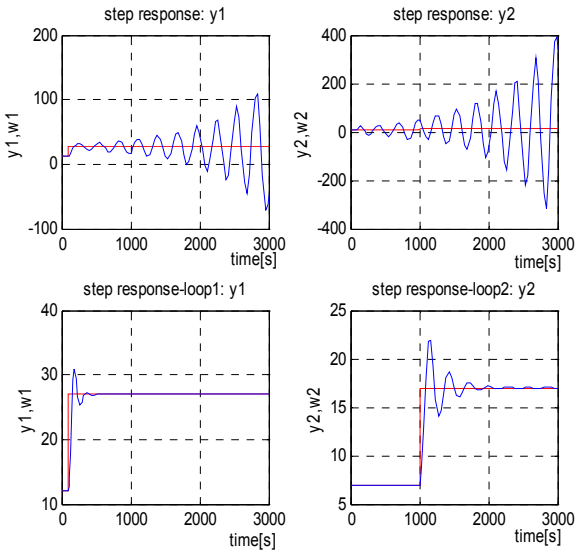


Fig.5 Step responses:
interconnected system – upper plots
individual loops – lower plots; big difference between individual loops and overall system

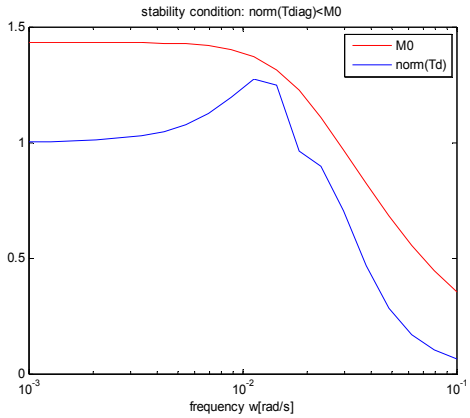


Fig.6a Decentralized control design: stability condition (satisfied)

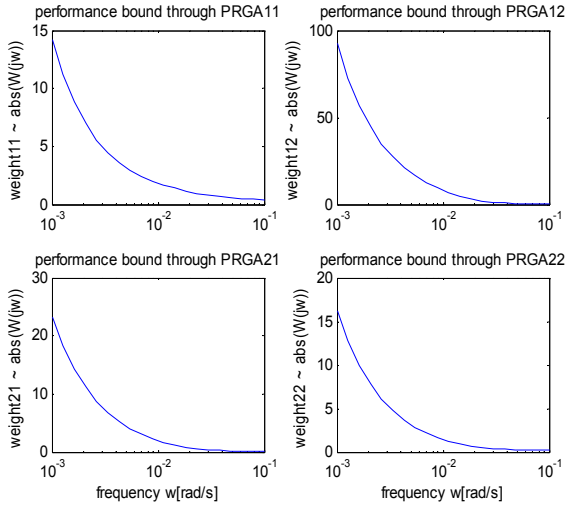


Fig.6b Decentralized control design: performance bound

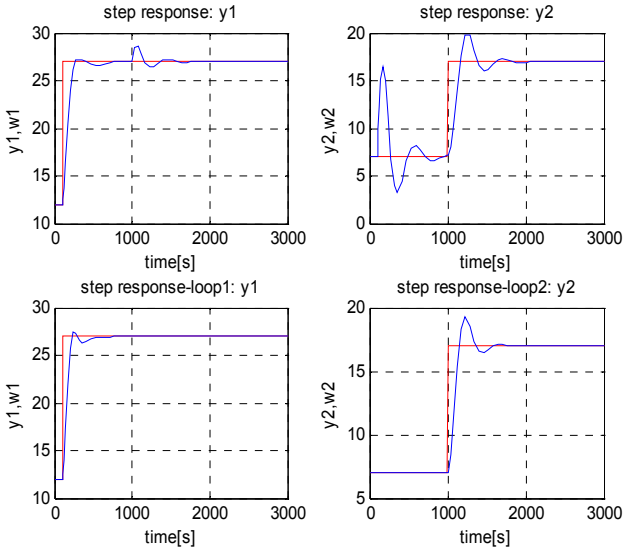


Fig.7 Step responses:
interconnected system – upper plots
individual loops – lower plots;
no substantial difference between individual loops and overall system

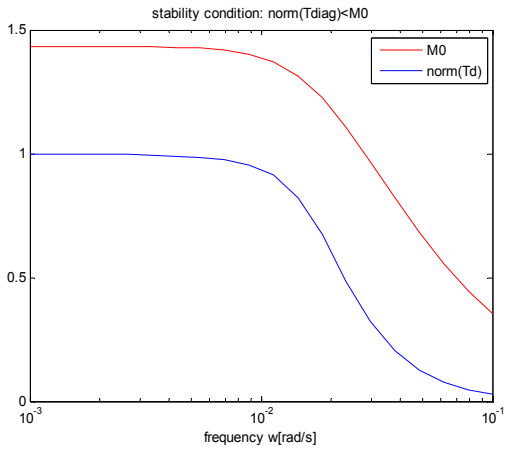


Fig.8a Decentralized control design: stability condition (satisfied)

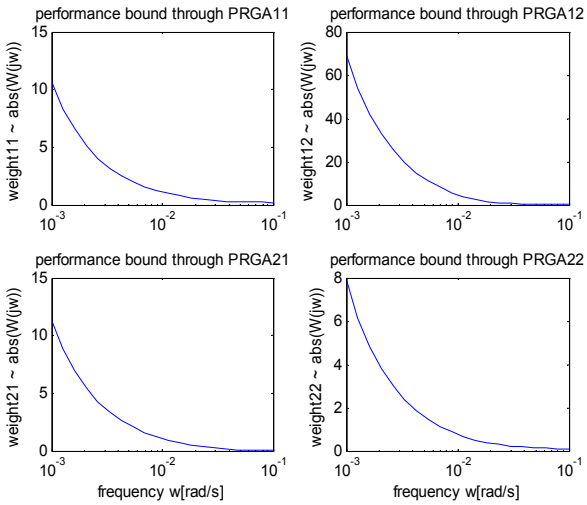


Fig.8b Decentralized control design: performance bound

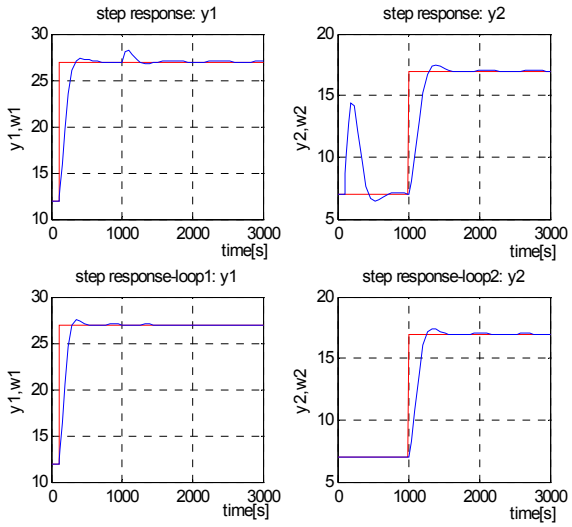


Fig.9 Step responses:
interconnected system – upper plots
individual loops – lower plots

Conclusion

The paper studies some basic aspects of decentralized control design concerning stability and performance. The decentralized control design strategy includes controller configuration (selection of appropriate pairing based on RGA), and individual control loops design based on overall stability condition and performance evaluation. This strategy is illustrated on example – case study of quadruple tank process. Two forms of overall stability conditions are used and briefly discussed, one of them (based on small gain theorem) formulated for stable systems which may have RHP zeros is less restrictive in higher frequencies, the other formulated for stable systems without RHP zeros is less restrictive in lower frequencies as shown in the example. Presented material provides simple illustration of stability versus performance relationship in decentralized control design and is believed to be useful in teaching complex systems control.

Acknowledgment

The work has been supported by Slovak Scientific Grant Agency, Grant N 1/0544/09. This support is very gratefully acknowledged.

REFERENCES

[1] HOVD, M., SKOGESTAD, S.: Simple Frequency-dependent Tools for Control System Analysis, Structure Selection and Design, *Automatica*, **28**, no.5, 1992, pp. 989-996

[2] JOHANSSON, K.H.: The Quadruple-Tank Process: A Multivariable Laboratory Process with an Adjustable Zero, *IEEE Transactions on Control Systems Technology*, **8**, (No. 3), 2000, pp.456-465

[3] ROSINOVÁ, D., MARKECH, M.: Robust Control of Quadruple – Tank process, *ICIC Express Letters*, 2008, **2**, (No. 3), 2008, pp.231-238

[4] ROSINOVÁ, D., VESELÝ, V.: Decentralized PID controller design for uncertain linear system, In: 11th IFAC/IFORS/IMACS/IFIP Symposium on Large Scale Complex Systems *Theory and Applications*, Gdansk, Poland, July, 2007 (CD)

[5] SCHMIDT, H.: *Model based design of decentralized control configurations*. Licentiate Thesis, Royal Institute of Technology, Stockholm, Sweden, 2002

[6] SKOGESTAD, S., POSTLETHWAITE, I.: *Multivariable feedback control*. John Wiley and Sons, New York, 2005

[7] VESELÝ, V., HARSÁNYI, L.: *Robust Control of Dynamic Systems*. STU Bratislava , (in Slovak), 2008

doc. Ing. Danica Rosinová, PhD.

Slovak University of Technology
Faculty of Electrical Engineering and Information Technology/ Institute of Control and Industrial Informatics
Ilkovičova 3
812 19 Bratislava
Tel.: +421 2 60291563
E-mail danica.rosinova@stuba.sk

Tuning decentralized controllers for robustness and performance

Alena Kozáková, Vojtech Veselý

Abstract
The paper presents a modification of the decentralized controller design technique for continuous-time systems (named "Equivalent Subsystems Method", ESM) proposed in [5,8] and further developed towards securing robust stability and nominal performance [6,7]. The proposed design procedure combines the ESM with a subsequent detuning to fulfil the M-Δ structure robust stability conditions adapted for the decentralized control. Robust decentralized controllers designed for two real plants show practical applicability of the proposed design philosophy.

Keywords: decentralized control, detuning, nominal stability, robust stability, unstructured uncertainty

Introduction

Many industrial processes are naturally multi-input multi-output (MIMO) as they arise as interconnection of a finite number of physically existing subsystems. Due to the interactions, MIMO systems are more difficult to control compared with the SISO ones. Multivariable controllers are used if strong interactions within the plant are to be compensated for. Decentralized controllers remain popular in the industry when practical reasons make restrictions on controller structure necessary or reasonable. Compared with centralized full-controller systems the decentralized control (DC) structure brings about certain performance deterioration; however weighted against important benefits, e.g. hardware, operation and design simplicity, and reliability improvement Therefore, decentralized control (DC) design techniques remain popular among practitioners, in particular the frequency domain ones which provide insightful solutions and link to the classical control theory.

Since the 80's several practice-oriented robust control design techniques have evolved differentiating in the design of local SISO controllers the main approaches being simultaneous design, independent design e.g. [2,3] and sequential design e.g. [2]. The method proposed in this paper belongs to the independent design according to which local controllers are designed independently without considering interactions with other subsystems. Main advantages with this approach are failure tolerance and direct local designs, the main limitation is that information about controllers in other loops is not exploited; therefore obtained stability and performance conditions are only sufficient and thus conservative. The paper deals with a further improvement of the robust decentralized controllers design technique for continuous-time uncertain systems (so-called "Equivalent Subsystems Method") first proposed as a DC design method for performance [5] and further adapted so as to simultaneously guarantee nominal performance and fulfilment of the *M-Δ* structure based robust stability conditions modified for the closed-loop under the decentralized controller [6,7,15,16]. This design technique considers the full transfer function matrix nominal system - unlike the existing robust DC approaches which take as nominal system just the diagonal part of the plant transfer matrix. The paper is organized as follows: theoretical background and problem formulation are given in Section 2, development of the robust decentralized controller design technique is presented in Section 3 and illustrated by two examples in Section 4. Conclusions are given at the end of the paper.

1. Theoretical Background and Problem Formulation

Consider a plant transfer function $G(s) \in R^{m\times m}$ and a diagonal controller $R(s) \in R^{m\times m}$ in a standard feedback configuration (Fig.1) where $w, u, y, e,$ are vectors of reference, control, output and control error, respectively, of compatible dimensions.

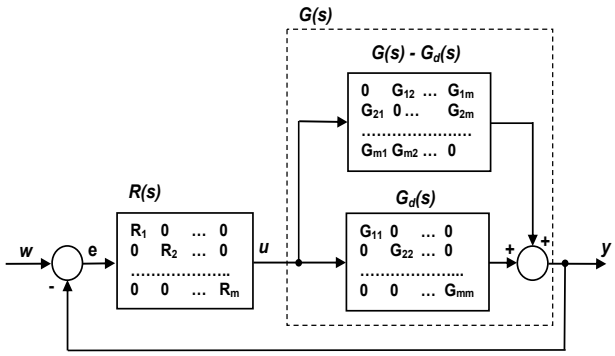


Fig. 1 Feedback loop under decentralized controller

Let the uncertain plant model be given as a set of *N* transfer function matrices in *N* different operating points, hence

$$G^k(s) = \{G_{ij}^k(s)\}_{m\times m},\ k = 1,2,\ldots,N \qquad (1)$$

with $G_{ij}^k(s) = \dfrac{y_i^k(s)}{u_j^k(s)},\quad i,j = 1,2,\ldots,m$

where $y_i^k(s)$ is the *i-th* output and $u_j^k(s)$ is the *j-th* input of the plant in the *k-th* experiment.

In this paper, unstructured uncertainty associated with the system model (1) will be described using additive (a), multiplicative input (i) and multiplicative output (o) forms, generating the related families of plants Π_i, $i = a,i,o$:

$$\Pi_a: \quad G(s) = G_N(s) + \ell_a(s)\Delta(s) \tag{2}$$

$$\ell_a(s) = \max_k \sigma_M[G^k(s) - G_N(s)]$$

$$\Pi_i: \quad G(s) = G_N(s)[I + \ell_i(s)\Delta(s)] \tag{3}$$

$$\ell_i(s) = \max_k \sigma_M\{(G^k(s))^{-1}[G^{-1}(s) - G_N(s)]\}$$

$$\Pi_o: \quad G(s) = [I + \ell_o(s)\Delta(s)]G_N(s) \tag{4}$$

$$\ell_o(s) = \max_k \sigma_M\{[G_N(s) - G^k(s)](G^k(s))^{-1}\}$$

where $G_N(s)$ denotes the nominal model, $\sigma_M(\cdot)$ is the maximum singular value of *(.)* and $\Delta(s) \in R^{m\times m}$ is uncertainty matrix such that $\sigma_M(\Delta) \le 1$.

Standard feedback configuration comprising the uncertain system can be transformed into the $M-\Delta$ structure; for individual uncertainty types the corresponding matrices M_k, $k = a, i, o$ are as follows [2]:

$$M_a = -\ell_a(s)[I + R(s)G_N(s)]^{-1}R(s)$$

$$M_i = -\ell_i(s)[I + R(s)G_N(s)]^{-1}R(s)G_N(s) \tag{5}$$

$$M_o = -\ell_o(s)G_N(s)R(s)[I + G_N(s)R(s)]^{-1}$$

Robust stability conditions in terms of the $M-\Delta$ structure are given in the following theorem.

Theorem 1 (Robust stability for unstructured perturbations)

Assume that the nominal system $M_k(s)$, $k = a,i,o$ is stable and the perturbation $\Delta(s)$ is stable. Then the $M-\Delta$ system is stable for all perturbations satisfying $\sigma_M[\Delta(j\omega)] \le 1$ if and only if

$$\sigma_M[M_k(s)] < 1, \quad \forall s \tag{6}$$

Nominal closed-loop stability of a MIMO system can be examined using the generalized Nyquist stability theorem.

Theorem 2 (Generalized Nyquist Stability Theorem)

The feedback system in Fig. 1 is stable if and only if

1. $\det F(s) \ne 0 \qquad \forall s \in D$

2a. $N[0, \det F(s)] = n_q$ (7)

where $Q(s) = G_N(s)R(s)$ is the open-loop matrix, n_q is the number of its right half-plane poles, $\det F(s) = \det[I + Q(s)]$ is the closed-loop characteristic polynomial, $N[0, \det F(s)]$ is the number of anticlockwise encirclements of the point $(0, j0)$ by the Nyquist plot of $\det F(s)$. Denote $q_i(s)$, $i = 1,2,...,m$ the set of characteristic loci (CL) of *Q(s)* in the complex plane [9].

1.1 Problem Formulation

Consider an uncertain MIMO system with *m* subsystems (1). A robust decentralized controller

$$R(s) = diag\{R_i(s)\}_{i=1,...,m} \tag{8}$$
$$\det R(s) \ne 0 \quad \forall s$$

is to be designed with $R_i(s)$ being transfer function of the *i-th* local controller. The designed controller has to guarantee stability and an acceptable performance of the controlled plant within the entire plant operating range described using either of the perturbed models (2), (3) or (4).

2. Development of the Robust DC Design Method

Consider the nominal model $G_N(s) \in R^{m\times m}$ split into the diagonal and the off-diagonal parts describing respectively models of decoupled nominal subsystems and nominal interactions $G_m(s)$

$$G_N(s) = G_d(s) + G_m(s) \tag{9}$$

where $G_d(s) = diag\{G_i(s)\}_{i=1,...,m}$ and

$\det G_d(s) \ne 0 \quad \forall s \in D$.

Factorize the nominal closed-loop characteristic polynomial $\det F(s)$ in terms of the split nominal system (Kozáková and Veselý, 2003; 2007)

$$\begin{aligned} \det F(s) &= \det\{I + [(G_d(s) + G_m(s)]R(s)\} = \\ &= \det[R^{-1}(s) + G_d(s) + G_m(s)]\det R(s) = \\ &= \det F_1(s)\det R(s) \end{aligned} \tag{10}$$

where $F_1(s) = R^{-1}(s) + G_d(s) + G_m(s)$ (11)

In view of (11), *Theorem 2* reads as follows:

Corollary 1

A closed-loop comprising the system (9) and the decentralized controller (8) is stable if and only if

1. $\det F_1(s) \ne 0 \qquad \forall s \in D$
2. $N[0, \det F_1(s)] + N[0, \det R(s)] = n_q$ (12)

In $F_1(s)$ in (12), $[R^{-1}(s) + G_d(s)]$ is a diagonal matrix related to subsystems. Denote

$$R^{-1}(s) + G_d(s) = P(s) \tag{13}$$

where $P(s) = diag\{p_i(s)\}_{m\times m}$. From (13) results

$$I + R(s)[G_d(s) - P(s)] = 0 \tag{14}$$

For individual subsystems, (14) breaks down to

$$1 + R_i(s)G_i^{eq}(s) = 0 \qquad i = 1,2,...,m \tag{15}$$

where $G_i^{eq}(s) = G_i(s) - p_i(s) \qquad i = 1,2,...,m$ (16)

is transfer function of the *i-th equivalent subsystem* [5]. Substituting (13) into (11) we obtain

$$\det F_1(s) = \det[P(s) + G_m(s)] \tag{17}$$

Using (17) it is possible to formulate stability conditions for the closed-loop system under a decentralized controller [5,8].

Corollary 2 (Nominal stability under DC)

A closed-loop comprising the nominal system (9) and a decentralized controller (8) is stable if there exists a stable matrix $P(s) = diag\{p_i(s)\}_{m\times m}$ *such that each equivalent subsystem (16) can be stabilized by its related local controller* $R_i(s)$*, i.e. each equivalent closed-loop characteristic polynomial*

$$CLCP_i^{eq} = 1 + R_i(s)G_i^{eq}(s) \quad i = 1,2,...,m$$

has stable roots and the following conditions are met

1. $det[P(s) + G_m(s)] \neq 0$ (18)

2. $N\{0,\ det[P(s) + G_m(s)]\} = n_q$ (19)

Corollary 3 (Robust stability under DC)

The $M-\Delta$ *structure is stable if there exists such* $P(s) = diag\{p_i(s)\}_{m\times m}$ *that conditions (18) and (19) are met and for either of the uncertainty forms (2), (3), (4) holds the corresponding inequality:*

Additive uncertainty

$$\sigma_M\{[P(j\omega) + G_m(j\omega)]^{-1}\} < \frac{1}{\ell_a(\omega)} \quad (20)$$

Multiplicative input uncertainty

$$\sigma_M\{[P(j\omega) + G_m(j\omega)]^{-1} G_N(s)\} < \frac{1}{\ell_i(\omega)} \quad (21)$$

Multiplicative output uncertainty

$$\sigma_M\{G_N(j\omega)[P(j\omega) + G_m(j\omega)]^{-1}\} < \frac{1}{\ell_o(\omega)} \quad (22)$$

Hence, the problem to be solved in the robust decentralized controller design reduces to finding an appropriate $P(s) = diag\{p_i(s)\}_{m\times m}$ that fulfils both Corollaries 2 and 3. Applying this approach allows to consider the full mean parameter value model as the nominal system.

Thus far, following methods of selecting *P(s)* have been proposed:

1. *Choosing* $P(s) = p(s)I$ *with identical entries in the diagonal*. If $p(s) = -g_\ell(s-\alpha)$ where $g_\ell(s)$ can be any fixed of the *m* characteristic functions of $[-G_m(s)]$ and $\alpha \geq 0$ is the specified feasible degree of stability, it is possible to achieve the degree of stability α for the full closed-loop system [5,8]. Moreover, if $p(s) = -g_\ell(s-\alpha)$ satisfies the $M-\Delta$ stability conditions for systems under a decentralized controller [6] then both the specified nominal performance and robust stability are guaranteed. To stabilize equivalent subsystems any graphical SISO frequency domain design technique can be applied independently. (e.g. Bode plots, Neymark D-partition method).

Application and main results of this design approach are illustrated in *Example 1.*

2. *Choosing* $P(s) = diag\{p_i(s)\}_{i=1,...,m}$ *with different diagonal entries*. In [15] a heuristic method has been proposed to find coefficients of stable $p_i(s)$ such that $structure[p_i(s)] = structure[G_i(s)],\ i = 1,...,m$.

For a decentralized fixed structure (PI, PID) controller a design procedure has been developed yielding improved damping of $G_i(s)$. General suggestions for choosing *P(s)* are given in [7]: for both $P^{-1}(s)$ and $G_m(s)$ stable, the necessary and sufficient closed-loop stability condition is $\sigma_M[G_m(s)] < \sigma_m[P(s)]$.

Moreover, to guarantee robust stability, conditions (20), (21) or (22) have to be met in all cases.

In this paper an innovative approach for generating $P(s)$ with different diagonal entries is proposed. Its underlying idea consists in generating $P(s)$ using (13) and to guarantee fulfilment of conditions (18), (19), (20), (21) and (22) formulated in the Corollaries 2 a 3 by local controller detuning.

The resulting robust controller design procedure involves two main stages: the initial design and the possible redesign. Individual design steps are as follows:

Initial design

1. Choice of the nominal model $G_N(s) = G_d(s) + G_m(s)$, computation of and plotting uncertainty bounds $\ell_k, k = a,i,o$ according to (2), (3) or (4).

2. Design of local controllers $R_i(s),\ i = 1,...,m$ for isolated subsystems using any standard frequency domain design method [e.g. 1,12,17] and setting up the resulting diagonal controller in form (for $\delta = 1$)

$$R(s) = \begin{bmatrix} \frac{R_1(s)}{\delta_1} & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & \frac{R_m(s)}{\delta_m} \end{bmatrix}$$

3. Generating $P(s) = R^{-1}(s) + G_d(s)$ according to (13).

4. Verification of the nominal stability condition (18) and (19) (*Corollary 2*)

5. Verification of robust stability (RS) conditions (20), (21), (22).

If any of the RS conditions is satisfied, the designed controller guarantees nominal stability and performance as well as closed-loop stability in the whole operating range of the plant specified by the *N* transfer functions matrices. The design procedure stops.

Redesign

If robust stability conditions fail to be satisfied, the design procedure is to be repeated either for relaxed nominal performance requirements or the original controller $R(s) = diag\{\frac{R_i(s)}{\delta_i}\}_{m\times m}$ is to be detuned using the detuning coefficients $\delta_i, i = 1,...,m$ so as to satisfy robust stability conditions in the tightest possible way.

According to the detuning procedure proposed in [11] that $\delta_k,\ k \in \{1,\ldots,m\}$, is changed (usually increased) which

contributes the most to the fulfillment of the RS condition. Selection of such δ_k, $k \in \{1,\ldots,m\}$ is carried out in m steps in such a way that in the *i-th* step $i \in \{1,\ldots,m\}$ just one particular δ_i is increased while the other remain unchanged and the RS condition is verified. The finally selected and changed $\delta_k, k \in \{1,\ldots,m\}$ is the one with the most significant contribution to the fulfillment of the RS condition. It specifies and is applied to the local controller to modify its parameters according to $\frac{R_k(s)}{\delta_k}$.

The design procedure is illustrated in the *Example 2*.

3. EXAMPLES

Example 1
(P(s) with identical entries in the diagonal)

Power system stabilizers (PSS) are used to enhance power system damping. In (Kozáková, 2004), the DC design methodology proposed in (Kozáková and Veselý, 2003) has been applied to design PSS with the fixed structure transfer function

$$PSS_i(s) = \frac{k_i s}{T_i s + 1}, \quad i = 1,2$$

for two generating units of the Slovak Power System. The linearized mathematical model of the MIMO system has been obtained from experiments on the model of the Slovak Power System.

$$G(s) = \begin{pmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{pmatrix}$$

where

$$G_{11}(s) = \frac{-4.4s^3 + 147.9s^2 + 731.7s - 2243}{s^4 + 8.3s^3 + 162.8s^2 + 509.5s + 5283}$$

$$G_{12}(s) = \frac{0.1278s^3 - 1.964s^2 + 15.21s - 28.61}{s^4 + 13.24s^3 + 65.76s^2 + 661.5s + 3.257}$$

$$G_{21}(s) = \frac{0.0097s^3 - 5.883s^2 + 18.77s - 165.8}{s^4 + 27.5s^3 + 52s^2 + 1191s + 5.4}$$

$$G_{22}(s) = \frac{-1.102s^3 + 134.7s^2 + 51.61s - 383.5}{s^4 + 7.599s^3 + 58.94s^2 + 335.6s + 58.04}$$

The PSS have been designed to reduce by 8.5dB the resonance peaks of equivalent subsystems. Local PSS´s for individual units have been designed using Bode plots of equivalent subsystems generated by $p_1(s)$. Using the standard design approach, parameters for both PSS have been chosen as follows

$$PSS_i(s) = \frac{0.0562s}{0.1s + 1} \qquad i = 1,2$$

Bode plots of compensated equivalent subsystems depicted in Fig. 2 with thick lines prove the required resonance peak reduction. Experimental studies on a physical model of the Slovak Power System have proved effectiveness of the designed PSS´s in improving the power system damping in the required frequency range. One illustrative result – response of the active power deviation in both generating units with implemented PSS to a three phase to ground fault during 0.2s at the middle of the transmission line between both units is shown in Fig. 3b

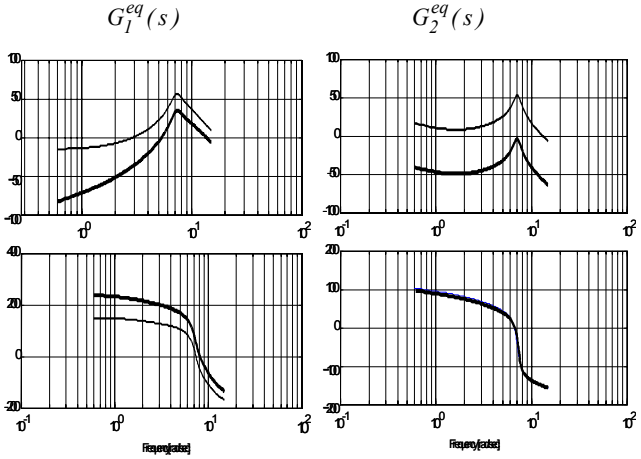


Fig. 2 Bode plots of equivalent subsystems (thin lines – uncompensated subsystems; thick lines – subsystems with PSS compensation)

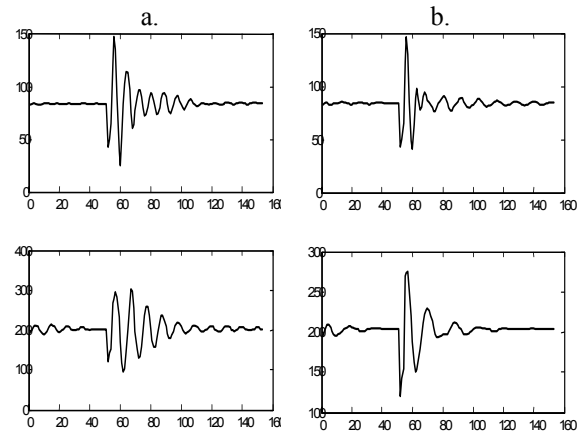


Fig. 3 Time responses of the active power deviation in Unit 1 (upper plots) and Unit 2 (lower plots) to a three phase to ground fault during 0.2s: a. uncompensated subsystems; b. subsystems with PSS compensation.

Example 2
(P(s) with different diagonal entries)

Consider the 3x3 transfer function model for a pilot scale binary distillation column used to separate ethanol and water (Hovd and Skogestad, 1994; Ogunnaike and Ray, 1994). A strong one-way interaction is evident from the large off-diagonal elements in the 3rd row.

$$G(s) = \begin{pmatrix} \frac{0.66e^{-2.6s}}{6.7s+1} & \frac{-0.61e^{-3.5s}}{8.64s+1} & \frac{-0.0049e^{-s}}{9.06s+1} \\ \frac{1.11e^{-6.5s}}{3.25s+1} & \frac{-2.36e^{-3s}}{5s+1} & \frac{-0.012e^{-1.2s}}{7.09s+1} \\ \frac{-33.68e^{-9.2s}}{8.15s+1} & \frac{46.2e^{-9.4s}}{10.9s+1} & \frac{0.87(11.61s+1)e^{-s}}{(3.89s+1)(18.8s+1)} \end{pmatrix}$$

Based on the RGA matrix

$$\Lambda = \begin{pmatrix} 1.95 & -0.65 & -0.3 \\ -0.66 & 1.88 & -0.22 \\ -0.29 & -0.23 & 1.52 \end{pmatrix}$$

the full system has been partitioned into the diagonal part (subsystems) and interactions (9). A low value of Niederlinski index (*N = 0.3752*) and a high value of the VA index - sufficient condition for correct input-output pairing (Kozáková, 1998), VA = 54.679 (instead of preferred VA<1) indicate difficulties in structural controllability of the plant.

For the design purpose, time delays have been replaced with the 6th order Padé approximants. Uncertainty bounds (2), (3) and (4) have been computed for $\pm 15\%$ changes in parameter values of all entries of $G(s) = G_N(s)$.

PID controller transfer functions for decoupled subsystems have been chosen

$$R_i(s) = \frac{r_d s^2 + r_0 s + r_{-1}}{ks}, \; i = 1,2,...,m$$

where *k* is the common factor applied simultaneously in all loops in the redesign step to fulfil the robust stability conditions (20), (21), (22).

The $\left|\frac{1}{\ell_k(\omega)}\right|, k = a,i,o$ plots are depicted in Fig. 4, local controllers have been designed for the worst case – i.e. the multiplicative output uncertainty.

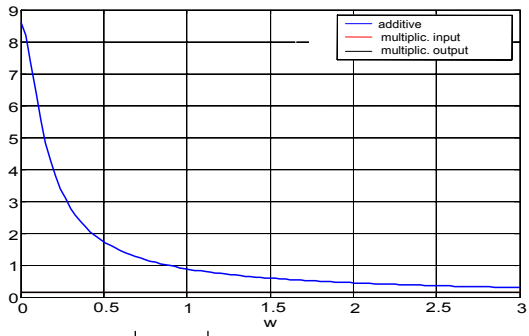


Fig. 4 The $\left|\frac{1}{\ell_k(\omega)}\right|$ - versus - ω plots for $k = a,i,o$

Final values of local controller parameters have been chosen as follows

$$R_1(s) = \frac{0.0637s^2 + 0.3186s + 0.0574}{s}$$

$$R_2(s) = \frac{-(0.1593s^2 + 0.1912s + 0.0478)}{s}$$

$$R_3(s) = \frac{0.1593s^2 + 2.5490s + 0.7966}{s}$$

Fig. 5 shows that with the chosen decentralized controller, the robust stability conditions are verified.

Closed-loop step responses of the nominal system for the reference signal [1 0 0]T are in Fig. 6, Fig. 7 shows simulation results for unit steps applied in all subsystems at different step times (12s, 5s, 20s for the 1st, 2nd and 3rd subsystem, respectively).

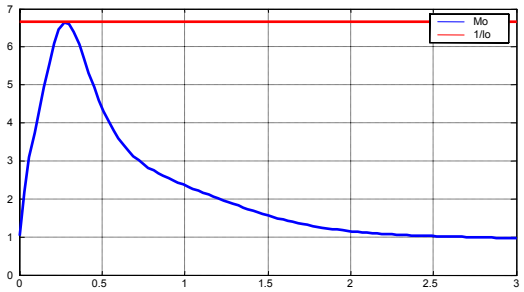


Fig. 5 Verification of the robust stability condition under the designed DC and multiplicative output uncertainty: $\sigma_M[M_0(s)] < \frac{1}{|\ell_0|}$

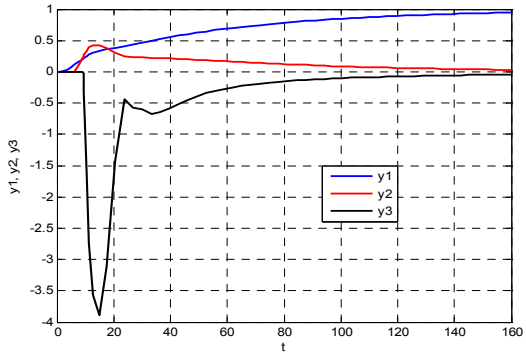


Fig. 6 Closed-loop step responses under the designed DC for the reference unit step in the first subsystem

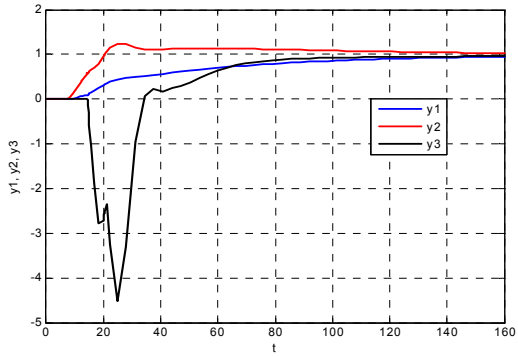


Fig. 7 Closed-loop step responses under the designed DC for the reference unit steps applied in all subsystems at different step times (12s, 5s, 20s)

Conclusion

In this paper an improvement to the existing robust decentralized controller design technique for continuous-time uncertain systems has been proposed. Applying the generalized Nyquist stability criterion and the M-Δ structure robust stability conditions adapted for the decentralized control structure, the problem to be solved in the robust decentralized controller design reduces to finding an appropriate $P(s) = diag\{p_i(s)\}_{m\times m}$ guaranteeing both nominal and robust closed-loop stability of the full system under a decentralized controller. Moreover, this approach allows considering the full mean parameter value model as the nominal system. An innovation in choosing $P(s)$ with different

diagonal entries has been proposed, resulting in a simple-to-use insightful graphical design procedure that involves two main stages: in the initial design stage, local controllers are designed for isolated subsystems, the matrix *P(s)* is generated and the nominal and robust stability conditions are verified. If they fail to be satisfied, in the redesign stage, the local controllers transfer functions are modified so as to satisfy robust stability conditions in the tightest possible way. The proposed practice-oriented approach has been applied in the design of robust decentralized PID controllers for real plant models (a 2x2 power system with two generating units and a 3x3 laboratory binary distillation column) show practical applicability of the proposed design philosophy.

Acknowledgment

This work has been supported by the Slovak Grant Agency under Grant. No. 1/0544/09.

References

[1] FILASOVÁ, A., KROKAVEC, D. (1999). Pair-wise decentralized control of large-scale systems. *Journal of Electrical Engineering*, 50, No. 3-4, 1-10. (ISSN 1335-3632)

[2] HOVD, M., SKOGESTAD, S. (1994). Sequential design of decentralized controllers. *Automatica*, 30, 1601-1607.

[3] KOZÁKOVÁ, A. (1998). Robust decentralized control of complex systems in the frequency domain. *2nd IFAC Workshop New Trends in Design of Control Systems*, Elsevier, Kidlington, UK.

[4] KOZÁKOVÁ, A. (2004). PSS design for a multivariable system in the frequency domain. In 6th Int. Conference Control of Power System, Štrbské Pleso, Slovak Republic.

[5] KOZÁKOVÁ, A., VESELÝ, V. (2003). Independent design of decentralized controllers for specified closed-loop performance. *European Control Conference ECC´03*, Cambridge, UK

[6] KOZÁKOVÁ, A., VESELÝ, V. (2005). A frequency domain design technique for robust decentralized controllers. *16th IFAC World Congress*, Prague, Czech Republic.

[7] KOZÁKOVÁ, A., VESELÝ, V. (2006). Robust decentralized controller design: independent design. *IFAC Workshop ALSIS´06*, Helsinki, Finland.

[8] KOZÁKOVÁ, A., VESELÝ, V. (2007). Independent design of decentralized controllers in the frequency domain. *Periodica Polytechnica Electrical Engineering*, 51/1-2, 1-9. ISSN 0324-6000.

[9] MACFARLANE, A.G.J., POSTLETHWAITE, I. (1977). The generalized Nyquist stability criterion and multivariable root loci. *Int. J. Control*, 25, 81-127.

[10] OGUNNAIKE, B.A., RAY, W.H. (1994). *Process dynamics, modelling and control*. Oxford University Press, New York – Oxford.

[11] OSUSKÝ, J. VESELÝ, V. (2007). Robust Decentralized Controller Design in Frequency Domain. *16th Int. Conference Process Control PC´07*. Štrbské Pleso, Slovak Republic.

[12] ROSINOVÁ D., MARKECH, M. (2008). Robust control of quadruple-tank process. *ICIC Express Letters /An International Journal of Research and Surveys*, 2 (3), 231-238. ISSN 1881-803X

[13] SKOGESTAD, S., MORARI, M. (1989). Robust performance of decentralized control systems by independent designs. *Automatica*, 25, 119-125.

[14] SKOGESTAD, S., POSTLETHWAITE, I. (1996). *Multivariable fedback control: analysis and design*. (3rd edn.) John Wiley & Sons Ltd., Chichester New York Brisbane Toronto Singapore.

[15] VESELÝ, V., KOZÁKOVÁ, A. (2005). Independent design of robust decentralized controllers. *1st Int. Workshop Advanced Control Circuits and Systems (ACCS´05)*, Cairo, Egypt.

[16] VESELÝ, V., KOZÁKOVÁ, A. (2005). A power system application of a robust decentralized controller design methodology. *Archives of Control Sciences*, 15, No. 2, 137-148. ISSN 000

[17] VÍTEČKOVÁ, M., JARACZ, K. (2008). Analityczna metoda strojenia regulatorów cyfrowych i analogowych dla inercyjnych obiektów pierwszego rzędu z opóźnieniem. *Annales Academiae Paedagogicae Cracoviensis. Studia Technica II.* 2008. Kraków, Wydawnictwo Naukowe Akademii Pedagogicznej, pp. 176-185. ISSN 1896-1223

Prof. Ing. Vojtech Veselý, DrSc.
Doc. Ing. Alena Kozáková, PhD.

Slovak University of Technology
Faculty of Electrical Engineering and Industrial Informatics
Institute of Control and Industrial Informatics
Ilkovičova 3
812 19 Bratislava
Tel.: ++ 421 2 602 91 563
E-mail: vojtech.vesely@stuba.sk
alena.kozakova@stuba.sk

Predictive Control of Nonlinear Process

Jana Paulusová, Mária Dúbravská

Abstract
In this paper two predictive control approaches are addressed, proposed and tested. The first method described in this paper is Generalized Predictive Control (GPC). The second one is Dynamic Matrix Control (DMC). The proposed algorithms are tested in model based predictive control of the concentration control in the chemical reactor, manipulating its flow rate.

Keywords: DMC, GPC, model predictive control

Introduction

Concept of model based predictive control (MBPC) has been heralded as one of the most significant control developments in recent ten years. Wide range of choice of model structures, prediction horizon, and optimization criteria allows the designer to easily tailor MBPC to its application in industry.

The main idea of Model Predictive Control (MPC) is the prediction of the output signal at each sampling instant. The prediction is implicit or explicit depending on the model of the process to be controlled. In the next step a control is selected to bring the predicted process output signal back to the reference signal by minimizing the area between the reference and the output signal.

Within basic methods, which are essentially referred as predictive control, DMC and GPC with the added capability of handling nonlinear systems are included.

1. MPBC Formulation

MBPC is a general methodology for solving control problems in the time domain. It is based on three main concepts:

- Explicit use of a model to predict the process output.
- Computation of a sequence of future control actions by minimizing a given objective function.
- The use of the receding horizon strategy: only the first control action in the sequence is applied, and the horizons are moved one sample period towards the future, optimisation is repeated.

1.1 Dynamic Matrix Control (DMC)

One of the first proposed MBPC methods, and still commercially the most successful one, is DMC. Cutler introduced this method in 1980.

MPC is an optimization based control methodology that explicitly utilizes a dynamic mathematical model of a process to obtain a control signal by minimizing an objective function. The model must describe the system well. The future process outputs $y(k+i)$ for $i=1,\ldots, p$, are predicted over the prediction horizon (p) using a model of the process.

These values depend on the current process state, and on the future control signals $u(k+i)$ for $i=0,\ldots,m-1$, over the control horizon (m), where $m<p$. The control variable is manipulated only within the control horizon and remains constant afterwards, $u(k+i)=u(k+m-1)$ for $i=m,\ldots, p-1$. The basic principle of MBPC is shown in Fig. 1.

Process interactions and deadtimes can be intrinsically handled with model predictive control schemes such as DMC. The block-scheme of MPC is shown in Fig. 1 and the principle of DMC in Fig. 2.

The sequence of future control signals is computed by optimizing a given (cost) function. Often, the system needs to follow a certain reference trajectory defined through set points. In most cases, the difference between system outputs and reference trajectory is used by combination with a cost function on the control effort.

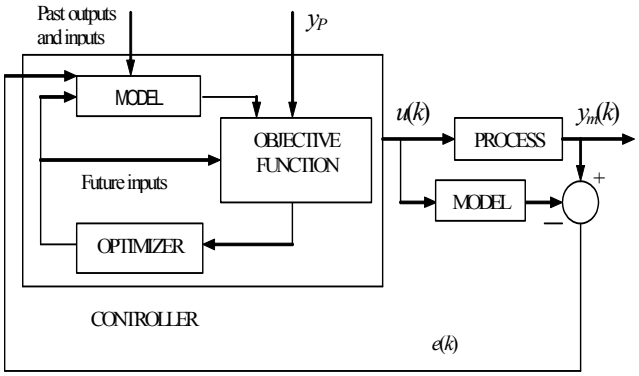


Fig.1 Block scheme of model predictive control (MPC)

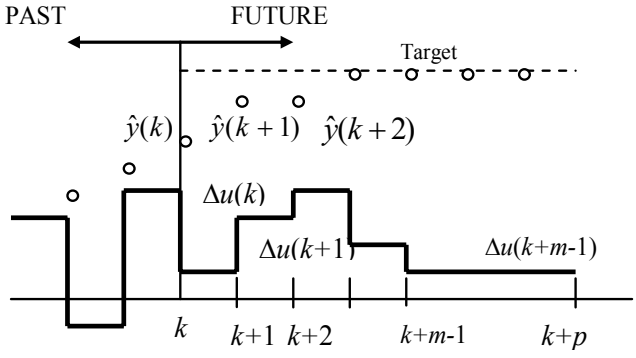


Fig.2 The principle of DMC

A general objective function is the following quadratic form

$$J = \sum_{i=1}^{p}[y_p(k+1\,|\,k) - \hat{y}(k+i\,|\,k)]^2\,\Gamma_y + \sum_{i=1}^{m}\Delta u(k+i-1\,|\,k)^2\,\Gamma_u \tag{1}$$

where

y_p - desired set point,

Γ_u , Γ_y - weight matrixes, where $\Gamma_u=\gamma_u I_u$ and $\Gamma_y=\gamma_y I_y$ (I_u is unit matrix with dimension mxm, I_y is unit matrix with dimension pxp and γ_u, γ_y are the weight parameters)

$\Delta u(k\text{-}i)=u(k\text{-}i)\text{-}u(k\text{-}i\text{-}1)$ - the change in manipulation variable,

p - the length of the prediction horizon,

m - the length of the control horizon,

$\hat{y}(k)$ - the process output, at sample instant is given as

$$\hat{y}(k) = \sum_{i\,=\,1}^{\infty} g_i \Delta u(k-i) \tag{2}$$

where

g_i - the step response coefficients.

The model employed is a step response of the plant. The model predictions along the prediction horizon p are

$$\hat{y}(k+j\,|\,k) = \sum_{i=1}^{\infty} g_i \Delta u(k+j-i) + d(k+j\,|\,k) \tag{3}$$

Disturbances are considered to be constant between sample instants

$$d(k+j\,|\,k) = y_m(k\,|\,k) - \sum_{i=1}^{\infty} g_i \Delta u(k+j-i) \tag{4}$$

where

$y_m(k|k)$ - the measured value of the process output at time k.

$$\hat{y}(k+j\,|\,k) = \sum_{i=1}^{k} g_i \Delta u(k+j-i) + f(k+j\,|\,k) \tag{5}$$

where

$$f(k+j\,|\,k) = y_m(k\,|\,k) + \sum_{i=1}^{N}(g_{k+1} - g_i)\Delta u(k-i) \tag{6}$$

The prediction of the process output along the length of the prediction horizon, can be written compactly using matrix notation

$$y_p(k) = G\Delta u(k) + f(k) \tag{7}$$

where

G - dynamic matrix

$$G = \begin{bmatrix} g_1 & 0 & \cdots & 0 \\ g_2 & g_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_m & g_{m-1} & \cdots & g_1 \\ \vdots & \vdots & \ddots & \vdots \\ g_p & g_{p-1} & \cdots & g_{p-m+1} \end{bmatrix} \tag{8}$$

By minimizing objective function the optimal solution is then given in matrix form

$$\Delta u(k) = (G^T\Gamma_y G + \Gamma_u)^{-1} G^T \Gamma_y (y_m(k) - f(k)) \tag{9}$$

1.2 Generalized Predictive Control (GPC)

The CARIMA model, representing the plant model in GPC design, is defined as:

$$A(z^{-1})y(k) = B(z^{-1})u(k-1) + C(z^{-1})\xi(k)/\Delta \tag{10}$$

where

Δ – is the differencing operator 1-z^{-1},

$\xi(k)$ – denotes white noise sequence,

A, B, C – are polynomials in the backward shift operator z^{-1}.

For simplicity here $C(z^{-1})$=1. To derive j-step ahead predictor of $y(k+j)$ based on (10) consider the identify [2]:

$$1 = E_j(z^{-1})A(z^{-1})\Delta + z^{-j}F_j(z^{-1}) \tag{11}$$

Then predicted process output is

$$\hat{y}(k+j\,|\,k) = G_j(z^{-1})\Delta u(k+j-1\,|\,k) + F_j(z^{-1})y(k) \tag{12}$$

where

$$G_j(z^{-1}) = E_j(z^{-1})B(z^{-1})$$

Then (12) can be written in vector form

$$\hat{y} = G\Delta u + f \tag{13}$$

where

$$\hat{y} = \begin{bmatrix} \hat{y}(k+1\,|\,k) \\ \hat{y}(k+2\,|\,k) \\ \cdots \\ \hat{y}(k+p\,|\,k) \end{bmatrix} \quad \Delta u = \begin{bmatrix} \Delta u(k) \\ \Delta u(k+1) \\ \cdots \\ \Delta u(k+m-1) \end{bmatrix}$$

$$G = \begin{bmatrix} g_0 & 0 & \cdots & 0 \\ g_1 & g_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ g_{p-1} & g_{p-2} & \cdots & g_0 \end{bmatrix} \quad f = \begin{bmatrix} f(k) \\ f(k+1) \\ \cdots \\ f(k+p) \end{bmatrix}$$

where

$$f(k+1) = [G_1(z^{-1}) - g_{10}]\Delta u(k) + F_1 y(k)$$

$$f(k+2) = z[G_1(z^{-1}) - z^{-1}g_{21} - g_{20}]\Delta u(k) + F_2 y(k) \quad \text{, etc.}$$

and

$$G_i(z^{-1}) = g_{i0} + g_{i1}z^{-1} + \ldots$$

Consider the minimization of cost function (1) the optimal solution in vector form is

$$\Delta u = (G^T\Gamma_y G + \Gamma_u)^{-1} G^T \Gamma_y (y_p - f) \tag{14}$$

2. Case Study and Simulation Results

The application considered involves an isothermal reactor in which the Van Vusse reaction kinetic scheme is carried out. In the following analysis, A is the educt, B the desired product, C and D are unwanted byproducts [6].

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$
$$2A \xrightarrow{k_3} D \tag{15}$$

From a design perspective the objective is to make k_2 and k_3 small in comparison to k_1 by appropriate choice of catalyst and reaction conditions. The concentration of B in the

product may be controlled by the manipulating the inlet flow rate and/or the reaction temperature.
The educt flow contains only cyclopentadiene in low concentration, C_{Af}. Assuming constant density and an ideal residence time distribution within the reactor, the mass balance equations for the relevant concentrations of cyclopentadiene and of the desired product cyclopentanol, C_A and C_B, are as follows:

$$\begin{aligned} \dot{C}_A &= -k_1 C_A - k_3 C_A^2 + \frac{F}{V}(C_{Af} - C_A) \\ \dot{C}_B &= k_1 C_A - k_2 C_B - \frac{F}{V} C_B \\ y &= C_B \end{aligned} \tag{16}$$

This example has been considered by a number of researchers as a benchmark problem for evaluating nonlinear process control algorithm.
By normalizing the process variables around the following operating point and substituting the values for the physical constants, the process model becomes:

$$\begin{aligned} \dot{x}_1(t) &= -50x_1(t) - 10x_1^2(t) + u(10 - x_1(t)) \\ \dot{x}_2(t) &= 50x_1(t) - 100x_2(t) + u(-x_2(t)) \\ y(t) &= x_2(t) \end{aligned} \tag{17}$$

where the deviation variable for the concentration of component A is denoted by x_1, the concentration of component B by x_2, and the inlet flow rate by u.

2.1 Simulation results

The comparison of time responses of controlled and referenced variables under DMC with GPC is shown in Fig. 3.

Parameters for DMC are p=10, m=5, γ_y=0.4, γ_u=0.4.
Parameters for GPC are p=3, m=3, γ_y=1, γ_u=0.1.

Quality of control is depended on the choice of parameters p, m, γ_y, and γ_u. For our selected parameters is quality of control under DMC better than GPC.

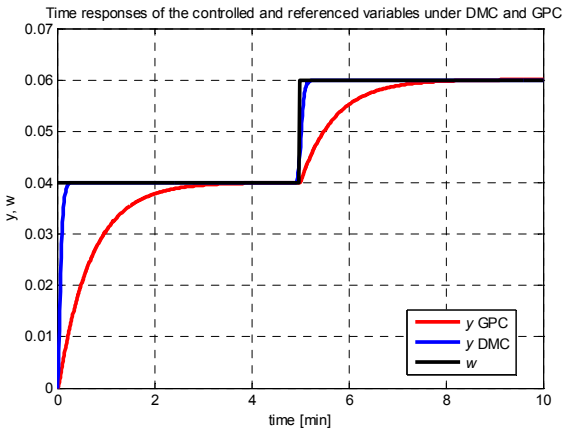


Fig.3 Time responses of the controlled and reference variables under DMC and GPC (w=y_p)

Conclusion

This paper has described two predictive approaches GPC and DMC algorithms. These methods are based on model, which is the predicted future behaviour of process. The advantages and disadvantages of both methods were presented here.

Acknowledgments

This paper has been supported by the Slovak Scientific Grant Agency, Grant No. 1/0544/09.

References

[1] BOZIN, A. S., AUSTIN P. C.: Dynamic matrix control of paper machine benchmark problem, Control Eng. Practice, 3 (No.10), pp. 1479-1482

[2] CLARCE, D. W., MOHTADI, C., TUFFS, P. S.: Generalized predictive control -Part I. The basic algorithm, Automatica, 23 (No.2), pp. 137-148, Printed in Great Britain

[3] PAULUSOVÁ, J., VESELÝ, V.: Predictive algorithms for control a DC Motor. International Slovak Society for Cybernetics and Informatics, Ždiar, Slovak Republic: 10-14 February 2008, pdf 35

[4] PAULUSOVÁ, J., KOZÁK, Š.: Nonlinear model-based predictive control. Control Systems Design 2003 CSD'03. Bratislava, Slovak Republic: 7-10 September, 2003, pp. 171-175

[5] PAULUSOVÁ, J., KOZÁK, Š.: Robust and Fuzzy Dynamic Matrix Control Algorithm. 5th International Carpathian Control Conference 2004. Zakopane, Poland: 25-28 May, pp. 145-152

[6] PAULUSOVÁ, J., DÚBRAVSKÁ, M.: Predictive Control of Nonlinear Process. International Slovak Society for Cybernetics and Informatics, Vyšná Boca, Slovak Republic: 10-13 February 2010, 20_Paulusova_Dubravska.pdf

Ing. Jana Paulusová
Ing. Mária Dúbravská

Slovak University of Technology
Faculty of Electrical Engineering
and Information Technology
Ilkovičova 3, 812 19 Bratislava, Slovak Republic
Tel.: +421 2 60291262 Fax: +421 2 65429521
e-mail: jana.paulusova@stuba.sk

Hybrid Predictive Control of Nonlinear Process

Jana Paulusová, Mária Dúbravská

Abstract
In this paper hybrid fuzzy model based predictive control (HFMBPC) is addressed, proposed and tested. The proposed hybrid fuzzy convolution model consists of a steady-state fuzzy model and a gain independent impulse response model. The proposed model is tested in model based predictive control of the concentration control in the chemical reactor, manipulating its flow rate. The paper deals with theoretical and practical methodology, offering approach for intelligent fuzzy robust control design and its successful application.

Keywords: fuzzy models, hybrid models, predictive control

Introduction

Predictive control has become popular over the past twenty years as a powerful tool in feedback control for solving many problems for which other control approaches have been proved to be ineffective. Predictive control is a control strategy that is based on the prediction of the plant output over the extended horizon in the future, which enables the controller to predict future changes of the measurement signal and to base control actions on the prediction.

The proposed HFMBPC has been received well in process industry.

1. The Hybrid Fuzzy Convolution

The output of the model can be formulated as [1]

$$y_m(k+1) = y_s + K(u_s, x_2, \ldots, x_n).\sum_{i=1}^{N} g_i(x_2, \ldots, x_n).(u(k-i+1)-u_s) \quad (1)$$

where $y_s + K(u_s, x_2, \ldots, x_n)$ is steady-state part, which is described by Takagi-Sugeno fuzzy model and $\sum_{i=1}^{N} g_i(x_2, \ldots, x_n).(u(k-i+1)-u_s)$ is dynamic part of model (the impulse response model). The gain independent impulse response model is $g_i(x_2,\ldots,x_n)$, the previous input values are $u(k\text{-}i\text{-}1)$ over N horizon, K is steady-state gain, u_s and y_s are steady-state input and output, $x_2, \ldots, x_n$ are other operating parameters having effects on the steady-state output.

The convolution is multiplied by steady-state gain

$$K = \frac{\partial f(u_s, x_2, \ldots, x_n)}{\partial u_s} \quad (2)$$

1.1 The Steady-State part of Hybrid Fuzzy- Neuro Convolution Model (HFNCM)

The steady state part is described by fuzzy-neuro model. A nonlinear discrete system can be expressed by fuzzy-neuro model (ANFIS) with n rules. The i-th rule of the model is described as follows [5]:

$$R^i : if\ x_1\ is\ A_{1,i}\ and \ldots and\ x_n\ is\ A_{n,i}\ then\ y_s = d_i \quad (3)$$

where n is the number of inputs, $x=[x_1,\ldots,x_n]^T$ is a vector of inputs of the model, $A_{j,i}(x_j)$ is the $i=1,2,\ldots,M_j$ –th antecedent fuzzy set referring to the j-th input, where M_j is the number of the fuzzy set on the j-th input domain.
The first element of the input vector is the steady-state input $x_1=u_s$.
The output is computed as weighted average of the individual rules consequents

$$y_s = \frac{\sum_{i=1}^{m} \mu_i d_i}{\sum_{i=1}^{m} \mu_i} \quad (4)$$

where the weights $0 \leq \mu_i \leq 1$ are computed as $\mu_i = \prod_{j=1}^{m} A_{ji}(x_j)$, where Π is fuzzy operator, usually been applied as the *min* or the *product* operator and m is number of rules.

Various types of membership functions were examined in our research. Different membership functions were employed for each fuzzy model: triangular and Gaussian as shown in Fig. 1.

The triangular membership functions are defined as follows:

$$A_{j,i}(x_j) = \frac{x_j - a_{j,i-1}}{a_{j,i} - a_{j,i-1}}, \qquad a_{j,i-1} \leq x_j < a_{j,i}$$
$$A_{j,i}(x_j) = \frac{a_{j,i+1} - x_j}{a_{j,i+1} - a_{j,i}}, \qquad a_{j,i} \leq x_j < a_{j,i+1} \quad (5)$$

where $x_j \in (a_{j,m_j}, a_{j,m_j+1}\rangle$.

The Gaussian membership functions are defined as follows [6]:

$$A_{j,i-1}(x_j) = e^{-\frac{(x_j - a_{j,i-1})^2}{2\delta_{j,i-1}^2}}$$
$$A_{j,i}(x_j) = e^{-\frac{(x_j - a_{j,i})^2}{2\delta_{j,i}^2}} \quad (6)$$
$$A_{j,i+1}(x_j) = e^{-\frac{(x_j - a_{j,i+1})^2}{2\delta_{j,i+1}^2}}$$

where $a_{j,i-1}$, $a_{j,i}$, $a_{j,i+1}$ are the centre, $\delta_{j,i-1}$, $\delta_{j,i}$, $\delta_{j,i+1}$ the width of the fuzzy sets and $x_j \in < a_{j,m_j-1}, a_{j,m_j+1} >$.

The gain of the steady-state fuzzy model can be computed as

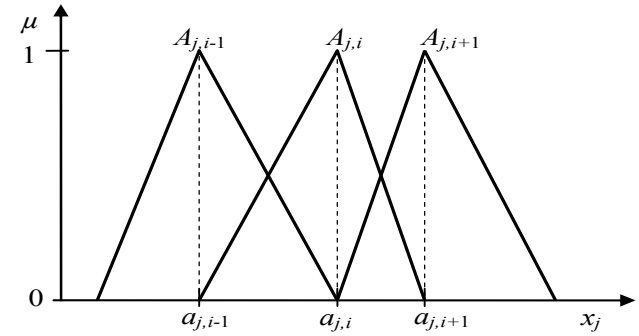
$$K_j = \frac{\partial y_s}{\partial u_s} = \sum_{i=m_j}^{m_j+1} \left[\left(\frac{\Gamma_{i-1}(u_s)}{a_{1,i} - a_{1,i-1}} - \frac{\Gamma_i(u_s)}{a_{1,i+1} - a_{1,i}} \right) \prod_{j=2}^{n} A_{j,i}(x_j) d_i \right], \quad (7)$$

$$K = \sum_{j=1}^{n} K_j$$

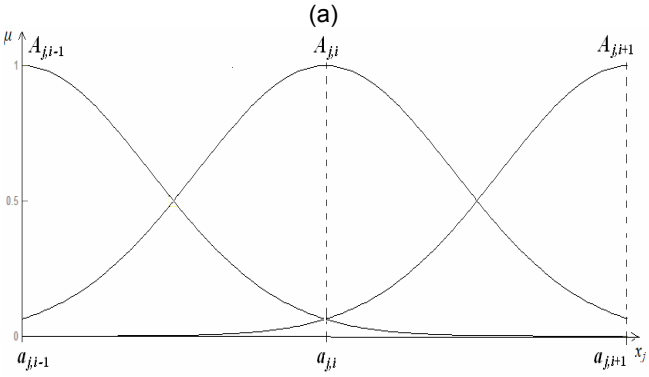
where

$$\Gamma_i = 1 \quad if \quad u_s \in (a_{1,i}, a_{1,i+1}\rangle$$

$$\Gamma_i = 0 \quad if \quad u_s \notin (a_{1,i}, a_{1,i+1}\rangle$$



(a)



(b)

Fig.1 Membership functions used for the fuzzy model: triangular (a), Gaussian (b)

1.2 Dynamic part of the HFNCM

The dynamic state part is described by the impulse response model (IRM). Parameters of the discrete IRM g_i (i=0,…, N, where N is the model horizon) can be easily calculated from the input-output data (u_i and y_i) of the process.

$$y(k) = \sum_{i=0}^{k} g_i u(k-i) \quad (8)$$

In matrix form

$$\begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} u_0 & 0 & \cdots & 0 & \cdots \\ u_1 & u_0 & \cdots & 0 & \cdots \\ \vdots & \vdots & & & \vdots \\ u_N & u_{N-1} & \cdots & u_0 & \cdots \end{bmatrix} \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_N \end{bmatrix}$$

The parameters are given as follows

$$g = \left(U^T U\right)^{-1} U^T y \quad (9)$$

2 Hybrid Fuzzy-Neuro Model Based Predictive Controller

The nonlinear HFNCM can be easily applied in model based predictive control scheme.
In most cases, the difference between system outputs and reference trajectory is used in combination with a cost function on the control effort. A general objective function is the following quadratic form [4]

$$J = \sum_{i=1}^{p} [\hat{y}(k+i \mid k) - r(k+i)]^2 \Gamma_y + \sum_{i=1}^{m} (\Delta u(k+i-1))^2 \Gamma_u \quad (10)$$

Here, r is desired set point, Γ_u (Γ_u=γK^2) and Γ_y are weight parameters, determining relative importance of different terms in the cost function, u and Δu are the control signal and its increment, respectively. Parameter p represents length of the prediction horizon, m is the length of the control horizon. Output predicted by the nonlinear fuzzy model is $\hat{y}(k)$.

$$\hat{y}(k) = K \sum_{i=1}^{\infty} s_i \Delta u(k-i) \quad (11)$$

where $s_i = \sum_{j=1}^{i} g_j$ are the step response coefficients and the change on the control variable is $\Delta u(k)$=$u(k)$-$u(k$-1).

Model predictions along the prediction horizon p are

$$\hat{y}(k+j \mid k) = K \sum_{i=1}^{\infty} s_i \Delta u(k+j-i) + e(k+j \mid k) \quad (12)$$

Disturbances are considered to be constant between sample instants

$$e(k+j \mid k) = y(k \mid k) - K \sum_{i=1}^{\infty} s_i \Delta u(k+j-i) \quad (13)$$

where $y(k \mid k)$ represents the measured value of the process output at time k.

So

$$\hat{y}(k+j \mid k) = K \sum_{i=1}^{N} s_i \Delta u(k+j-i) + f(k+j \mid k) \quad (14)$$

where

$$f(k+j \mid k) = y(k \mid k) + K \sum_{i=1}^{N} (s_{k+1} - s_i) \Delta u(k-i) \quad (15)$$

Prediction of the process output along the length of the prediction horizon, can be written compactly using matrix notation

$$\hat{y}(k) = KS\Delta u(k) + f(k) \quad (16)$$

Matrix S is called the system's dynamic matrix (17) [4]

$$S = \begin{bmatrix} s_1 & 0 & \cdots & 0 \\ s_2 & s_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ s_m & s_{m-1} & \cdots & s_1 \\ \vdots & \vdots & \ddots & \vdots \\ s_p & s_{p-1} & \cdots & s_{p-m+1} \end{bmatrix}_{p \times m} \quad (17)$$

By minimizing its objective function (10) the optimal solution is then given

$$\Delta u(k) = \frac{1}{K}(S^T \Gamma_y S + \gamma I)^{-1} S^T \Gamma_y e(k) \quad (18)$$

In many control applications the desired performance cannot be expressed solely as a trajectory following problem. Many practical requirements are more naturally expressed as constraints on process variables such as manipulated variable constraints, manipulated variable rate constraints or output variable constraints. The solution calls into existence of quadratic programming solution of the control problem.

2.1 Algorithm for the HFCM based control

The algorithm has the following steps (Abonyi, *et al*, 1999):

- Calculation impulse response model g_i from (9),
- Calculation of u_s from y_s=$y(k)$, considering the inversion of the fuzzy-neuro model,
- Calculation of the value of the steady-state gain K by (7),
- Calculation of S by (17) and e by (13),
- Calculation of the controller output from the first element of the calculated Δu vector generated from (18).

3 Case Study and Simulation Results

3.1 Case Study

The application considered involves an isothermal reactor in which the Van Vusse reaction kinetic scheme is carried out. In the following analysis, A is the educt, B the desired product, C and D are unwanted byproducts [2].

$$A \xrightarrow{k_1} B \xrightarrow{k_2} C$$
$$2A \xrightarrow{k_3} D \quad (19)$$

From a design perspective the objective is to make k_2 and k_3 small in comparison to k_1 by appropriate choice of catalyst and reaction conditions. The concentration of B in the product may be controlled by manipulating of the inlet flow rate and/or the reaction temperature.

The educt flow contains only cyclopentadiene in low concentration, C_{Af}. Assuming constant density and an ideal residence time distribution within the reactor, the mass balance equations for the relevant concentrations of cyclopentadiene and of the desired product cyclopentanol, C_A and C_B, are as follows:

$$\dot{C}_A = -k_1 C_A - k_3 C_A^2 + \frac{F}{V}(C_{Af} - C_A)$$
$$\dot{C}_B = k_1 C_A - k_2 C_B - \frac{F}{V} C_B \quad (20)$$
$$y = C_B$$

This example has been considered by a number of researchers as a benchmark problem for evaluating nonlinear process control algorithm.

By normalizing the process variables around the following operating point and substituting the values for the physical constants, the process model becomes:

$$\dot{x}_1(t) = -50\,x_1(t) - 10\,x_1^2(t) + u(10 - x_1(t))$$
$$\dot{x}_2(t) = 50\,x_1(t) - 100\,x_2(t) + u(-x_2(t)) \quad (21)$$
$$y(t) = x_2(t)$$

where the deviation variable for the concentration of component A is denoted by x_1, the concentration of component B by x_2, and the inlet flow rate by u.

3.2 Simulation Results

The comparison of time responses of output of HFNCM with nonlinear plant is shown in Fig. 2. Time responses of the controlled and reference variables under HFNMBPC are shown in Fig. 3 and Fig. 4.

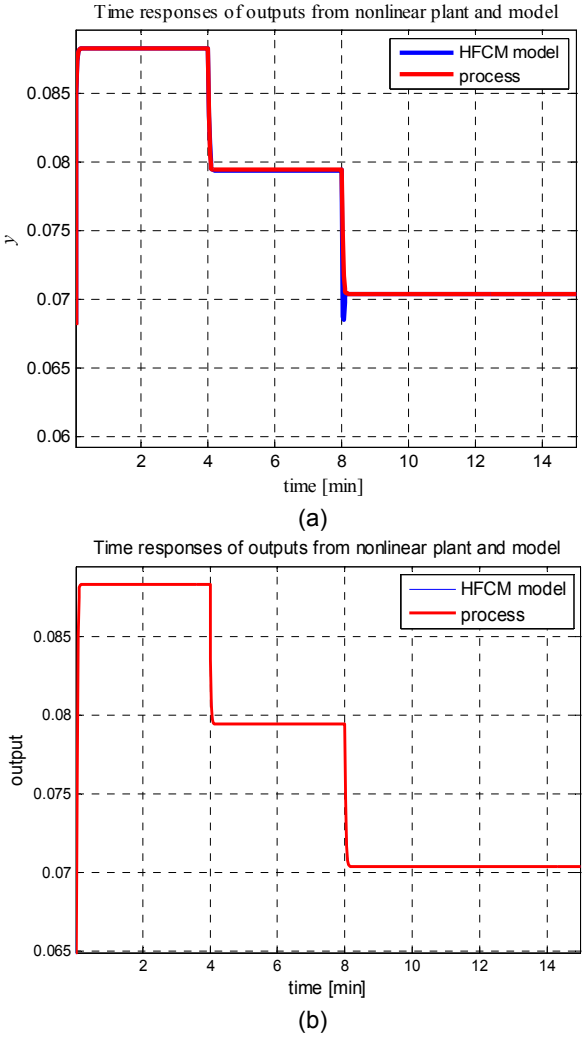


Fig.2 Time responses of output from the nonlinear plant and the HFNCM model with membership functions triangular (a), Gaussian (b)

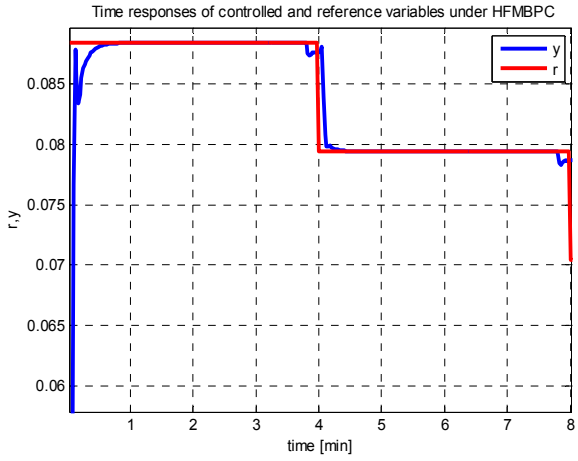
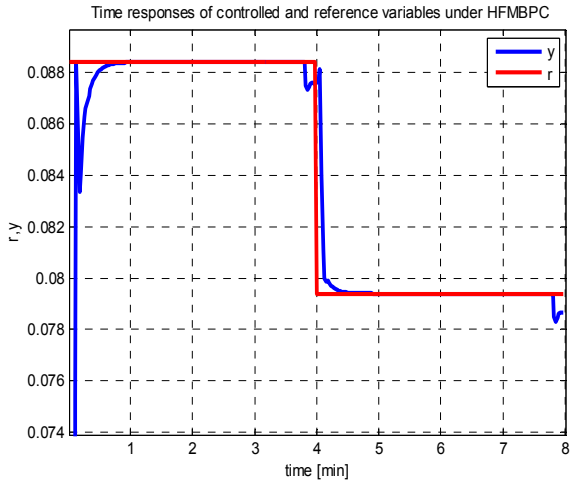


Fig.3 Time responses of the controlled and reference variables under HFNMBPC (m=5, p=10, Γ_y=K, Γ_u=γK^2, γ=1) with triangular membership functions



(b)

Fig.4 Time responses of the controlled and reference variables under HFNMBPC (m=5, p=10, $\Gamma_y=K$, $\Gamma_u=\gamma K^2$, γ=1) with Gaussian membership functions

Conclusion

The HFMBPC uses the advantage of fuzzy systems in the representation of the steady-state behavior of the system. Other advantage is that it tries to combine knowledge about the system in form of a priori knowledge and measured data in the identification of a control relevant model.

Simulation example illustrates the potential offered by the HFNCMBPC.

Acknowledgments

This paper has been supported by the Slovak Scientific Grant Agency, Grant No. 1/0544/09.

References

[1] ABONYI, J., BÓDIZS, Á., NAGY, L., SZEIFERT, F.: Hybrid fuzzy convolution model and its application in predictive control. Chemical Engineering Research and Design 78 (2000), pp. 597-604.

[2] PAULUSOVÁ, J., KOZÁK, Š.: Robust Predictive Fuzzy Control. 7th Portuguese Conference on Automatic Control, CONTROLO'2006, Lisbon, Portugal: 11-13 September 2006, MA-7-1

[3] PAULUSOVÁ, J., KOZÁK, Š.: The comparison of the conventional controllers with fuzzy controllers. Tatranské Matliare, Slovac Republic, May 31-June 3, 1999, 168-171

[4] PAULUSOVÁ, J., KOZÁK, Š.: Nonlinear model-based predictive control. Control Systems Design 2003 CSD'03. Bratislava, Slovak Republic: 7-10 September, 2003, pp. 171-175.

[5] PAULUSOVÁ, J., KOZÁK, Š.: Robust and Fuzzy Dynamic Matrix Control Algorithm. 5th International Carpathian Control Conference 2004. Zakopane, Poland: 25-28 May, pp. 145-152

[6] PAULUSOVÁ, J., DÚBRAVSKÁ, M.: Hybrid Predictive Control of Nonlinear Process. International Slovak Society for Cybernetics and Informatics, Vyšná Boca, Slovak Republic: 10-13 February 2010, 21_Paulusova_Dubravska.pdf

Ing. Jana Paulusová
Ing. Mária Dúbravská

Slovak University of Technology
Faculty of Electrical Engineering
and Information Technology
Ilkovičova 3, 812 19 Bratislava, Slovak Republic
Tel.: +421 2 60291262 Fax: +421 2 65429521
e-mail: jana.paulusova@stuba.sk

Diskrétny systém riadenia s premenlivou štruktúrou a predikciou stavu

Michal Boršč, Branislav Thurský, Jaroslav Hricko

Abstrakt
Článok sa zaoberá vlastnosťami a návrhom diskrétnych systémov riadenia s premenlivou štruktúrou. Objektom riadenia je lineárny časovo invariantný systém. Algoritmus riadenia je kombináciou diskrétneho stavového riadenia s konštantnými parametrami a časovo optimálneho riadenia s premenlivou štruktúrou. Pri návrhu sa využíva metóda asymptotikej hyperroviny. Podľa tejto metódy sa navrhuje systém stavového riadenia s konštantnými parametrami tak, aby stavovom priestore existovala hyperrovina, ku ktorej sa asymptoticky zbiehajú trajektórie riadeného objektu. Asymptotická hyperrovina je plochou prepínania parametrov riadiaceho systému s premenlivou štruktúrou. Pre elimináciu oscilácií na asymptotickej hyperrovine sa využíva prediktívne riadenie štruktúry systému podľa informácie o aktuálnom a budúcom možnom stave riadeného objektu.

Kľúčové slová: diskrétne riadenie, stavové riadenie, systém s premenlivou štruktúrou, kĺzavý režim, časovo optimálne riadenie.

Úvod

Systémy riadenia s premenlivou štruktúrou sa väčšinou navrhujú tak, aby v im zodpovedajúcom stavovom priestore existovala plocha prepnutia, na ktorej sa skokom menia parametre a štruktúra riadiaceho systému a tým vzniká na nej kĺzavý režim. V diskrétnom číslicovom systéme riadenia parametre sa menia len v okamihoch vzorkovania signálov. To môže vyvolať v kĺzavom režime na ploche prepnutia nežiaduce oscilácie [6].

V tomto článku sa budeme venovať problémom správania sa diskrétneho systému riadenia s premenlivou štruktúrou v okolí plochy prepnutia a eliminácií oscilácií. Pre riadenie lineárneho objektu použijeme kombináciu systému diskrétneho stavového riadenia s konštantnými parametrami a systému riadenia s premenlivou štruktúrou.

Systém diskrétneho stavového riadenia bude navrhnutý tak, aby v zodpovedajúcom stavovom priestore existovala asymptotická hyperrovina, ku ktorej sa asymptoticky blížia trajektórie riadeného objektu. Procesy na asymptotickej hyperrovine majú mať predpísanú kvalitu.

V systéme riadenia s premenlivou štruktúrou sa parametre menia v danom alebo zvolenom rozsahu tak, aby riadený objekt vždy prešiel z ľubovoľného stavu na asymptoticku hyperrovinu za minimálny čas. Asymptotická hyperrovina je zároveň plochou prepnutia parametrov riadiaceho systému. Pre elimináciu oscilácií na asymptotickej hyperrovine v dôsledku diskrétneho riadenia s periódou vzorkovania T použijeme predikciu stavu riadeného objektu o čas T dopredu.

Niektorí autori tiež riešia problém oscilácií prediktívnym riadením [7]. Navrhujú takú zmenu parametrov, pri ktorej diskrétne riadený objekt v príslušnom takte riadenia prejde presne na plochu prepnutia. V našej práci predikciu nepoužijeme na to, aby zmenou parametrov bol dosiahnutý presne stav na hyperrovine, lebo pri tomto spôsobe riadenia sa používajú nielen hraničné hodnoty parametrov ale aj hodnoty z vnútra dovoleného intervalu, ktoré treba počítať. V našom prípade budeme zisťovať či pri aktuálnom riadení s hraničnými hodnotami parametrov objekt na konci taktu diskrétneho riadenia prejde alebo neprejde na opačnú stranu asymptotickej hyperroviny ako je v aktuálnom stave. V prípade, že by mal prejsť na opačnú stranu, signál od premenlivých zložiek parametrov sa vypne a objekt v tomto takte a spravidla potom aj v ďalších taktoch je riadený len podľa algoritmu stavového riadenia s konštantnými parametrami.

1. Návrh stavového riadenia lineárneho systému s konštantnými parametrami metódou asymptotickej hyperroviny

Pri návrhu diskrétneho systému riadenia s premenlivou štruktúrou väčšina autorov vychádza z diskrétneho modelu riadeného objektu a riadiaceho systému tvare diferenčných a algebrických stavových rovníc [3], [4], [5]. Autonómne mikroprocesorové riadiace systémy pracujú s vysokou frekvenciu a malou periódou vzorkovania. V takom prípade je možno navrhovať najprv spojité riadenie pre spojitý model objektu a potom vykonať diskretizáciu spojitého systému, resp. algoritmu riadenia a prípadne urobiť ďalšie úpravy algoritmu. Tento prístup bol použitý aj v našej práci.

Teoretickým východiskom syntézy diskrétneho systému riadenia s premenlivou štruktúrou je metóda asymptotickej hyperroviny [1]. Táto práca nadväzuje tiež na publikáciu [2], ktorá sa zaoberá časovo optimálnym riadením systémov s premenlivou štruktúrou.

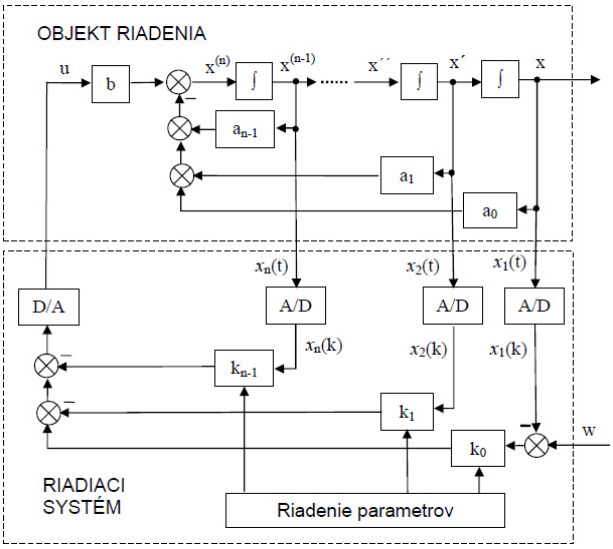
Pri analýze a syntéze systému stavového riadenia s premenlivou štruktúrou vychádzame z kanonického

tvaru matematického modelu, zobrazeného na obr. 1. Stavovými veličinami riadeného objektu sú: výstupná veličina *x* a jej *n-1* derivácií podľa času, to znamená

$x_1=x, \quad x_2=x', \ \ldots\ , x_n=x^{(n-1)}$ (1)

Vektor stavových veličín ďalej budeme označovať

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \ldots & x_n \end{bmatrix}^T$$



Obr.1 Diskrétny systém stavového riadenia s premenlivou štruktúrou

Fig. 1 Discrete variable structure state-space control system

Vstupnou veličinou riadeného objektu je akčná veličina *u*. Veličina w je vstupnou referenčnou veličinou celého systému riadenia. Veličiny *b, a0, a1,..., an-1* sú parametrami lineárneho modelu riadeného objektu. Veličiny *k0, k1,..., kn-1* sú parametrami riadiaceho systému, ktoré sa menia v závislosti od dynamického stavu riadeného objektu. Variabilné parametre riadiaceho systému vyjadríme ako súčet konštantnej k_{ic} a variabilnej zložky k_{iv}

$k_i=k_{ic}+k_{iv}$; pre i=0,1,...,n-1 (2)

Rozsah variabilnej zložky uvažujeme symetrický vo vzťahu k nulovej hodnote

$$-k_{iA} \le k_{iv} \le k_{iA} \tag{3}$$

Vektor konštantných zložiek parametrov riadiaceho systému označíme

$$\mathbf{k}_c = \begin{bmatrix} k_{0c} & k_{1c} & \ldots & k_{(n-1)c} \end{bmatrix}^T,$$

vektor variabilných zložiek

$$\mathbf{k}_v = \begin{bmatrix} k_{0v} & k_{1v} & \ldots & k_{(n-1)v} \end{bmatrix}^T$$

a vektor hraničných hodnôt variabilných zložiek

$$\mathbf{k}_A = \begin{bmatrix} k_{0A} & k_{1A} & \ldots & k_{(n-1)A} \end{bmatrix}^T$$

Pre zjednodušenie výkladu budeme skúmať dynamiku systému stavového riadenia pre *w*=0, to znamená pre rovnovážny stav v počiatku stavového priestoru.

Spojitý systém stavového riadenia s premenlivou štruktúrou podľa obr.1 bez A/D a D/A prevodníkov opíšeme homogénnou diferenciálnou rovnicou

$$x^{(n)} + \sum_{i=0}^{n-1} (a_i + bk_{ic}) x^{(i)} + \sum_{i=0}^{n-1} bk_{iv} x^{(i)} = 0 \tag{4}$$

V prípade systému stavového riadenia len s konštantnými parametrami riadiaceho systému rovnica (4) bude mať tvar

$$x^{(n)} + \sum_{i=0}^{n-1} (a_i + bk_{ic}) x^{(i)} = 0 \tag{5}$$

Ďalej vychádzame z predpokladu, že charakteristická rovnica k diferenciálnej rovnici (5) má aspoň jeden reálny záporný koreň

$$\lambda = -\frac{1}{T_\sigma} < 0 \tag{6}$$

Diferenciálnu rovnicu (5) potom môžeme napísať v tvare:

$$\left(\frac{d}{dt} + \frac{1}{T_\sigma}\right)\left(\frac{d^{n-1}}{dt^{n-1}} + q_{n-2}\frac{d^{n-2}}{dt^{n-2}} + \ldots + q_1\frac{d}{dt} + q_0\right)\sigma(t) = 0 \tag{7}$$

Označme

$$\left(\frac{d^{n-1}}{dt^{n-1}} + q_{n-2}\frac{d^{n-2}}{dt^{n-2}} + \ldots + q_1\frac{d}{dt} + q_0\right)x(t) = \sigma(t) \tag{8}$$

Rovnica (8) pre $\sigma(t)$=0 v uvažovanom kanonickom stavovom priestore (1) opisuje hyperrovinu prechádzajúcu počiatkom stavového priestoru

$$x_n + q_{n-2}x_{n-1} + \ldots + q_1x_2 + q_0x_1 = \mathbf{q}^T\mathbf{x} = 0 \tag{9}$$

kde

$$\mathbf{q} = \begin{bmatrix} q_0 & q_1 & \ldots & q_{n-1} \end{bmatrix}^T$$

je vektor parametrov hyperroviny.

Rovnici (9) zodpovedá diferenciálna rovnica

$$\left(\frac{d^{n-1}}{dt^{n-1}} + q_{n-2}\frac{d^{n-2}}{dt^{n-2}} + \ldots + q_1\frac{d}{dt} + q_0\right)x(t) = 0 \tag{10}$$

Z rovníc (7) a (8) vyplýva vzťah

$$\frac{d\sigma(t)}{dt} + \frac{1}{T_\sigma}\sigma(t) = 0 \tag{11}$$

Veličina σ (t) vyjadruje odchýlku od hyperroviny.

Podľa približnej Eulerovej metódy numerického riešenia diferenciálnych rovníc rovnicu (11) transformujeme na diferenčnú rovnicu s krokom T rovným perióde vzorkovania diskrétneho systému

$$\sigma(k+1) = \left(1 - \frac{T}{T_\sigma}\right)\sigma(k) \tag{12}$$

Z diferenčnej rovnice (11) vplýva, že pre

$$T < T_\sigma \qquad (13)$$

a ľubovoľný počiatočný stav σ(0) vzdialenosť *σ(t)* od hyperroviny (9) sa bude po exponenciále blížiť k nule. To znamená, že stav systému sa asymptoticky blíži k hyperrovine opísanej rovnicou (9). Z toho ďalej vyplýva, že ak je počiatočný stav systému na asymptotickej hyperrovine, potom aj celá trajektória riešenia homogénnej diferenciálnej rovnice systému (10) leží na tejto hyperrovine.

V špeciálnom prípade, ak

$$T = T_\sigma \qquad (14)$$

riadený objekt prejde z ľubovoľného počiatočného stavu na asymptotickú hyperrovinu za jeden takt. V praktických aplikáciách, keď perióda vzorkovania T je veľmi malá a časovú konštantu T_σ nie je vhodne voliť príliš malú, tento špecifický prípad nemá význam.

Voľbou časovej konštanty T_σ pre danú periódu vzorkovania T určujeme rýchlosť približovania sa stavu systému k asymptotickej hyperrovine. Voľbou parametrov q_0, $q_{1,\dots,}q_{n-2}$ určíme želaný charakter procesov na asymptotickej hyperovine, ktorý zodpovedá riešeniu diferenciálnej rovnice (10).

Požadované parametre T_σ a q_0, $q_{1,\dots,}q_{n-2}$ zabezpečíme v systéme stavového riadenia koeficientmi k_{ic} . Pre ich výpočet upravíme rovnicu (7) do tvaru

$$x^{(n)} + \left(\frac{1}{T_\sigma} + q_{n-2}\right)x^{(n-1)} + \ldots + \left(\frac{q_1}{T_\sigma} + q_0\right)x' + \frac{q_0}{T_\sigma}x = 0 \qquad (15)$$

Porovnaním koeficientov v diferenciálnych rovniciach (5) a (15) dostaneme vzťahy pre výpočet konštantných parametrov k_{ic}:

$$k_{0c} = \frac{1}{b}\left(\frac{q_0}{T} - a_0\right);\quad k_{1c} = \frac{1}{b}\left(\frac{q_1}{T} + q_0 - a_1\right);\ \ldots$$
$$k_{(n-1)c} = \frac{1}{b}\left(\frac{1}{T} + q_{n-2} - a_{n-1}\right) \qquad (16)$$

2. Časovo optimálne riadenie systému s premenlivou štruktúrou

Predpokladáme, že konštantné zložky koeficientov riadiaceho systému k_{ic} sú navrhnuté tak, aby charakteristická rovnica k diferenciálnej rovnici (5) mala reálny záporný koreň. Použitím vzťahov (4) a (9) upravíme diferenciálnu rovnicu (6) do tvaru

$$\frac{d\sigma}{dt} = -\frac{1}{T_\sigma}\sigma - \sum_{i=0}^{n-1} bk_{iv}x^{(i)} \qquad (17)$$

Z rovnice (17) pre σ≠0 vyplýva podmienka kĺzavého režimu a dosiahnuteľnosti asymptotickej hyperoviny :

$$\sigma\frac{d\sigma}{dt} = -\frac{1}{T_\sigma}\sigma - \sigma\sum_{i=0}^{n-1} bk_{iv}x^{(i)} < 0 \qquad (18)$$

Z rovnice (18) vyplývajú pre variabilné zložky parametrov riadiaceho systému podmienky časovo optimálneho procesu:

$$\max_{k_{iv}} \sigma bk_{iv}x^{(i)} \quad \text{pre } i=0,1,\ldots,n-1 \qquad (19)$$

Pre b>0 z podmienky (19) a pre rozsah zmien parametrov k_{iv} vyjadrený vzťahom (3), vyplýva algoritmus zmeny variabilných zložiek parametrov riadiaceho systému, ktorý zabezpečí časovo optimálny prechod systému z ľubovoľného stavu na asymptotickú hyperrovinu

$$k_{iv} = k_{iA}sign(\sigma x^{(i)}) = \frac{k_{i\max} - k_{i\min}}{2}sign\,\sigma.sign x^{(i)}$$

pre i=0,1,…,n-1 (20)

3. Realizácia diskrétneho systému riadenia s premenlivou štruktúrou

Výsledný algoritmus výpočtu akčnej veličiny u vzniká paralelným spojením, teda sčítaním výstupov algoritmov systému stavového riadenia s konštantnými koeficientmi k_{ic} a systému s premenlivými parametrami k_{iv}

$$u = -\sum_{i=0}^{n-1}(k_{ic} + k_{iv})x^{(i)} = -\sum_{i=0}^{n-1} k_{ic}x^{(i)} - \sum_{i=0}^{n-1} k_{iA}\left|x^{(i)}\right| sign\,\sigma \qquad (21)$$

Vzťah (21) bol odvodený pre referenčnú hodnotu w=0. V tomto prípade veličina σ sa počíta podľa vzťahu (9). Pre referenčnú veličinu w=const≠0 vzťah (21) upravíme do tvaru

$$u = k_{0c}(w - x) - \sum_{i=1}^{n-1} k_{ic}x^{(i)} + \left(k_{0A}|w - x| - \sum_{i=1}^{n-1} k_{iA}\left|x^{(i)}\right|\right) sign\,\sigma \qquad (22)$$

kde odchýlka od hyperroviny σ sa počíta podľa vzťahu

$$\sigma = q_0(x - w) + q_1x' + \ldots + q_{n-1}x^{(n-1)} \qquad (23)$$

Algoritmus výpočtu akčnej veličiny vyjadrený vzťahmi (22) a (23) môžeme napísať vo vektorovej forme v tvare

$$u = \mathbf{k}_c^T(\mathbf{w} - \mathbf{x}) + \mathbf{k}_A^T\mathbf{x}_{abs}sign\left(\mathbf{q}^T(\mathbf{w} - \mathbf{x})\right) \qquad (24)$$

kde vektor referenčných hodnôt **w** je

$$\mathbf{w} = \begin{bmatrix} w & 0 & \ldots & 0 \end{bmatrix}^T$$

a vektor absolútnych hodnôt stavových veličín $\mathbf{x}_{abs}$ je

$$\mathbf{x}_{abs} = \begin{bmatrix} |w - x_1| & |x_2| & \ldots & |x_n| \end{bmatrix}^T$$

Algoritmus riadenia systému s premenlivou štruktúrou bol odvodený pre spojité procesy. Riadiaci systém neobsahuje zotrvačné členy a v algoritme sú použité len operácie násobenia vstupných signálov riadiaceho systému, preto ho možno použiť aj pre diskrétne riadenie. Pri použití tvarovacieho člena nultého rádu akčná veličina sa udržuje na konštantnej vypočítanej hodnote počas celého taktu riadenia. To čiastočne ovplyvňuje optimálnosť procesov, lebo prepínanie hodnôt parametrov nenastáva presne vtedy, keď je to potrebné, t.j. v priebehu aktuálneho taktu, ale až na začiatku následného taktu. Pri malej perióde vzorkovania v porovnaní so zotrvačnosťou procesov v riadenom objekte tieto odchýlky môžu byť zanedbateľné.

Väčšie problémy môžu nastať v okolí hyperroviny prepínania parametrov. Prechodom stavu riadeného objektu z jednej strany prepínacej roviny na druhú v diskrétnom

systéme vznikajú oscilácie s periódou vzorkovania *T*. Ak je stav systému vzdialený od rovnovážneho, budú zmeny akčnej veličiny pomerne veľké.

Problém oscilácií možno riešiť odstránením kĺzavého režimu a to tak, že okolo asymptotickej hyperroviny vytvoríme sektor, v ktorom sa bude uplatňovať len stavové riadenie s konštantnými parametrami. Podobný prístup bol použitý v práci [4].

Pre elimináciu oscilácií okolo asymptotickej hyperroviny použijeme iný spôsob. Navrhneme taký algoritmus riadenia zmeny parametrov k_{iv}, pri ktorom sa využíva predikcia stavu. K aktuálnemu stavu vyjadrenému vektorom stavových veličín **x**(*k*) a pre aktuálnu hodnotu akčnej veličiny *u(k)* v *k*-tom takte bude počítaná predikcia stavu v takte *k+1* podľa Eulerovej približnej metódy numerického riešenia diferenciálnych rovníc

$$\begin{aligned} x_1(k+1) &= x_1(k) + Tx_2(k) \\ &\cdots\cdots\cdots\cdots\cdots\cdots \\ x_{n-1}(k+1) &= x_{n-1}(k) + Tx_n(k) \\ x_n(k+1) &= x_n(k) + Tu(k) \end{aligned} \qquad (25)$$

Hodnota veličiny *u(k)* použitá vo vzťahoch (25) sa počíta podľa vzťahov (22) a (23), resp. vzťahu (24).

Vychádzajúc zo vzťahu (23) odchýlka aktuálneho stavu objektu od asymptotickej hyperroviny sa počíta podľa vzťahu

$$\sigma(k) = \mathbf{q}^T(\mathbf{w} - \mathbf{x}(k)) \qquad (26)$$

a predikcia odchýlky v takte *k+1* podľa vzťahu

$$\sigma(k+1) = \mathbf{q}^T(\mathbf{w} - \mathbf{x}(k+1)) \qquad ..(27)$$

Pre elimináciu kĺzavého režimu upravíme algoritmus výpočtu akčnej veličiny (24) tak, že namiesto člena

$$sign\left(\mathbf{q}^T(\mathbf{w} - \mathbf{x})\right)$$

dosadíme trojhodnotovú premennú veličinu

$$z = 0.5\left(sign\,\sigma(k) + sign\,\sigma(k+1)\right) \qquad (28)$$

Nový algoritmus bude opísaný vzťahom

$$u(k) = \mathbf{k}_c^T(\mathbf{w} - \mathbf{x}(k)) + z.\mathbf{k}_A^T\mathbf{x}_{abs}(k) \qquad (29)$$

Ak stav systému v takte *k* a stav vypočítaný predikciou pre takt *k*+1 sa nachádza na tej istej strane asymptotickej hyperoviny, teda ak

$$sign\,\sigma(k) = sign\,\sigma(k+1)$$

hodnota premennej bude *z*=1 alebo *z*=-1.

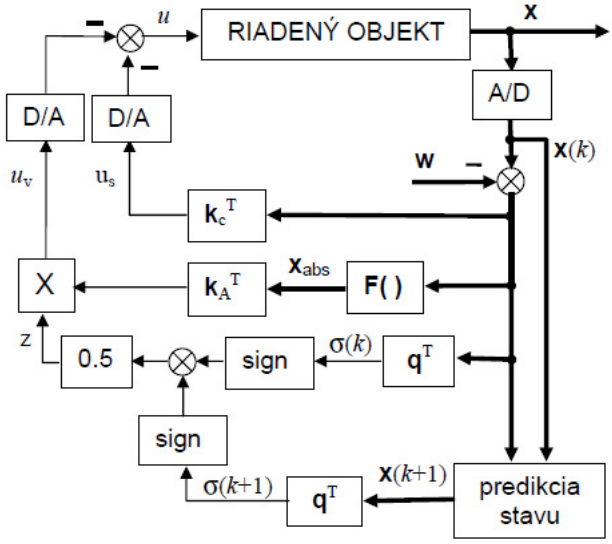
Ak stav systému v takte *k* a stav vypočítaný predikciou pre takt *k*+1 sú na rôznych stranách asymptotickej hyperoviny, teda ak

$$sign\,\sigma(k) \neq sign\,\sigma(k+1)$$

hodnota premennej bude *z*=0.

Pri riadení podľa takto upraveného algoritmu na hyperrovine nevzniká kĺzavý režim. Ak by podľa algoritmu riadenia s premenlivou štruktúrou počas príslušného taktu mal systém prejsť na druhú stranu hyperroviny, zložka akčnej veličiny spôsobená meniacimi sa zložkami parametrov k_{iv} bude v tomto takte vypnutá. Uplatňuje sa len stavové riadenie s konštantnými parametrami. Trajektória systému v stavovom priestore sa asymptotický blíži k asymptotickej hyperrovine. Vzdialenosť od asymptotickej hyperroviny, pri ktorej dochádza k zmene algoritmu riadenia len na stavové riadenie s konštantnými parametrami je v každom prípade iná, vždy je však minimálna. To je výhoda oproti systému, v ktorom pásmo stavového riadenia s konštantnými zložkami parametrov je dopredu pevne vymedzené. Hranice pevne vymedzeného pásma by sa museli dopredu počítať pre najnepriaznivejšie možné stavy.

Schéma diskrétneho systému riadenia s premenlivou štruktúrou s využitím predikcie stavu je na obr. 2.



Obr.2 Diskrétny systém riadenia s premenlivou štruktúrou a predikciou stavu

Fig. 2 Discrete variable structure control system with state prediction

Pre ilustráciu opísaného spôsobu návrhu diskrétneho systému s premenlivou štruktúrou a demonštráciu jeho uvedieme príklad riadenia servomechnizmu, ktorý je opísaný diferenciálnou rovnicou tretieho tvaru

$$x''' + a_2x'' + a_1x' + a_0x = bu$$

pre hodnoty parametrov

$a_2=10;\quad a_1=25;\quad a_0=0;\quad b=50;$

Pre návrh stavového riadenia metódou asymptotickej hyperroviny zvolíme časovú konštantu

T_σ=0.1 s

Perióda vzorkovania je

T=0.01 s

Koeficienty rovnice asymptotickej hyperroviny vypočítame podľa požiadaviek na kvalitu procesov na asymptotickej hyperrovine. Použili sme štandardný tvar charakteristickej rovnice pre aperiodický proces

$s^2+2\Omega s+\Omega^2=0$

Vlastnú frekvenciu systému volíme s ohľadom na fyzikálne obmedzenia a praktické požiadavky na dynamiku systému

Ω=10

Pre zvolenú hodnotu budú koeficienty rovnice asymptotickej hyperroviny:

$q_0 = \Omega^2 = 100;$ $q_1 = 2\Omega = 20;$ $q_2 = 1;$

Podľa vzťahov (15) vypočítame konštantné parametre stavového riadenia LTI systému:

$k_{0c} = (q_0/T_\sigma - a_0)/b = 20$

$k_{1c} = (q_1/T\sigma + q_0 - a_1)/b = 5.5$

$k_{2c} = (q_2/T + q_1 - a_2)/b = 0.40$

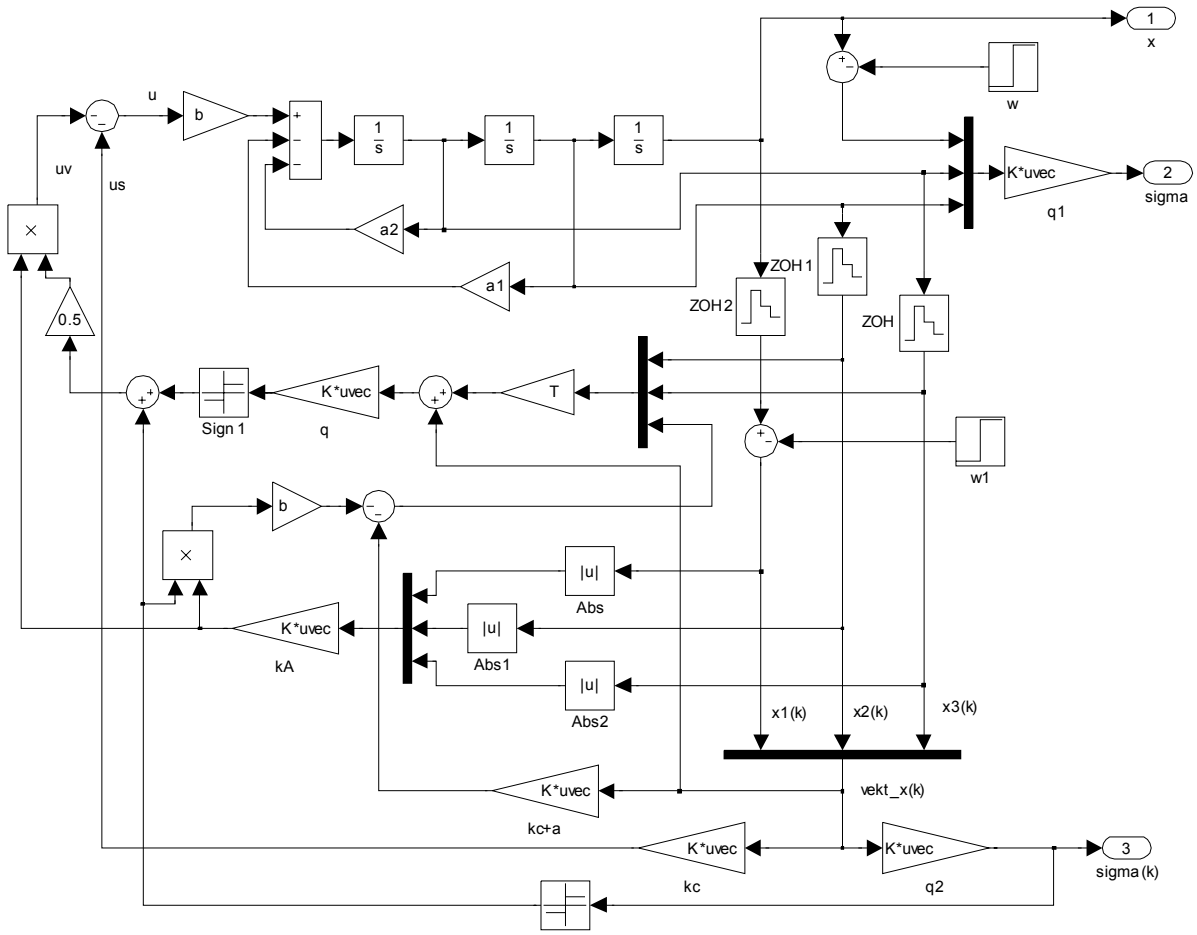
Rozsah zmien variabilných zložiek koeficientov stavového riadenia zvolíme tak, aby pri rozbehoch servomechanizmu systém nebol tlmený, preto zvolíme

$k0A = 0$

$k1A = a1/b + k1c = 6$

$k2A = a2/b + k2c = 0.6$

Simulačný model diskrétneho systému riadenia s premenlivou štruktúrou a predikciou je na obr. 3.



Obr. 3 Simulačný model diskrétneho systému riadenia s premenlivou štruktúrou a predikciou

Fig. 3 Simulation model diskrete variable structure control system with prediction

Výsledky simulácie sú zobrazene grafmi na obr. 4. pre hodnotu referenčnej veličiny w=1 a počiatočný nulový stav.

Na hornom obrázku sú časové priebehy výstupnej veličiny x a to:

1 - pre stavové riadenie s konštantnými parametrami,

2 – pre systém riadenia s variabilnou štruktúrou a kĺzavým režimom bez predikcie,

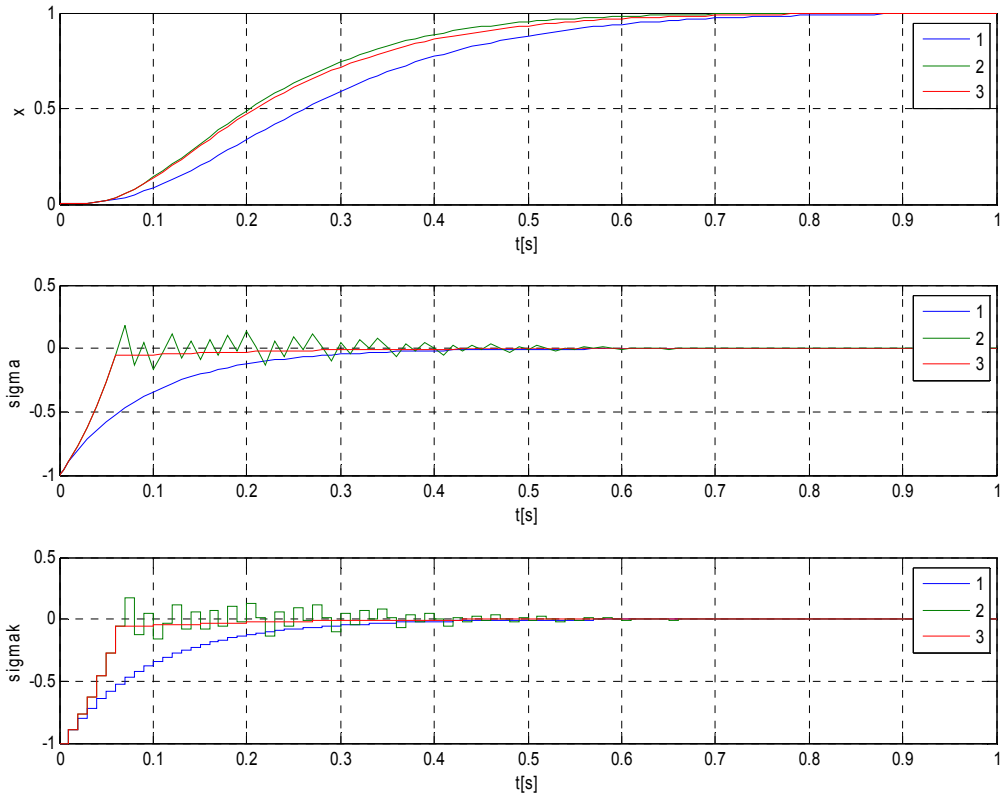
3- pre systém riadenia s variabilnou štruktúrou a predikciou stavu.

Na strednom obrázku sú časové priebehy skutočnej odchýlky od asymptotickej hyperroviny σ, pre tie isté tri varianty systémov riadenia.

Na spodnom obrázku sú časové priebehy vzorkovanej (stupňovej) odchýlky od asymptotickej hyperroviny $\sigma(k)$, ktorej hodnota sa mení v taktoch riadenia k.

Výsledky ukazujú, že časové priebehy výstupnej veličiny x pri použití systému riadenia s premenlivou štruktúrou bez predikcie a s predikciou sú takmer zhodné a sú lepšie, ako v systéme stavového riadenia s konštantnými parametrami. V systéme riadenia s premenlivou štruktúrou bez predikcie nastáva však kĺzavý režim a okolo hyperroviny prepnutia vznikajú oscilácie. V systéme riadenia s premenlivou štruktúrou a predikciou dochádza v blízkosti hyperroviny prepnutia v predstihu k vypínaniu signálu od subsystému s variabilnými parametrami a ďalej sa uplatní len stavové riadenie s konštantnými parametrami. Tým sú odstránené oscilácie okolo hyperroviny prepnutia.

Simulačný experiment potvrdil funkčnosť navrhovanej koncepcie diskrétneho riadenia s premenlivou štruktúrou. Navrhnutý algoritmus však predpokladá existenciu asymptotickej hyperroviny v stavovom priestore.



Obr. 4 Časové priebehy výstupnej veličiny *x* a odchýlky od hyperroviny σ

Fig. 4 Time response of output variable *x* and error from hypperplane σ

Záver

Skúmanie vlastností lineárnych systémov stavového riadenia, ktoré sú navrhované tak, aby v ich stavovom priestore existovala asymptotická hyperrovina ukázalo rád možnosti ich kombinácie so systémami s premenlivou štruktúrou. Možno ich použiť ako základ robustného systému s premenlivou štruktúrou, v ktorom dochádza ku kĺzavému režimu. Umožňujú však tvoriť aj štruktúry, v ktorých nevzniká kĺzavý režim a tým eliminovať nežiaduce oscilácie na hyperrovine prepínania štruktúry.

Pre vylúčenie kĺzavého režimu a oscilácií v diskrétnom systéme riadenia s premenlivou štruktúrou sa okolo asymptotickej hyperroviny vymedzuje pásmo, najlepšie v tvare sektora, v ktorom sa riadenie s premenlivou štruktúrou vypína a zostáva len stavové riadenie s konštantnými parametrami. Oblasť stavového riadenia s konštantnými parametrami možno minimalizovať použitím navrhnutého algoritmu s premenlivou štruktúrou a predikciou stavu o jeden takt diskrétneho riadenia dopredu.

Poďakovanie

Táto práca vznikla na Katedre mechatronických systémoy Fakulty mechatroniky Trenčianskej univerzity Alexandra Dubčeka v Trenčíne v rámci výskumnej úlohy podporovanej Vedeckou grantovou agentúrou Ministerstva školstva Slovenskej republiky: VEGA 1/4056/07 - Analýza a syntéza mechatronických systémov

Literatúra

[1] BORŠČ, M., THURSKÝ, B., HRICKO, J.: Synthesis of Optimal and Robust Systems with Variable Control Structure. TnUAD 2010 (v tlači)

[2] BORŠČ, M., VITKO, A., THURSKÝ, B.: Optimal stabilization of modal control with variable structure. Transactions of the Institute of Measurement and Control, Vol. 30, No. 1. 63-76 (2008) ISSN: 0142-3312

[3] BUČEVAC, Z., M.: Discrete-time variable structure control systems - multivariable linear plant case. University of Niš, The scientific journal Facta Universitatis, Series: mechanics, automatic control and robotics vol.2, no 9, 1999 pp. 983 - 994

[4] FURUTA, K., PAN, Y.: Discrete-time Variable Structure Control. In: First Stage of VSS: People and Events. In: Variable Structure Systems: Towards the 21st Century. Springer Verlag Berlin/Heidelberg, Vol 274 (2002), ISSN 0170-8643, pp. 57-81

[5] ISKRENOVIĆ-MOMČILOVIC, O.: Discrete–time variable structure controller for aircraft elevator kontrol. Journal of ELECTRICAL ENGINEERING, VOL. 59, NO. 2, 2008, 92–96

[6] KARDOŠ, J.: Theory of Variable Structure Systéme and the Time Sub-Optimal Position Control, Edition 1, HMH, Ltd., Bratislava 2007, ISBN 978-80-969725-0-0

[7] XIAO, L., SU, H., ZHANG, X., CHU, J.: A New Discrete Variable Structure Control Algorithm Based on Sliding Mode Prediction. American Control Conference, June 8-10, 2005. Portland, OR, USA

Abstract

This paper deals about properties and design of the discrete variable structure control systems. Object of control is linear time invariant system. The control algorithm is combination of discrete state-space control with constant parameters and time optimal control with variable structure. By design is used asymptotic hyper-plane method. According this method is designed state-space control system with constant parameters so that in state-space exists like this hyper-plane towards who's asymptotic converge trajectories of the controlled object. Asymptotic hyper-plane is switching plane of parameters of control system with variable structure. For elimination of oscillations on the asymptotic hyper-plane is used predictive control of system structure according to information about actual and expectant possible state of controlled object.

Michal Borščč, prof. Ing., CSc.

Branislav Thurský, Ing.

Jaroslav Hricko, Ing.

Trenčianská univerzita Alexandra Dubčeka v Trenčíne
Fakulta mechatroniky
Katedra mechatronických systémov
Pri parku 19
911 06 Trenčín
Tel.: 00421 32 7417544

E-mail borsc@tnuni.sk

E-mail thursky@tnuni.sk

E-mail hricko@tnuni.sk

Systém riadenia s premenlivou štruktúrou bez kĺzavého režimu

Michal Boršč, Branislav Thurský, Jaroslav Hricko

Abstrakt
V článku je opísaný návrh systému stavového riadenia s premenlivou štruktúrou metódou asymptotickej hyperroviny. Riadiaci systém je reprezentovaný paralelným spojením dvoch systémov. Prvý je systém riadenia s konštantnými parametrami. Navrhuje sa tak, aby v stavovom priestore existovala asymptotická hyperrovina, ku ktorej sa zbiehajú trajektórie prechodných procesov, a aby procesy na asymptotickej hyperrovine mali želanú kvalitu. Druhý je systém riadenia s premenlivou štruktúrou, ktorý je transformovaný na tzv. kvázi reléový. Navrhuje sa tak, aby prechod systému z ľubovoľného stavu na asymptotickú hyperrovinu bol časovo optimálny a zabezpečoval na nej kĺzavý režim. Systém riadenia, je ďalej upravený tak, aby bol odstránený kĺzavý režim a aby sa pritom kvalita procesov podstatne nezhoršila.

Kľúčové slová: stavové riadenie, systém s premenlivou štruktúrou, kĺzavý režim, časovo optimálne riadenie.

Úvod

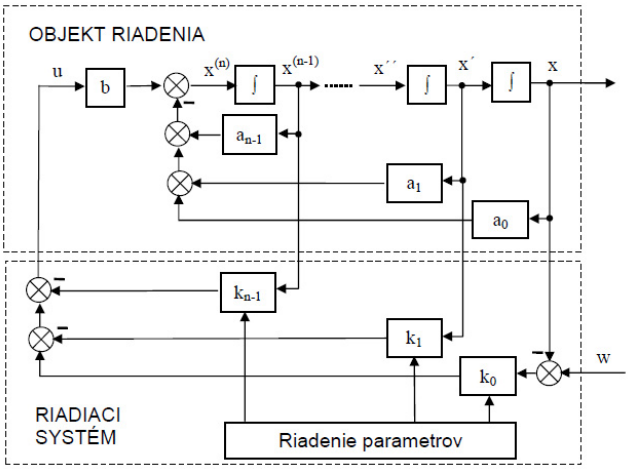
Systémom riadenia s premenlivou štruktúrou nazývame taký systém riadenia, v ktorom sa algoritmus riadenia, alebo parametre algoritmu riadenia menia, spravidla skokom, v závislosti od okamžitého dynamického stavu riadeného objektu [5]. V systémoch riadenia s premenlivou štruktúrou možno dosiahnuť väčšiu robustnosť a variabilitu dynamiky procesov než v lineárnych systémoch riadenia s konštantnými parametrami. Výskum systémov s premenlivou štruktúrou sa sústreďuje hlavne na zabezpečenie tzv. kĺzavého režimu a kvality procesov v kĺzavom režime. V kĺzavom režime sú procesy invariantné voči zmenám parametrov riadeného objektu. Niektoré práce sa venujú optimalizácií procesov v kĺzavom režime, napr. [6], [7]. Pri praktickej realizácií systémov riadenia s premenlivou štruktúrou, v dôsledku nepredvídaných zmien parametrov riadeného objektu môže kĺzavý režim prechádzať do nežiaducich nízkofrekvenčných autooscilácií. Problematike odstraňovania takých autooscilácií je venovaná práca [4]. Mennej prác sa venuje využitiu variability dynamiky procesov, to znamená využitiu zmeny parametrov na zmenu charakteru a kvality procesov v závislosti od dynamického stavu riadeného objektu. Napríklad v práci [3] je použité premenlivé tlmenie na zlepšenie dynamických vlastností semi-aktívneho pruženia vozidla. Algoritmom riadenia s premenlivou štruktúrou navrhnutým v práci [2] sa zabezpečujú časovo optimálne procesy mimo oblasti kĺzavého režimu a aperiodické procesy v kĺzavom režime, t. j. v stavovom priestore na hyperrovine prepínania štruktúry riadiaceho systému.

Pri riadení objektov, ktorých parametre sa praktický nemenia, napríklad elektromechanických systémov, nestojí v popredí úloha zabezpečiť robustnosť systému, ale zabezpečiť kvalitu dynamických procesov. V lineárnych systémoch s konštantnými parametrami sú procesy rovnako tlmené v každom dynamickom stave. Analýza procesov v stavovom priestore však ukazuje, že to nie je potrebné. Pre stabilizáciu procesov stačí, ak sú procesy tlmené len v určitých oblastiach stavového priestoru, najmä v okolí rovnovážneho stavu. Takto možno prechodné procesy stabilizácie zrýchliť. Na druhej strane ukazuje sa možnosť prechodné procesy v okolí rovnovážneho stavu viac zatlmiť. Tým sa proces ustaľovania systému v tejto fáze predlží, ale zvýši sa aj robustnosť systému. Ak sú procesy zatlmené len v určitej oblasti, neovplyvni to charakter prechodného procesu v ostatných fázach a stavoch.

V tomto príspevku opisujeme metódu návrhu systému stavového riadenia s premenlivou štruktúrou pre lineárne objekty s konštantnými parametrami a skokové zmeny želanej hodnoty riadenej veličiny. Pre daný objekt vymedzíme v stavovom priestore oblasť stabilizácie procesov. Hodnoty premenlivých parametrov riadiaceho systému v oblasti stabilizácie majú zabezpečiť želanú dynamiku procesu stabilizácie podľa nejakého kritéria kvality prechodných procesov. Mimo oblasti stabilizácie systém riadenia má meniť parametre tak, aby riadený objekt vždy prešiel do oblasti stabilizácie za minimálny čas, teda aby procesy z hľadiska zmeny parametrov boli časovo optimálne. Obmedzenia na akčné veličiny pre zjednodušenie výkladu nebudeme uvažovať. Riadiaci systém má zároveň zabezpečiť, aby riadený objekt po dosiahnutí oblasti stabilizácie sa ďalej pohyboval už len v tejto oblasti, a to bez kĺzavého režimu.

1. Návrh stavového riadenia lineárneho systému s konštantnými parametrami metódou asymptotickej hyperroviny

Teoretickým východiskom syntézy systému riadenia s premenlivou štruktúrou bude metóda asymptotickej hyperroviny [1], vypracovaná v rámci riešenia projektu VEGA 1/4056/07 - Analýza a syntéza mechatronických systémov. Táto práca nadväzuje tiež na publikáciu [2], ktorá sa zaoberá časovo optimálnym riadením systémov s premenlivou štruktúrou. V práci [1] bol riešený problém zabezpečenia robustnosti systému riadenia s premenlivou štruktúrou v kĺzavom režime pri meniacich sa parametroch riadeného objektu v známom rozsahu. V tomto príspevku aplikujeme metódu na syntézu systému riadenia s premenlivou štruktúrou bez kĺzavého režimu.



Obr.1 Systém stavového riadenia s premenlivou štrukturou

Fig. 1 Variable structure state-space control system

Pri analýze a syntéze systému stavového riadenia s premenlivou štruktúrou vychádzame z kanonického tvaru matematického modelu, zobrazeného na obr. 1. Stavovými veličinami riadeného objektu sú: výstupná veličina *x* a jej n-1 derivácií podľa času, t. j. x, x´, x´´,...,$x^{(n-1)}$. Vstupnou veličinou riadeného objektu je akčná veličina *u*. Veličina w je vstupnou referenčnou veličinou celého systému riadenia. Veličiny *b, a_0, a_1,..., a_{n-1}* sú parametrami lineárneho modelu riadeného objektu. Veličiny *k_0, k_1,..., k_{n-1}* sú parametrami riadiaceho systému. Parametre riadiaceho systému sa budú meniť v závislosti od dynamického stavu riadeného objektu. Úlohou je navrhnúť algoritmus riadenia objektu zmenou parametrov riadiaceho systému k_i s cieľom zabezpečiť vyššiu kvalitu prechodných procesov, než akú je možno dosiahnuť v systémoch stavového riadenia s konštantnými parametrami.

Uvažovaný rozsah zmien parametrov riadiaceho systému je

$$k_{i\,\min} \le k_i \le k_{i\,\max}; \qquad i=0,1,...,n-1 \tag{1}$$

Variabilné parametre riadiaceho systému vyjadríme ako súčet konštantnej a variabilnej zložky

$$k_i=k_{ic}+k_{iv} \tag{2}$$

Hodnoty konštantných zložiek zvolíme rovné stredným hodnotám z rozsahu zmien parametrov

$$k_{ic} = \frac{k_{i\max} + k_{i\min}}{2} \tag{3}$$

V takom prípade variabilné zložky vyhovujú nerovniciam

$$\left|k_{iv}\right| \le \frac{k_{i\max} - k_{i\min}}{2} = k_{iA} \tag{4}$$

To znamená, že rozsah ich zmien je symetrický k nulovej hodnote

$$-k_{iA} \le k_{iv} \le k_{iA} \tag{5}$$

Pre zjednodušenie výkladu budeme skúmať dynamiku systému stavového riadenia pre *w*=0, to znamená pre rovnovážny stav v počiatku stavového priestoru. Systém stavového riadenia s premenlivou štruktúrou znázornený na obr.1 v tomto prípade môžeme opísať homogénnou diferenciálnou rovnicou

$$x^{(n)}+(a_{n-1}+bk_{n-1})x^{(n-1)}+...+(a_1+bk_1)x'+(a_0+bk_0)x=0 \tag{6}$$

Po oddelení členov s konštantnými a variabilnými zložkami parametrov homogénna diferenciálna rovnica systému (6) bude mať tvar

$$x^{(n)} + \sum_{i=0}^{n-1}(a_i + bk_{ic})x^{(i)} + \sum_{i=0}^{n-1} bk_{iv}x^{(i)} = 0 \tag{7}$$

V prípade systému stavového riadenia len s konštantnými parametrami riadiaceho systému rovnica (7) bude mať tvar

$$x^{(n)} + \sum_{i=0}^{n-1}(a_i + bk_{ic})x^{(i)} = 0 \tag{8}$$

Skúmajme vlastnosti špecifického prípadu systému stavového riadenia s konštantnými koeficientmi, keď charakteristická rovnica k diferenciálnej rovnici (7) má aspoň jeden reálny záporný koreň $\lambda<0$. Reálnemu zápornému koreňu zodpovedá časová konštanta $T=-1/\lambda$. Diferenciálnu rovnicu (8) v takom prípade môžeme napísať v tvare:

$$\left(\frac{d}{dt}+\frac{1}{T}\right)\left(\frac{d^{n-1}}{dt^{n-1}}+q_{n-2}\frac{d^{n-2}}{dt^{n-2}}+...+q_1\frac{d}{dt}+q_0\right)x(t)=0 \tag{9}$$

Označme

$$\left(\frac{d^{n-1}}{dt^{n-1}}+q_{n-2}\frac{d^{n-2}}{dt^{n-2}}+...+q_1\frac{d}{dt}+q_0\right)x(t)=\sigma(t) \tag{10}$$

Rovnica (10) pre $\sigma(t)=0$ v uvažovanom kanonickom stavovom priestore opisuje hyperrovinu prechádzajúcu počiatkom stavového priestoru. Zodpovedá jej diferenciálna rovnica

$$\left(\frac{d^{n-1}}{dt^{n-1}}+q_{n-2}\frac{d^{n-2}}{dt^{n-2}}+...+q_1\frac{d}{dt}+q_0\right)x(t)=0 \tag{11}$$

Z rovníc (9) a (10) vyplýva vzťah

$$\frac{d\sigma(t)}{dt}+\frac{1}{T}\sigma(t)=0 \tag{12}$$

Veličina $\sigma(t)$ vyjadruje odchýlku od hyperroviny.

Rovnicu (12) vynásobíme veličinou $\sigma(t)\neq 0$ a upravíme do tvaru:

$$\sigma(t)\frac{d\sigma(t)}{dt}=-\frac{1}{T}\sigma^2(t)<0 \tag{13}$$

Zo vzťahu (13) vyplýva, že pre ľubovoľný počiatočný stav odchýlka $\sigma(t)$ sa monotónne blíži k nulovej hodnote. To znamená, že stav systému sa asymptoticky blíži k hyperrovine opísanej rovnicou (11). Z toho ďalej vyplýva, že ak je počiatočný stav systému na asymptotickej hyperrovine, potom aj celá trajektória riešenia homogénnej diferenciálnej rovnice systému (11) leží na tejto hyperrovine.

Časovú konštantu T a parametre asymptotickej hyperoviny q_0, $q_{1,...,}q_{n-2}$ vo vzťahoch (9) môžeme voliť ľubovoľne. Voľbou parametrov q_0, $q_{1,...,}q_{n-2}$ určíme želaný charakter procesov na asymptotickej hyperovine. Voľbou časovej konštanty T určíme želanú rýchlosť konvergencie systému k asymptotickej hyperrovine. Kritérium kvality prechodných procesov systému stavového riadenia *n*-tého rádu bude teraz vyjadrené dvoma nezávislými kritériami: kritériom

kvality procesov na asymptotickej hyperrovine a rýchlosťou konvergencie procesov k asymptotickej hyperrovine v stavovom priestore. Kvalitu procesov danú voľbou časovej konštanty T a parametrov asymptotickej hyperoviny q_0, $q_{1,...,}q_{n-2}$ zabezpečíme konštantnými parametrami spätných väzieb od stavových veličín k_{0c}, $k_{1c},...,$ $k_{(n-1)c}$.

Rovnicu (9) upravíme do tvaru:

$$x^{(n)} + \left(\frac{1}{T} + q_{n-2}\right)x^{(n-1)} + ... + \left(\frac{q_1}{T} + q_0\right)x' + \frac{q_0}{T}x = 0 \qquad (14)$$

Porovnaním koeficientov v diferenciálnych rovniciach (8) a (14) dostaneme vzťahy pre výpočet konštantných parametrov k_{ic}:

$$k_{0c} = \frac{1}{b}\left(\frac{q_0}{T} - a_0\right);\quad k_{1c} = \frac{1}{b}\left(\frac{q_1}{T} + q_0 - a_1\right);\ ...$$
$$k_{(n-1)c} = \frac{1}{b}\left(\frac{1}{T} + q_{n-2} - a_{n-1}\right) \qquad (15)$$

Tým vzniklá nová metóda syntézy stavového riadenia lineárneho systému s konštantnými parametrami.

2. Časovo optimálne riadenie systému s premenlivou štruktúrou

Stavové riadenie s konštantnými parametrami riadiaceho systému k_{ic} sa vyznačuje nedostatkami typickými pre časovo invariantné lineárne systémy riadenia. V každom dynamickom stave je systém rovnako tlmený. Väčšiu variabilitu dynamiky dosiahneme v systémoch stavového riadenia s premenlivou štruktúrou.

Budeme hľadať taký algoritmus zmeny variabilných zložiek parametrov riadiaceho systému k_{iv}, ktorý zabezpečí časovo optimálne procesy. Formulácia úlohy časovo optimálneho riadenia lineárneho systému zmenou parametrov s cieľovým stavom v rovnovážnom stave nie je korektná. Spojitý lineárny systém, riadený zmenou parametrov, teoreticky nemôže prejsť do rovnovážneho stavu za konečný čas. Budeme preto riešiť inú úlohu, a to previesť riadený objekt za minimálny čas z ľubovoľného stavu na hyperrovinu v stavovom priestore, ktorá obsahuje rovnovážny stav. Ako cieľovú hyperrovinu zvolíme asymptotickú hyperrovinu opísanú vzťahom (11).

Predpokladáme, že konštantné zložky koeficientov riadiaceho systému k_{ic} sú navrhnuté tak, aby charakteristická rovnica (7) mala reálny záporný koreň. Variabilné zložky koeficientov riadiaceho systému k_{iv} sa budú meniť v rozsahu vyjadrenom vzťahom (4), resp. (5). Za týchto predpokladov a s použitím vzťahov (8) a (9) upravíme diferenciálnu rovnicu (6) do tvaru

$$\frac{d\sigma}{dt} = -\frac{1}{T}\sigma - \sum_{i=1}^{n} bk_{iv}x^{(i-1)} \qquad (16)$$

Z rovnice (16) pre $\sigma \neq 0$ vyplýva podmienka kĺzavého režimu a dosiahnuteľnosti asymptotickej hyperoviny

$$\sigma\frac{d\sigma}{dt} = -\frac{1}{T}\sigma^2 - \sigma\sum_{i=1}^{n} bk_{iv}x^{(i-1)} < 0 \qquad (17)$$

Diferenciálna rovnica (16) pri splnení podmienky (17) opisuje prechodný proces systému končiaci v stavovom priestore na asymptotickej hyperovine. Keďže ide o sústavu len prvého rádu, pre časovo optimálny proces je potrebné zabezpečiť zmenou variabilných zložiek parametrov k_{iv} maximálnu rýchlosť približovania sa stavu systému k hyprerovine. Z rovnice (17) vyplývajú pre variabilné zložky parametrov riadiaceho systému nasledovné podmienky časovo optimálneho procesu

$$\max_{k_{iv}} \sigma bk_{iv}x^{(i)} \quad \text{pre } i=0,1,\ldots,n-1 \qquad (18)$$

Pre $b>0$ z podmienky (18) pre rozsah zmien parametrov vyjadrený vzťahom (4), vyplýva algoritmus zmeny variabilných zložiek parametrov riadiaceho systému, ktorý zabezpečí časovo optimálny prechod systému z ľubovoľného stavu na asymptotickú hyperrovinu

$$k_{iv} = k_{iA}sign(\sigma x^{(i)}) = \frac{k_{i\max} - k_{i\min}}{2}sign\,\sigma.sign x^{(i)}$$

pre i=0,1,...,n-1 (19)

Zo vzťahov (18) a (19) vyplýva, že čím je väčší rozsah zmien variabilnej zložky parametrov spätných väzieb vyjadrený hodnotami veličín k_{iA}, tým rýchlejšie systém prejde na asymptotickú hyperrovinu. Algoritmus (19) zároveň zabezpečuje existenciu kĺzavého režimu na asymptotickej hyperrovine. Poznamenáme, že zväčšovaním hodnôt veličín k_{iA} zvyšuje sa rozsah možných zmien parametrov riadeného objektu, voči ktorým bude systém invariantný.

3. Realizácia stavového riadenia s premenlivou štruktúrou kvázi reléovým riadiacim systémom bez kĺzavého režimu

Systém riadenia s premenlivou štruktúrou podľa navrhnutého algoritmu môže byť realizovaný dvoma spôsobmi. Pri prvom spôsobe koeficienty spätných väzieb od stavových veličín k_i budú nadobúdať svoje hraničné hodnoty dané vzťahom (1). Parametre k_i, sa menia jednotlivo, ak sa mení znamienko príslušnej stavovej veličiny. Ak systém v stavovom priestore prechádza cez asymptotickú hyperrovinu, mení sa znamienko premennej σ. Vtedy sa menia súčasne hodnoty všetkých koeficientov k_i.

Výhodnejší je druhý spôsob, ktorý spočíva v rozčlenení riadiaceho systému na dva paralelné systémy: systém stavového riadenia s konštantnými koeficientmi k_{ic} a riadiaci systém s premenlivými parametrami k_{iv} , ktoré sa menia podľa algoritmu (19). V tomto prípade akčná veličina bude počítaná podľa vzťahu:

$$u = -\sum_{i=0}^{n-1}(k_{ic} + k_{iv})x^{(i)} = -\sum_{i=0}^{n-1}k_{ic}x^{(i)} - \sum_{i=0}^{n-1}k_{iA}\left|x^{(i)}\right|sign\,\sigma \qquad (20)$$

Vo vzťahu (20) prvý člen reprezentuje stavové riadenie s konštantnými parametrami a druhý riadiaci systém s premenlivými parametrami. Riadiaci systém s premenlivými parametrami v tomto prípade bol transformovaný do tzv. kvázi reléového systému - reléového systému s premenlivou saturáciou výstupného signálu.

Vzťah (20) bol odvodený pre referenčnú hodnotu w=0. V tomto prípade veličina σ sa počíta podia vzťahu (9). Pre referenčnú veličinu w=const≠0 vzťah (20) upravíme do tvaru

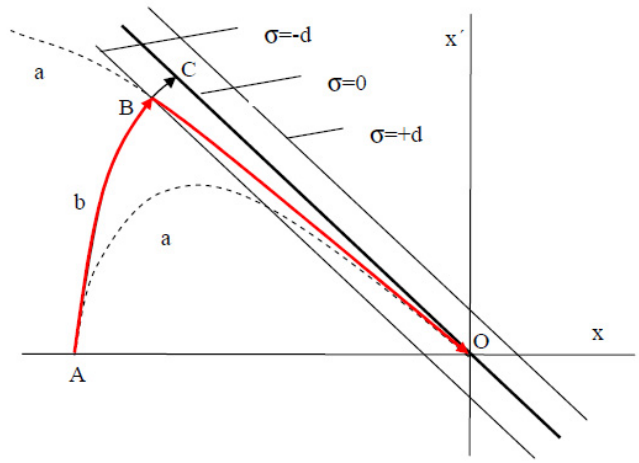
$$u = k_{0c}(w-x) - \sum_{i=1}^{n-1}k_{ic}x^{(i)} + \left(k_{0A}|w-x| - \sum_{i=1}^{n-1}k_{iA}\left|x^{(i)}\right|\right)sign\,\sigma \qquad (21)$$

kde odchýlka od hyperroviny σ sa počíta podľa vzťahu

$$\sigma = q_0(x - w) + q_1 x' + \ldots + q_{n-1} x^{(n-1)} \qquad (22)$$

Systém stavového riadenia s konštantnými parametrami sám o sebe zabezpečuje procesy asymptoticky sa blížiace v stavovom priestore k asymptotickej hyperrovine. Druhý riadiaci systém, ktorý zabezpečuje časovo optimálne procesy prechodu z ľubovoľného stavu na asymptotickú hyperovinu a realizuje na nej kĺzavý režim, je kvázi reléový. V takejto kombinácií dvoch paralelných riadiacich systémov veľmi jednoduchou úpravou možno odstrániť kĺzavý režim. Stačí okolo asymptotickej hyperoviny vymedziť úzke pásmo, v ktorom druhá časť algoritmu, t. j. algoritmus s premenlivou štruktúrou sa nebude uplatňovať.

Úprava spočíva v náhrade ideálneho reléového člena trojpolohovým reléovým členom s pásmom necitlivosti. Trajektórie prechodných procesov v stavovom priestore, pre systém druhého rádu v stavovej (fázovej) rovine, sú znázornené na obr. 2.



Obr.2 Trajektórie systémov stavového riadenia s konštantnými parametrami a s premenlivou štruktúrou

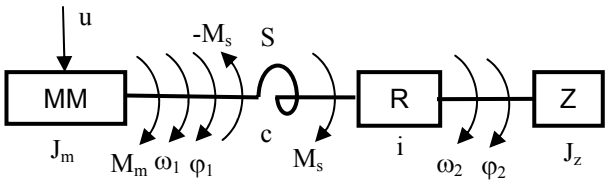
Fig. 2 State-space control systems trajectories with constant parameters and with variable structure

Asymtotická hyprerovina zodpovedá hodnote veličiny σ=0. Pásmo okolo asymtotickej hyperroviny, v ktorom sa uplatňuje len stavové riadenie s konštantnými parametrami, je vymedzené hyperrovinami σ=-d a σ=+d. Hodnotou veličiny *d* je určená šírka pásma na jednu i druhu stranu od hyperoviny. Krivky s označením *a* sú trajektórie systému stavového riadenia len s konštantnými parametrami a krivka s označením *b* je trajektória systému stavového riadenia zloženého zo systému s konštantnými a variabilnými parametrami. Krivka ABO je trajektória systému, v ktorom sa kombinuje stavové riadenie s premenlivou štruktúrou na úseku AB a stavové riadenie s konštantnými parametrami na úseku AO vo vymedzenom pásme okolo asymptotickej hyperoviny. Z obrázku je zrejme, že na úseku AB pri časovo optimálnom riadení premenlivých parametrov dochádza k zrýchleniu prechodného procesu v porovnaní s procesom pri stavovom riadení s konštantnými parametrami. Ak pásmo stavového riadenia s konštantnými parametrami nie je vymedzené (d=0), v takom prípade úsek trajektórie AB sa predlžuje až po bod C, kde nastáva kĺzavý režim. Ďalší pohyb v kĺzavom režime je po asymptotickej hyperrovine.

4. Príklad návrhu riadenia polohového servomechanizmu s pružným členom

Navrhneme riadenie polohového servomechanizmu s pružným členom – pružnou spojkou (Obr.3). Pohon servomechanizmu sa skladá z elektromotora s riadeným polovodičovým meničom energie MM, pružnej spojky S, prevodovky R a poháňaného telesa Z.

Metódu syntézy systému riadenia s premenlivou štruktúrou budeme ilustrovať na zjednodušenom modeli riadeného objektu. Budeme uvažovať dynamickú záťaž, vyjadrenú momentom zotrvačnosti motora J_m a momentu zotrvačnosti poháňaného telesa J_z. Statickú záťaž, moment zotrvačnosti prevodovky a tlmenie pružného člena zanedbáme. Akčný člen, t.j. menič s motorom budeme považovať za bezzotrvačný generátora hnacieho momentu M_m.



Obr. 3 Pohon s pružnou spojkou

Fig. 3 Drive with elactic coupling

Zostavíme matematický model riadeného objektu a upravíme ho do tvaru vhodného pre syntézu systému radenia s premenlivou štruktúrou navrhovanou metódou.

Pohybové rovnice mechanickej časti pohonu sú

$$J_m \frac{d\omega_1}{dt} = M_m - M_s \qquad (23)$$

$$J_z \frac{d\omega_2}{dt} = M_s \qquad (24)$$

kde
ω_1 je uhlová rýchlosť motora,
ω_2 – uhlová rýchlosť poháňaného telesa,
M_m – hnací moment motora,
M_s – moment pôsobiaci na spojke
J_m - moment zotrvačnosti motora
J_z – moment zotrvačností poháňaného telesa

Pre uhlové rýchlosti motora a poháňaného telesa a uhly potočenia motora φ_1 a poháňaného telesa φ_2 platia vzťahy

$$\frac{d\varphi_1}{dt} = \omega_1 \qquad (25)$$

$$\frac{d\varphi_2}{dt} = \omega_2 \qquad (26)$$

Moment motora vyjadríme vzťahom

$$M_m = K_m u \qquad (27)$$

kde
K_m je konštanta meniča a motora ako bezzotrvačného generátora momentu,
u – vstupná riadiaca veličina meniča.

Moment na spojke v závislosti od pootočenia hriadeľa motora a vstupného hriadeľa prevodovky vyjadríme vzťahom

$$M_s = c(\varphi_1 - i\varphi_2) \qquad (28)$$

kde
M_s je moment pôsobiaci na spojke,
c – konštanta pružnosti spojky,
i – prevodové číslo prevodovky.

Z rovníc (23) ÷ (28) môžeme odvodiť jednu diferenciálnu rovnicu, ktorou opíšeme celý objekt riadenia v tvare

$$\frac{d^4\varphi_2}{dt^4} + c\left(\frac{1}{J_m} + \frac{i}{J_z}\right)\frac{d^2\varphi_2}{dt^2} = \frac{cK_m}{J_m J_z}u \qquad (29)$$

Parametre objektu riadenia sú:

J_m = 0.002 kgm^2

J_z = 0.2 kgm^2

c=200 Nm/rad

i =10

K_m=2.5 Nm/V

Koeficienty diferenciálnej rovnice (30) budú:

a_4=1

a_3=a_1=a_0=0

a_2=c*(1/J_m+i/J_z)=11000

b=c*K_m/(J_m*J_z)=12500

Takto zjednodušený idealizovaný model riadeného objektu predstavuje sériové spojenie netlmeného kmitavého člena 2. rádu a dvoch integračných, teda astatických členov. Dosiahnuť dobrú kvalitu jeho riadenia štandardnými PID regulátormi je problematické. Vhodná je štruktúra stavového riadenia. Pre návrh stavového riadenia metódou asymptotickej hyperroviny zvolíme časovú konštantu

T=0.05 s

Koeficienty rovnice asymptotickej hyperroviny vypočítame podľa požiadaviek na kvalitu procesov na asymptotickej hyperrovine. Použili sme štandardný tvar charakteristickej rovnice:

s^3+2.5Ωs^2+2.5Ω^2s+Ω^3=0

Vlastnú frekvenciu systému volíme s ohľadom na fyzikálne obmedzenia a praktické požiadavky na dynamiku systému

Ω=20

Pre zvolenú hodnotu budú koeficienty rovnice asymptotickej hyperroviny:

q_0 =Ω^3=8000; q_1=2.5Ω^2=1000;
q_2=2.5Ω=50; q_3=1;

Podľa vzťahov (15) vypočítame konštantné parametre stavového riadenia LTI systému:

k_{0c}=(q_0/T-a_0)/b = 12.8000

k_{1c}=(q_1/T+q_0-a_1)/b = 2.2400

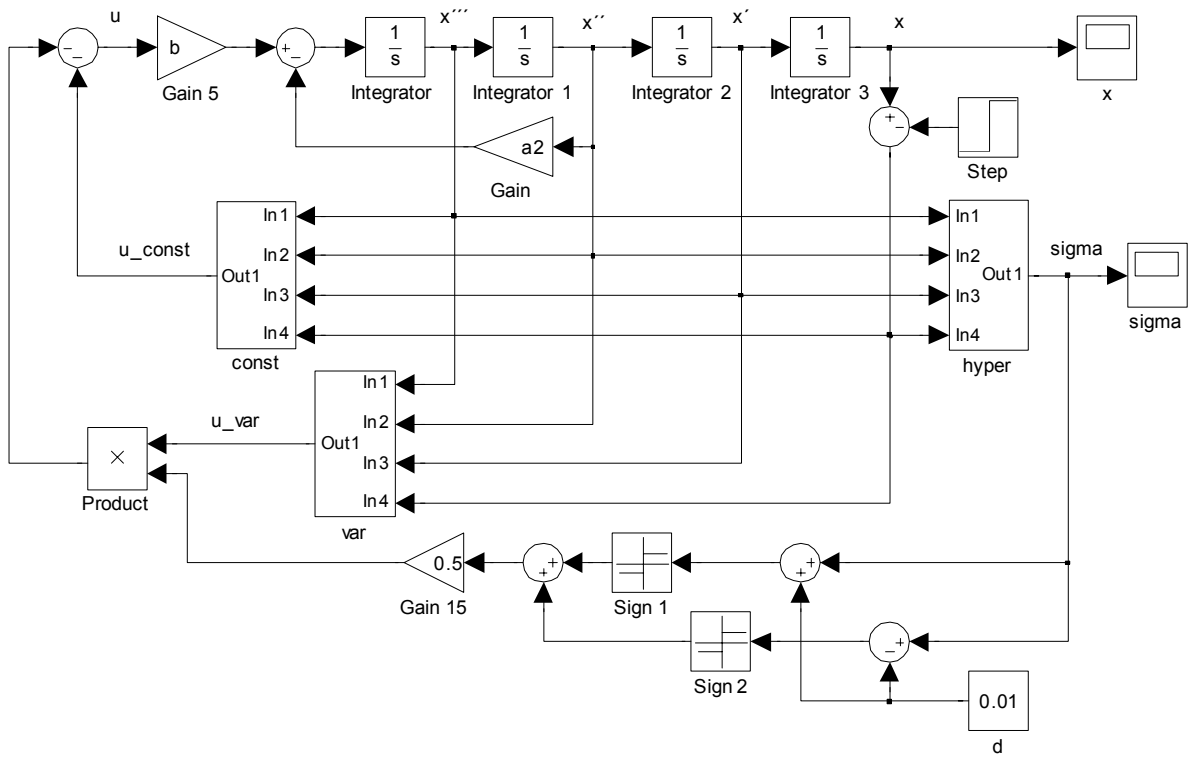
k_{2c}=(q_2/T+q_1-a_2)/b = -0.7200

k_{3c}=(q_3/T+q_2-a_3)/b = 0.0056

Rozsah zmien variabilných zložiek koeficientov stavového riadenia zvolíme tak, aby pri rozbehoch servomechanizmu systém nebol tlmený, preto zvolíme

k_{0A}=0; k_{1A}=k_{1c}=2.2400; k_{2A}=a_2/b+k_{2c}=0.1600;

k_{3A}=k_{3c}=0.0056



Obr. 4 Simulačný model systému riadenia s premenlivou štruktúrou
Fig. 4 Simulation model variable structure control system

Simulačný model systému riadenia s premenlivou štruktúrou je na obr.4. Na modeli sme označili výstupnú veličinu

$$x = \varphi_2$$

Subsystém *const* je riadiaci systém s konštantnými koeficientmi. Generuje riadiaci signál podľa vzťahu

$$u\_const = k_{0c}x + k_{1c}x' + k_{2c}x'' + k_{3c}x'''$$

Subsystém *var* generuje signál podľa vzťahu

$$u\_var = k_{0A}|x| + k_{1A}|x'| + k_{2A}|x''| + k_{3A}|x'''|$$

ktorý je ďalej násobený signálom z trojpolohového reléového člena realizovaného blokmi *Sign* a blokom nastavovania šírky pásma *d* okolo asymptotickej hyprerroviny. Veličina σ je generovaná subsystémom *hyper* podľa vzťahu

$$sigma = q_{00}(x-w) + q_{10}x' + q_{20}x'' + q_{30}x'''$$

Aby veličiny x, σ a d boli zobrazené na modeli v rovnakej mierke, koeficienty asymptotickej hyperroviny boli prepočítané z pôvodných koeficientov podľa vzťahov

$q_{00}=1$

$q_{10}=q_1/q_0= 0.1250$

$q_{20}=q_2/q_0= 0.0063$

$q_{30}=1/q_0 = 1.2500\text{e-}004$

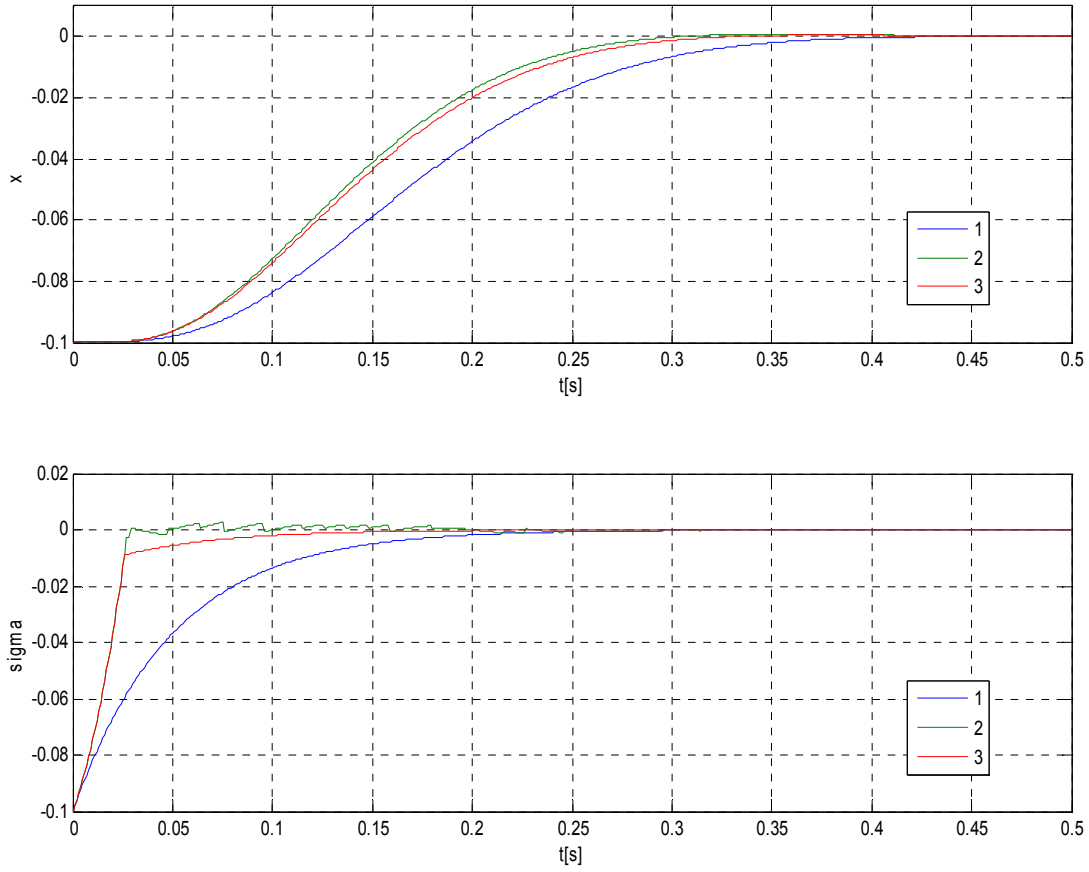
Výsledky simulácie sú zobrazene grafmi na obr. 5. pre *w*=0 a počiatočný stav [-0.1 0 0]. Na hornom obrázku sú časové priebehy výstupnej veličiny *x* a to:

1 - pre stavové riadenie s konštantnými parametrami,

2 – pre systém s variabilnou štruktúrou a kĺzavým režimom (d=0)

3- pre systém s variabilnou štruktúrou bez kĺzavého režimu (d=0.01)

Na dolnom obrázku sú časové priebehy odchýlky od asymptotickej hyperroviny σ, pre tie isté tri varianty systémov riadenia.



Obr. 5 Časové priebehy výstupnej veličiny x a odchýlky od hyperroviny σ

Fig. 5 Time response of output variable x and error from hypperplane σ

Záver

Systémy riadenia s premenlivou štruktúrou podobne, ako systémy stavového riadenia, vyžaduje úplnú informáciu o dynamickom stave riadeného objektu. Tá sa získava buď priamym meraním, alebo nepriamo pomocou pozorovateľov stavu. Z tohto hradiska je výhodné vychádzať pri syntéze systémov riadenia s premenlivou štruktúrou zo štruktúr stavového riadenia. Tento prístup bol použitý aj v našej práci. Bola vypracovaná metóda asymptotickej hyperroviny pre syntézu systémov stavového riadenia s konštantnými parametrami, ktorá ďalej viedla k pomerne jednoduchým postupom syntézy systémov s premenlivou štruktúrou a to ako pri syntéze systémov s kĺzavým režimom, tak systémov bez kĺzavého režimu. Simulačné experimenty ukázali, že v systémoch riadenia

s premenlivou štruktúrou bez kĺzavého režimu možno dosiahnuť takmer rovnakú kvalitu prechodných procesov, ako v systémoch s kĺzavým režimom. Kvalita prechodných procesov je značne lepšia ako v porovnateľných systémoch stavového riadenia s konštantnými parametrami. Ďalšou výhodou systémov riadenia s premenlivou štruktúrou bez kĺzavého režimu oproti systémom stavového riadenia s konštantnými parametrami je v tom, že v systémoch s premenlivou štruktúrou možno zabezpečiť robustnosť v oblasti ustálených stavov, bez toho, aby sa to negatívne prejavilo na celkovej dynamike prechodných procesov.

Poďakovanie

Táto práca vznikla na Katedre mechatronických systémoy Fakulty mechatroniky Trenčianskej univerzity Alexandra Dubčeka v Trenčíne v rámci výskumnej úlohy podporovanej Vedeckou grantovou agentúrou Ministerstva školstva Slovenskej republiky: VEGA 1/4056/07 - Analýza a syntéza mechatronických systémov

Literatúra

[1] BORŠČ, M., THURSKÝ, B., HRICKO, J.: Synthesis of Optimal and Robust Systems with Variable Control Structure. TnUAD 2010 (v tlači)

[2] BORŠČ, M., VITKO, A., THURSKÝ, B.: Optimal stabilization of modal control with variable structure. Transactions of the Institute of Measurement and Control, Vol. 30, No. 1. 63-76 (2008) ISSN: 0142-3312

[3] BOURMISTROVA A., STOREY I. AND SUBIC A.: Multiobjective Optimisation of Active and Semi-Active Suspension Systems with Application of Evolutionary Algorithm. International Conference on Modeling and Simulation, Melbourne, 12-15 December 2005.

[4] KADOŠ, J.: Theory of Variable Structure Systéme and the Time Sub-Optimal Position Control, Edition 1, HMH, Ltd., Bratislava 2007, ISBN 978-80-969725-0-0

[5] UTKIN, V. I.: First Stage of VSS: People and Events . In: Variable Structure Systems: Towards the 21st Century. Springer Verlag Berlin/Heidelberg, Vol 274 (2002), ISSN 0170-8643, p.p. 1-33

[6] UTKIN, V. I.: Sliding Modes in Control and Optimization, Springer Verlag, Berlin 1993.

[7] XU, J. X., ZHANG, J.: On Quasi-optimal Variable Structure Control Approaches. In: In: Variable Structure Systems: Towards the 21st Century. Springer Verlag Berlin/Heidelberg, Volume 274 (2002), ISSN 0170-8643, pp. 175-200

Abstract

In this paper is depicting the state-space control system with variable structure system design using asymptotic hyper-plane method. The control system is presented with two systems parallel connection. First control system is with constant parameters. Its design it is realized so that in state-space exist asymptotic hyper-plane towards whose converting trajectories of transition process and lest process on the asymptotic hyper-plane was wished quality. The second control system with variable structure that is transformed of the so-called quasi-relay. It's designed so that system transition from random state on asymptotic hyper-plane was time optimal and preserve on his sliding mode. The control system is in next modified so that was removed sliding mode and in order not to in the process was process quality substantially worse.

Michal Boršč, prof. Ing., CSc.

Branislav Thurský, Ing.

Jaroslav Hricko, Ing.

Trenčianská univerzita Alexandra Dubčeka v Trenčíne
Fakulta mechatroniky
Katedra mechatronických systémov
Pri parku 19
911 06 Trenčín
Tel.: 00421 32 7417544

E-mail borsc@tnuni.sk

E-mail thursky@tnuni.sk

E-mail hricko@tnuni.sk

Implementation of predictive control on industrial controllers

Eva Miklovičová, Marián Mrosko

Abstract
Model predictive control (MPC) has developed greatly over the last decades both within the research control community and in industry. Predictive algorithms are available in various commercial control packages but their implementation costs could be considerable. Automation of industrial processes is often realized using small industrial controllers that offer only simple control structures, like PID control loop. The aim of this paper is to employ the classical PID algorithm implemented on industrial computers to do advanced control without the necessity of the specialized software. The proposed PID control design procedure is verified by an application to a laboratory plant.

Keywords: predictive control, PID controller, PLC

Introduction

Model predictive control (MPC) refers to a family of advanced control methods which make explicit use of a model of the process to predict the future process behavior and to calculate a future control sequence minimizing an objective function [1]. The objective function is formulated as a combination of the set-point tracking performance and control effort. As predictive control belongs to the category of the open-loop optimization techniques, its implementation is based on the receding horizon strategy, i.e. only the first control signal of the future sequence is used at each sampling instant and the calculation is repeated in the next sampling time. This allows to incorporate a feedback into the control loop and to improve the control performances in the presence of disturbances and unmodelled dynamics.

First predictive control algorithms have been proposed at the end of the 1970s and quickly became popular and developed considerably over the last three decades both within the research control community and in industry. Their popularity is mainly due to the fact, that they can be used to control a great variety of processes including time-delayed systems or nonminimum phase or the unstable ones. The multivariable case can easily be dealt with as well. Another important feature is that the constraints can be systematically incorporated into the design procedure, which can influence the resulting control system performances and the process operation safety. MPC technology can now be found in a wide variety of application areas including petrochemical, chemical, food processing, automotive and aerospace industries.

Despite the wide development of advanced control methods, the PID controllers are still commonly used in industry for its structural simplicity and design rules of thumb. The PID control function can be found in many medium-size programmable logical controllers (PLC) and all large PLC. Such functions can be used directly by entering the parameters for given PID controller structure.

The aim of the paper is to derive the conditions for the equivalence of the MPC and the PID control implemented on simple programmable logical controller. More specifically, the PID control structure implemented on SIMATIC S7-200 programmable logical controller has been considered. The PID controller parameters are obtained by equating the discrete PID control law with the classical pole-placement control structure of generalized predictive control (GPC) given some conditions on the orders of the polynomials involved in the GPC control structure [3]. As these orders depend on the process model, the process model order is restricted to a maximum of two, the first order model results in a PI controller while a second order plant yields a PID structure. On the other hand, there is no restriction on the choice of GPC tuning parameters so that the advantages of model predictive control can fully be exploited.

Performance of the predictive PID scheme is shown to be equivalent to GPC by means of simulation. An application of the predictive PID algorithm for the control of a laboratory plant is also presented.

1. Predictive control

Generalized predictive control (GPC) developed in [2] belongs to the most popular predictive algorithms based on the parametric plant model. It can handle various control problems for a wide range of plants, its implementation is relatively simple and due to several design parameters it can be tuned to specific applications.

Consider that the operation of the single-input single-output (SISO) plant around the particular setpoint can be described by the following CARIMA model

$$A(z^{-1})y(t) = B(z^{-1})u(t-d-1) + v(t) \tag{1}$$

$$D(z^{-1})v(t) = C(z^{-1})\xi(t) \tag{2}$$

with

$$\begin{aligned} A(z^{-1}) &= 1 + a_1 z^{-1} + \ldots + a_{na} z^{-na} \\ B(z^{-1}) &= b_0 + b_1 z^{-1} + \ldots + b_{nb} z^{-nb} \\ C(z^{-1}) &= 1 + c_1 z^{-1} + \ldots + c_{nc} z^{-nc} \\ D(z^{-1}) &= 1 - z^{-1} \end{aligned} \tag{3}$$

where *u(t)* is the control variable, *y(t)* the measured plant output, *d* denotes the minimum plant model time-delay in sampling periods, *v(t)* represents the external disturbances and *ξ(t)* is the random variable with zero mean value and finite variance. For simplicity in the following the $C(z^{-1})$ polynomial is chosen to be 1.

The GPC control objective is to compute the future control sequence in such a way that the future plant output is driven close to the prescribed reference trajectory; this is accomplished by minimizing the following cost function

$$J(t,ph,ch,sh,\rho)=E\left\{\sum_{j=sh}^{ph}\left(\hat{y}(t+j/t)-y^*(t+j)\right)^2+\right.$$
$$\left.+\rho\left(D(z^{-1})u(t+j-sh)\right)^2\right\} \quad (4)$$

subject to:

$$D(z^{-1})u(t+i)=0 \text{ for } ch\le i\le ph \quad (5)$$

where *sh*, *ph* and *ch* are positive scalars defining the starting horizon, prediction horizon and control horizon, ρ is a nonnegative control weighting scalar. $\hat{y}(t+j/t)$ denotes the j-step ahead prediction of *y(t)* based on the data available up to time t and $y^*(t+j)$ is the future reference trajectory, that can be generated as an output of the reference model of the form

$$A_m(z^{-1})y^*(t+d+1)=B_m(z^{-1})u^*(t) \quad (6)$$

The j-step ahead predictor is expressed as follows

$$\hat{y}(t+j/t)=G_{j-d}(z^{-1})D(z^{-1})u(t+j-d-1)+y_0(t+j/t)$$
$$y_0(t+j/t)=H_{j-d}(z^{-1})D(z^{-1})u(t-1)+F_j(z^{-1})y(t) \quad (7)$$

where the polynomials $F_j(z^{-1}), G_{j-d}(z^{-1}), H_{j-d}(z^{-1})$ are solutions of the following Diophantine equations

$$1=A(z^{-1})D(z^{-1})E_j(z^{-1})+z^{-j}F_j(z^{-1}) \quad (8)$$

$$E_j(z^{-1})B(z^{-1})=G_{j-d}(z^{-1})+z^{-j+d}H_{j-d}(z^{-1}) \quad (9)$$

The cost function (4)-(5) may be rewritten in the suitable vector form

$$J(t,ph,ch,sh,\rho)=\left(G_1U(t+ch-1)+Y_0(t)-Y^*(t+ph)\right)^T$$
$$\left(G_1U(t+ch-1)+Y_0(t)-Y^*(t+ph)\right)$$
$$+\rho U(t+ch-1)^T U(t+ch-1) \quad (10)$$

where

$$Y^*(t+ph)=\left[y^*(t+d+1),\ldots,y^*(t+ph)\right]^T$$
$$Y_0(t)=\left[y_0(t+d+1/t),\ldots,y_0(t+ph/t)\right]^T$$
$$U(t+ch-1)=\left[D(z^{-1})u(t),\ldots,D(z^{-1})u(t+ch-1)\right]^T$$

$$G_1=\begin{bmatrix} g_{sh-d-1} & \cdots & g_0 & 0 & 0 & 0 \\ g_{sh-d} & \cdots & \cdots & g_0 & 0 & 0 \\ \vdots & \cdots & \cdots & \cdots & \ddots & \vdots \\ g_{ch-1} & \cdots & \cdots & \cdots & \cdots & g_0 \\ \vdots & \cdots & \cdots & \cdots & \cdots & \vdots \\ g_{ph-d-1} & \cdots & \cdots & \cdots & \cdots & g_{ph-ch-d} \end{bmatrix} \quad (11)$$

The vector that minimizes the cost function (10) is given by

$$U(t+ch-1)=-\left[G_1^TG_1+\rho I_{ch}\right]^{-1}G_1^T\left(Y_0(t)-Y^*(t+ph)\right) \quad (12)$$

The GPC control law is implemented in a receding horizon sense, which means that only the first component of the optimal vector $U(t+ch-1)$ is taken into account and the optimization process is repeated in the next sampling period

$$D(z^{-1})u(t)=-\sum_{j=sh}^{ph}\gamma_j\left(y_0(t+j/t)-y^*(t+j)\right) \quad (13)$$

where the coefficients γ_j for $j=sh,\ldots,ph$ are the first line components of the matrix $\left[G_1^TG_1+\rho I_{ch}\right]^{-1}G_1^T$.

The control law (13) may also be implemented using a standard pole-placement control structure (shown in Fig. 1)

$$S(z^{-1})D(z^{-1})u(t)+R(z^{-1})y(t)=T(z^{-1})y^*(t) \quad (14)$$

where

$$R(z^{-1})=\sum_{j=sh}^{ph}\gamma_jF_j(z^{-1})=r_0+r_1z^{-1}+\ldots+r_{nr}z^{-nr} \quad (15)$$

$$S(z^{-1})=1+\sum_{j=sh}^{ph}\gamma_jz^{-1}H_{j-d}(z^{-1})=1+s_1z^{-1}+\ldots+s_{ns}z^{-ns} \quad (16)$$

$$T(z^{-1})=\sum_{j=sh}^{ph}\gamma_jz^{-ph+j}=t_0+t_1z^{-1}+\ldots+t_{nt}z^{-nt} \quad (17)$$

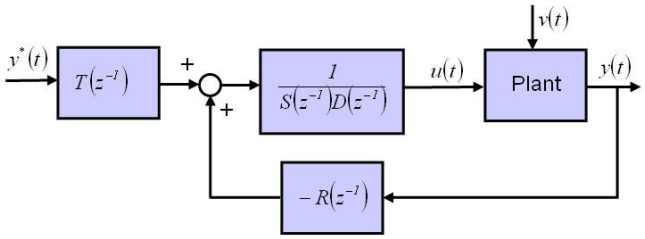


Fig. 1 Standard pole-placement control structure

2. PID algorithm implemented on PLC

Advanced predictive algorithms take part of several commercial control packages [4]. However, control of many industrial processes is done through the use of small computers called the programmable logic controllers (PLCs), where hardware and software are specifically adapted to industrial environment. This type of controllers usually offers only simple control structures, such as on-off control or PID control loops.

The Siemens SIMATIC S7-200 series is a line of micro-programmable logic controllers that can control a variety of small applications. The PID control law implemented on this PLC is of the form [5]

$$u(t)=M_P(t)+M_I(t)+M_D(t) \quad (18)$$

with

$$M_P(t)=K_ce(t) \quad \text{is the proportional term} \quad (19)$$

$$M_I(t)=K_c\frac{T_s}{T_i}e(t)+M_X \quad \text{is the integral term} \quad (20)$$

$$M_D(t)=K_c\frac{T_d}{T_s}\left(e(t)-e(t-1)\right) \quad \text{is the differential term}$$

where $e(t)=w(t)-y(t)$ is the control error and $w(t)$ denotes the value of setpoint. The bias M_X is the running sum of all previous values of the integral term. T_s represents the sam-

ple time. K_c, T_i and T_d are the loop gain, derivative time constant and integral time constant, respectively, that have to be specified.

To avoid step changes or bumps in the output due to derivative action on setpoint changes, this equation is modified to assume that the setpoint is a constant ($w(t)=w(t-1)$). This results in the calculation of the change in the process variable instead of the change in the error

$$M_D(t)=K_c\frac{T_d}{T_s}\left(w(t)-y(t)-w(t)+y(t-1)\right)=K_c\frac{T_d}{T_s}\left(y(t-1)-y(t)\right) \tag{21}$$

The resulting PID control structure is depicted in Figure 2.

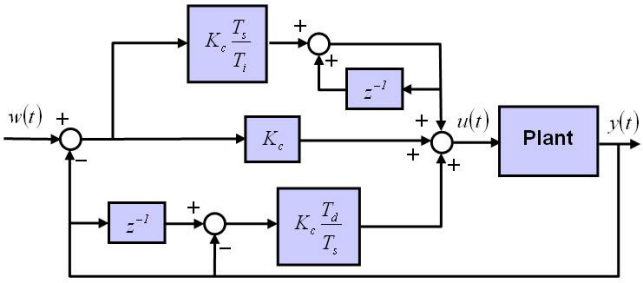


Fig. 2 PID control structure

3. Predictive PID control

The aim is to implement the predictive control algorithm by means of the above described PID control structure. Let us compare the pole placement control law

$$S\left(z^{-1}\right)D\left(z^{-1}\right)u(t)=T\left(z^{-1}\right)y^*(t)-R\left(z^{-1}\right)y(t) \tag{22}$$

to the incremental form of the PID control law (18) – (21)

$$\begin{aligned}D\left(z^{-1}\right)u(t)&=u(t)-u(t-1)=\\&=\left(K_c+K_c\frac{T_s}{T_i}\right)w(t)-K_cw(t-1)+\\&+\left(-K_c-K_c\frac{T_s}{T_i}-K_c\frac{T_d}{T_s}\right)y(t)+\\&+\left(K_c+2K_c\frac{T_d}{T_s}\right)y(t-1)-K_c\frac{T_d}{T_s}y(t-2)=\\&=P_w\left(z^{-1}\right)w(t)-P_y\left(z^{-1}\right)y(t)\end{aligned} \tag{23}$$

where

$$P_w\left(z^{-1}\right)=\left(K_c+K_c\frac{T_s}{T_i}\right)-K_cz^{-1} \tag{24}$$

$$P_y\left(z^{-1}\right)=\left(K_c+K_c\frac{T_s}{T_i}+K_c\frac{T_d}{T_s}\right)+\left(-K_c-2K_c\frac{T_d}{T_s}\right)z^{-1}+K_c\frac{T_d}{T_s}z^{-2} \tag{25}$$

and in case of PI control law

$$P_y\left(z^{-1}\right)=\left(K_c+K_c\frac{T_s}{T_i}\right)-K_cz^{-1} \tag{26}$$

Assuming that $w(t)=y^*(t)$ it yields that

$$S\left(z^{-1}\right)=1 \tag{27}$$

$$R\left(z^{-1}\right)=P_y\left(z^{-1}\right) \tag{28}$$

$$T\left(z^{-1}\right)=P_w\left(z^{-1}\right) \tag{29}$$

i.e. the GPC control law is equivalent to the PID control law if the $T\left(z^{-1}\right)$ polynomial is of the first order and the $R\left(z^{-1}\right)$ polynomial is of the second order. In case of PI control law $R\left(z^{-1}\right)$ has to be the first order polynomial. These conditions can be satisfied by the proper choice of the plant model structure (1), namely the second order plant model ($na=2,\ nb=0$) for PID control and the first order plant model ($na=1,\ nb=0$) for PI control, respectively.

The PID tuning constants K_c, T_i and T_d can be derived from equation (28)

$$K_c=-r_1-2r_2 \tag{30}$$

$$T_i=-\frac{r_1+2r_2}{r_0+r_1+r_2}T_s \tag{31}$$

$$T_d=\frac{-r_2}{r_1+2r_2}T_s \tag{32}$$

As it can be seen from the equations (30) – (32), the PID tuning constants depend only on the parameters of the $R\left(z^{-1}\right)$ polynomial. According to (15), the calculation of these parameters necessitates the knowledge of the G_1 matrix and $F_j\left(z^{-1}\right)\ j=sh,\dots ph$ polynomials.

The coefficients of G_1 matrix (11) can be obtained from the samples of plant step response, i.e. for the unit input step

$$\{u(0),u(1),u(2),\dots\}=\{1,1,1,\dots\} \tag{33}$$

the plant output signal is of the form

$$\{y(0),y(1),y(2),y(3),\dots\}=\{0,g_0,g_1,g_2,\dots\} \tag{34}$$

The $F_j\left(z^{-1}\right)\ j=sh,\dots ph$ polynomials are solutions of the Diophantine equations (8) with

$$\begin{aligned}E_j(z^{-1})&=e_0+e_1z^{-1}+\dots+e_{j-1}z^{-j+1}\\F_j(z^{-1})&=f_0^j+f_1^jz^{-1}+\dots+f_{na}^jz^{-na}\end{aligned} \tag{35}$$

and they can be calculated recursively as follows

$$E_1\left(z^{-1}\right)=1 \tag{36}$$

$$F_1\left(z^{-1}\right)=z\left(1-A\left(z^{-1}\right)D\left(z^{-1}\right)\right) \tag{37}$$

$$\begin{aligned}E_j\left(z^{-1}\right)&=E_{j-1}\left(z^{-1}\right)+f_0^{j-1}z^{-j+1}\\F_j\left(z^{-1}\right)&=z^j\left(1-E_j\left(z^{-1}\right)A\left(z^{-1}\right)D\left(z^{-1}\right)\right)\end{aligned} \quad \text{for} \quad j=2,\dots,ph \tag{38}$$

In order to compare the performances of the proposed PID control loop to those of the GPC control scheme a simulation of the second order plant

$$G(s)=\frac{1.308s+2.702}{s^2+0.5108s+0.6756} \tag{39}$$

has been performed with the sample time $T_s=1s$. At the time 60 s a step disturbance of 0.1 has been added to the plant output. The control design parameters are summarized in Table 1.

GPC	ch=2	ph=15	ρ=1
PID	K_c=0,2102	T_i=0,7777	T_d=1,3241

Table 1: Parameter values

As it can be seen from Fig. 3 the plant output and the control signal plots for both control schemes are identical.

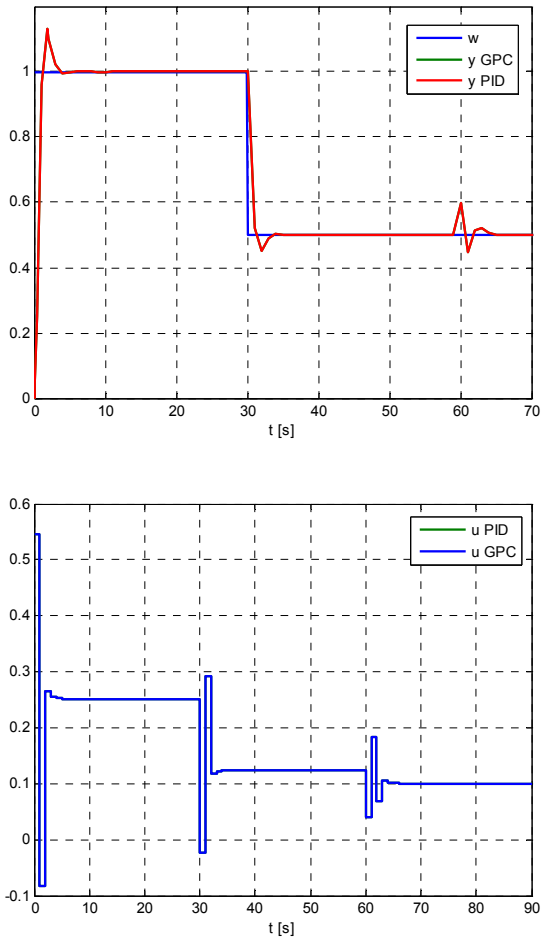


Figure 3 Comparison of GPC and PID control performances

4. Experimental evaluation

In order to verify the proposed predictive PID control design the control of simple cylindrical laboratory tank using the Simatic S7-200 has been realized. Because the GPC synthesis is based on the plant model transfer function, it is necessary to identify the ARX model of this nonlinear system around the operating point using the least-squares method. The system input is the inflow valve opening and the system output is tank level. The tank has also the outflow valve which has been used to generate a disturbance. In Figure 4 the time response of the real system is compared to the identified model time response round the operating point. The identified model transfer function is as follows:

$$G(z^{-1}) = \frac{0.01122z^{-1}}{1 - 1.207z^{-1} + 0.2146z^{-2}}, \quad T_{VZ} = 1s \qquad (40)$$

Based on the plant model (40) the predictive PID controller has been designed according to the section 3. Table 2 summarizes the control design parameters and the resulting PID constants.

Time response of tank level and desired value are in Figure 5. The time response of the manipulated variable and disturbance has also been monitored (Figure 6).

GPC	ch=1	ph=10	ρ=1
PID	K_c=4,6121	T_i=6,7646	T_d=0,2654

Table 2: Parameter values

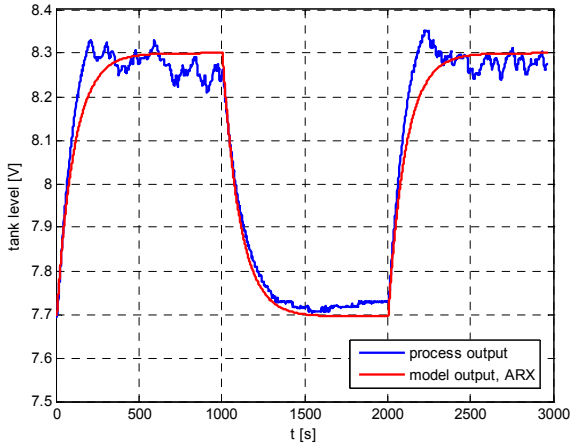


Figure 4 System identification

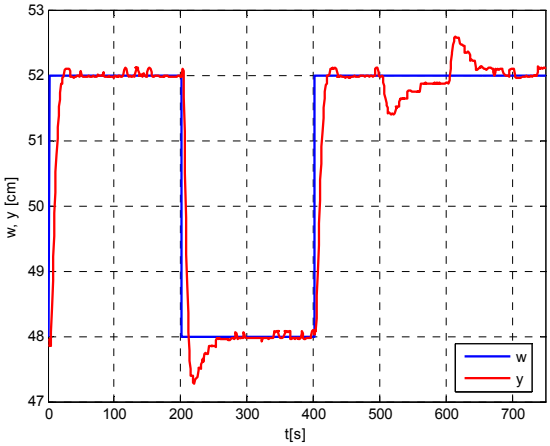


Figure 5 Time response of tank level

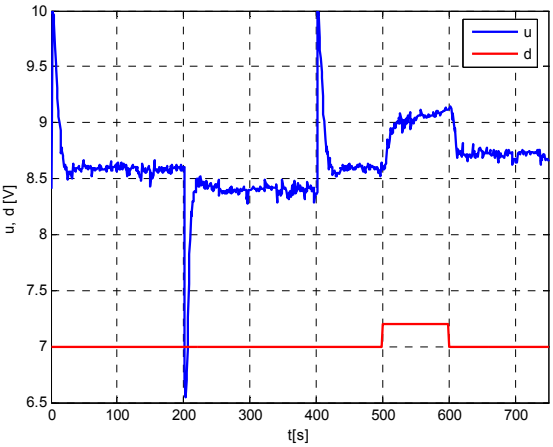


Figure 6 Time response of manipulated variable and disturbance

Conclusion

In this paper the PID control design based on predictive control approach has been proposed so that the benefits of predictive control can be taken even if simple industrial controllers are used. More specifically, the PID control structure implemented on programmable logical controller SIMATIC S7-200 has been considered. Another advantage is that using the proposed method, parameters of discrete form of PID controller are directly provided. Calculation of PID controller parameters is simple also for practitioners.

Acknowledgement

This work has been supported by the Slovak Scientific Grant Agency, Grant No. 1/0592/10.

References

[1] CAMACHO E.F., BORDONS, C. (2004): Model predictive control. Springer-Verlag, London.

[2] CLARKE,. D.W., MOHTADI, C., TUFFS, P.S. (1987): Generalized predictive control – Part I. The basic algorithm. Part II. Extensions and interpretations. Automatica, 23, 137-160.

[3] MILLER, R.M., SHAH, S.L., WOOD R.K., KWOK, E.K. (1999): Predictive PID, ISA Transactions, 38, 11-23.

[4] QIN, S.J., BADGWELL, T.A. (2003): A survey of industrial model predictive control technology, Control Engineering Practice, 11, 733-764.

[5] S7-200 Programmable Controller System Manual, Edition 05/2003.

doc. Ing. Eva Miklovičová, PhD.

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3
812 19 Bratislava
eva.miklovicova@stuba.sk

Ing. Marián Mrosko

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3
812 19 Bratislava
marian.mrosko@stuba.sk

Simulation Methods for Bottleneck Detection in Manufacturing Systems

Michal Leporis, Zdenka Králová

Abstract
The paper presents a comparison of several methods for production line bottleneck analysis using discrete event simulation approach and the simulation software WITNESS. An experimental environment has been created for processing and comparison of results obtained from the WITNESS simulation experiments. The advantages and constraints of the cited methods are discussed.

Keywords: production system, bottleneck detection, simulation, WITNESS

Introduction

The term “bottleneck” is used to describe a point of congestion in any system from computer networks to a factory assembly line. In such a system, there is always some process, task, machine, etc. that is the limiting factor preventing a greater throughput and thus determines the capacity of the entire system. Knowing the bottleneck allows increasing the flow by improving just one process in the system rather than all its remaining parts. Vice versa, if there is a bottleneck, nothing done elsewhere in the value stream can improve the throughput (Goldratt, Cox, 1984).

Both theory and practice of production management pay great attention to the bottleneck analysis in order to increase throughput of a production system, i.e. the rate at which the system generates money through sales of its products.

The bottleneck in production system occurs when workloads arrive at a given point more quickly than that point can handle them. The bottleneck situation causes unneeded inventory and prolongs manufacturing lead times. In a wider sense of the word, any element of a production system (machine, conveyor, AGV, buffer, labor etc.) can turn to a bottleneck.

As a result of the bottleneck analysis, particular recommendations can be drawn to improve the production system in the most effective way, significantly increasing its throughput and capacity.

BOTTLENECK DETECTION METHODS

Detecting a bottleneck in a production system is not a trivial task. Current bottleneck detection methods can be separated into two categories: analytical and simulation-based. A special conception of bottleneck detection has been developed based on evaluation of the real-time data from the manufacturing system (Li et al., 2007).

For analytical methods, the system performance is assumed to be described by a statistical distribution. Although an analytical model is suitable for long term prediction, this type of model is not adequate for solving problems of short term bottleneck detection.

For real production processes with complex structure and dynamics the analytical approach is practically inapplicable; in such a case, simulation-based methods seem to be more useful. Although creation of an adequate simulation model of a system is time-consuming, results of simulation experiments provide sufficient information enabling to detect a bottleneck. Advanced simulation tools offer complete statistics about the average utilization, waiting, blocking, breakdown etc. for each element of the model as results of the experiments. Other useful data can be obtained using special procedures. Furthermore, a simulation model can help identify the possibilities for system improvements and verify their impact on the overall system performance.

Leading companies, especially those operating in the automotive industry, have developed their own software tools for the manufacturing system bottleneck analysis and identification based on simulation models; e.g. General Motors Corporation created an internal throughput-analysis tool called C-MORE, which is a combination of decomposition-based analytical methods and customized discrete-event-simulation solvers. TOYOTA Central Research and Development Laboratories implemented their bottleneck detection methods into the software tool GAROPS Analyzer (Roser et al., 2001).

Bottleneck analysis algorithmization requires definition of an explicit criterion for bottleneck detection and a suitable method for transformation the obtained set of simulation results into a particular indicator. This provides a possibility to unify approaches to the bottleneck analysis for various types of production systems.

Simulation methods for bottleneck detection differ with respect to the criterion and the way of transformation of the simulation results to the values of the criterion.

This paper deals with four bottleneck detection methods developed over the last decade:

- Active period method (Roser et al., 2001)
- Turning point method (Li et al., 2007),
- Arrow-based method (Biller et al., 2008),
- Criticality indicators based method (Králová, Bielak, 2004).

Active period method

The active period method developed by (Roser et al., 2001) at Toyota Central Research and Development Laboratories is based on the analysis of machine status information determining periods during which a machine is active without interruption (Figure 1). Five distinct states are recognized for each machine: Working, Waiting, Blocked, Tool Change and Under Repair. For analysis, Waiting and Blocked are considered inactive. Active periods are occasionally interrupted by inactive periods during which the machine is waiting for the arrival of parts (Waiting) or for their removal (Blocked). The term "Machine" includes any element performing activity, e.g. machine, conveyor, AGV, etc.

The machine with the longest average active period is considered to be a bottleneck, as this machine is least likely to be interrupted by other machines, and in turn is most likely to dictate the overall system throughput.

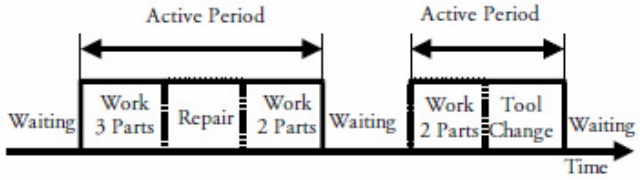


Fig. 1 Illustration of the active periods of machine during the simulation run

This method does not use the summary statistics obtained as a result of a simulation experiment, but is based on the analysis of the log file recording the relevant data about the events occurred during the simulation run (start and finish of the operation, repair, tool change, etc.).

Turning point method

A data driven bottleneck detection method proposed in (Li, et al., 2007) detects bottlenecks using the term "turning point". A turning point is defined to be the machine where the trend of blockage and starvation changes from blockage being higher than starvation to starvation being higher than blockage. Furthermore, the sum of a "turning point" machine's blockage and starvation is smaller than for its neighboring machines. Thus, the "turning point" machine has the highest percentage of the sum of operating time and downtime compared to other machines in the segment.

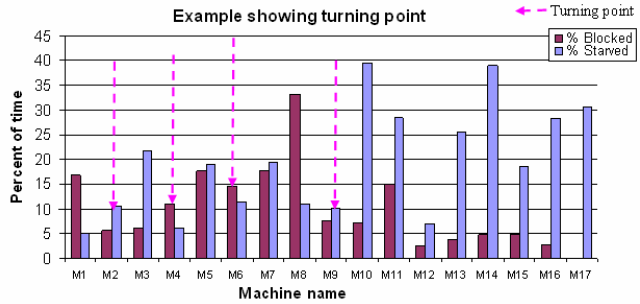


Fig. 2 Example of turning points determination

The j-th machine is the turning point in a n-machine segment with finite buffers if

$$(TB_i - TS_i) > 0 : i \in [1,\ldots, j-1],\ j \neq 1, j \neq n$$
$$(TB_i - TS_i) < 0 : i \in [j+1,\ldots, n],\ j \neq 1, j \neq n$$
$$TB_j + TS_j < TB_{j-1} + TS_{j-1},\ j \neq 1, j \neq n$$
$$TB_j + TS_j < TB_{j+1} + TS_{j+1},\ j \neq 1, j \neq n$$

$$\text{If } j=1\text{: } (TB_1 - TS_1) > 0 \ \&\ (TB_2 - TS_2) < 0 \ \&\ TB_1 + TS_1 < TB_2 + TS_2$$
$$\text{If } j=n\text{: } (TB_{n-1} - TS_{n-1}) > 0 \ \&\ (TB_n - TS_n) < 0 \ \&\ TB_n + TS_n < TB_{n-1} + TS_{n-1}$$

where TB_j is the blockage time for the j-th machine;

TS_j is the starvation time for the j-th machine;

j-1 is the index of the nearest upstream machine and

j+1 is the index of the nearest downstream machine.

According to (Li et al., 2007) both the analytical and the simulation-based verification was carried out. It has been proved that the turning point method can provide quick bottleneck identification.

Arrow-based method

The method described in (Biller et al., 2008) is built on the concept of (Kuo et al., 1996, 2008) who proposed an indirect method of bottlenecks identification for open serial lines. The Arrow-based method detects the bottlenecks in longer lines arranging the probabilities of starvations (*STi*) and blockages (*BLi*) for each machine as shown in Figure 3 and placing arrows directed from one machine to another according to the following rules:

if $BLi > STi+1$, assign the arrow pointing from *mi* to *mi+1*

if $BLi < STi+1$, assign the arrow pointing from *mi+1* to *mi*

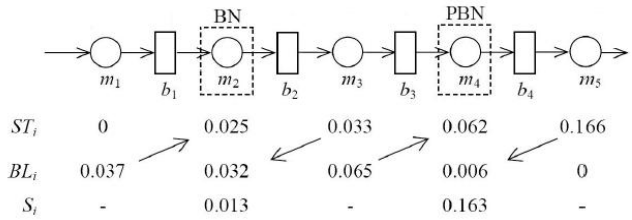


Fig. 3 Example of the bottleneck identification in the open serial line

A single machine with no emanating arrows is the bottleneck. If there are multiple machines with no emanating arrows (as in Figure 3), the one with the largest severity is the primary bottleneck.

The severity of the bottleneck is defined as

$$S_1 = |ST_2 - BL_1| \quad (1)$$

$$S_i = |ST_{i+1} - BL_i| + |ST_i - BL_{i-1}| \quad \text{pre } i = 2,\ldots,M-1 \quad (2)$$

$$S_M = |ST_M - BL_{M-1}| \quad (3)$$

This method was modified by (Biller, et al., 2008) as a two-stage procedure for serial lines with rework loops.

Criticality indicators based method

The approach described in (Králová, Bielak, 2004) is based on the evaluation of the so-called "criticality indicator" for each workplace and comparison of the indicator values to detect the critical place. The way of the indicator evaluation allows finding the critical places, including the bottlenecks needing the capacity expansion, as well as "the reserves" – workplaces allowing a better utilization.

For the i-th workplace KR_i, the criticality indicator is calculated from the simulation statistics considering the differences of the individual rates for this workplace (the average rates of utilization, starvation, blocking, waiting for labor) with respect to the whole-system average of this rate.

The integrated approach to the evaluation of criticality is based on aggregating the indicators of all related indices

into one value. The particular criticality indicator for each workplace is created by summing deviations of statistical indicators for the workplaces from the mean values of indicators for all workplaces in the system. The aggregated indicator is a function of the rates: busy, blocking, waiting for parts and waiting for labor. A graphical presentation in MS Excel provides a summary of system bottlenecks and reserves and their relationships as well.

KR_i is calculated by the formula:

$$KR_i = \left(\frac{\sum_{i=1}^{n} B_i}{n} - B_i \right) + \left(I_i - \frac{\sum_{i=1}^{n} I_i}{n} \right) + \left(Bl_i - \frac{\sum_{i=1}^{n} Bl_i}{n} \right) + \left(L_i - \frac{\sum_{i=1}^{n} L_i}{n} \right)$$

where:

- KR_i – the criticality indicator for the i-th workplace [%]
- *Bi* – the average utilization rate for the i-th machine (Busy) [%]
- *Ii* – the average starvation rate for the i-th machine (Idle) [%]
- *Bli* – the average blocking rate for the i-th machine (Blocked) [%]
- *Li* – the average waiting rate for labor for the i-th machine (Labor) [%]

The workplace with the minimal value of KRi is regarded as a bottleneck, the workplace with the maximal value of KRi as a maximal capacity reserve.

WITNESS SIMULATION SOFTWARE

WITNESS is a comprehensive discrete event and continuous process simulator. It is designed to model dynamics of complex systems. It is an established simulation tool for analysis and validation of business process to achieve a desired process performance or to support continuous process improvement activities used by thousands of companies worldwide (Markt, Mayer, 1999).

WITNESS provides a graphical environment to build simulation models. It enables to represent a real world process in a dynamic animated computer model and allows automating simulation experiments, optimizing material flow across the facility and generating animated models. A simulation model allows incorporating all the variability of real life experience (variable reliability, process times, resource efficiency etc.).

The WITNESS simulation package is capable of modeling a variety of discrete (e.g., part-based) and continuous (e.g. fluids and high-volume fast-moving goods) elements. Depending on the type of element, each can be in any of a number of states; these states can be idle (waiting), busy (processing), blocked, in-setup, broken down, waiting labor (cycle/setup/repair) etc.

Complex routing and control logic is achieved with numerous input and output rules as well as special actions using functions. The format for using actions is similar to that of a simple programming language.

Results of simulation can be viewed on the screen either in tabular or graphic format. In addition, several graphical elements are available for summarizing statistics from a model. Pie charts, time-series and histograms provide a meaningful, easy-to-read format for data from a simulation run. Reports allow user to examine the performance of elements in the model and provide him with relevant information about their interaction, details and status. Reports can help to identify areas where the model's operation can be improved.

WITNESS Optimizer provides a plug-in module which can intelligently test different combinations of changes within a model and carry out the desired experimentation

COMPARISON OF BOTTLENECK DETECTION METHODS

An experimental environment for processing the results obtained from WITNESS simulation experiments allows comparing effectiveness and limitations of the cited methods. The environment consists of the discrete event simulation model in WITNESS and the MS Excel user interface allowing setup of input data of the model and viewing analysis results for each method according to its criterion. The analysis is based on the simulation statistics about the resource utilization, starvation, blocking, waiting for labor, set-up and breakdown characteristics and on the evaluation of the relationships between the downstream and upstream activities. Serial recording of the important events such as operation start and finish, machine repair, tool change, etc. is assured via procedures in the input/output rules in the WITNESS model.

The basic experimental model represents a serial production line with thirty workplaces with buffers. Other variants of the model differ from the basic one by material flow branching and connection (Figure 4) and by rework loops. The simulation models were extended by the element "Labor" to enable verifying the capability to detect a labor as a bottleneck Values of the relevant parameters, such as machine cycle times, breakdown frequency, repair and setup times, etc. can be initiated through the MS Excel user interface.

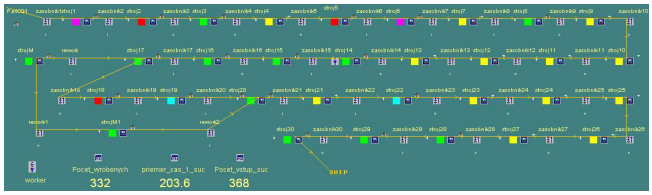


Fig. 4 View of the WITNESS model of the experimental production line

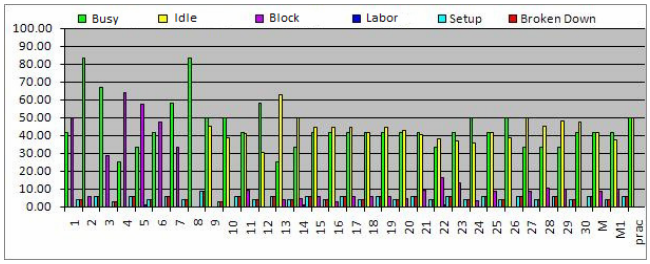


Fig. 5 Illustration of the WITNESS simulation statistics obtained by a procedure in MS Excel

At first, a simulation model of a fully synchronized production line was used to evaluate the maximum capacity of the system. Afterwards, various combinations of the values of cycle times, times between failures, repair times, set-up times, etc. have been prepared for the experiment. Numerous series of experiments have been carried out for several types of production systems in order to study the ability of the particular methods to detect a bottleneck.

Next samples illustrate the graphical output of the analysis results for the particular methods.

	1	2	3	4	5	6
count interval activity	481	160	521	561	479	480
length interval activity	3600	6750	5160	2600	2996	3800
average length interval activity	7.4844	42.188	9.904	4.6346	6.2547	7.9167

Fig. 6 Illustration of the calculation for the Active period method

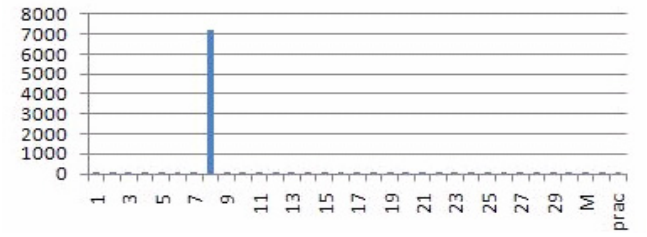


Fig. 7 Result of the bottleneck analysis using the Active period method

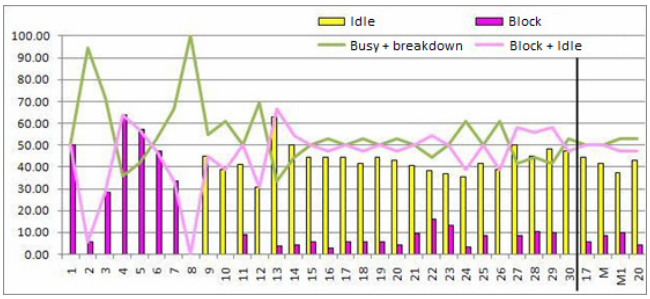


Fig. 8 Result of the bottleneck analysis using the Turning point method

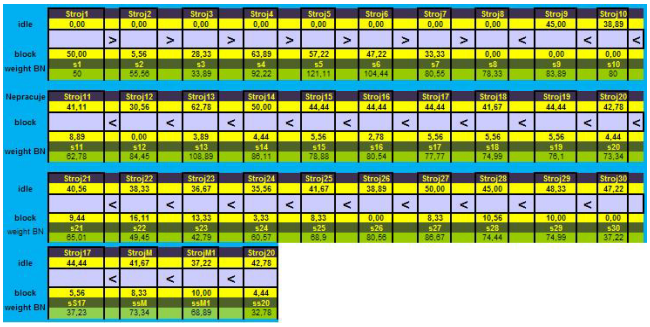


Fig. 9 Sample of the results for the first stage of the Arrow-based method. Primary bottleneck is the workstation number 8 with the value S_8 = 78.333.

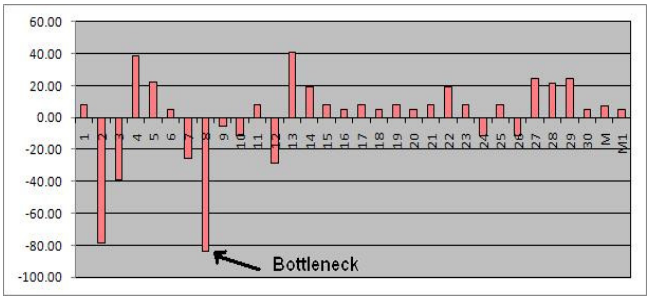


Fig. 10 Results of the bottleneck analysis using the Criticality indicator based method

The advantages of *the Active period method* are:

- simple and reliable bottleneck evaluation,
- the likelihood of being the bottleneck is reliably identified for all machines and AGV's, conveyors or labors
- indicators are computed for each workplace separately, thus the method can be implemented independently of the production system structure (the workplace order, branching of the process, rework loops, etc.) .

The experiments revealed the drawbacks of this method in case when several bottlenecks of the same severity occurred; in such a case only one of the bottlenecks was marked and therefore after extending the capacity of this workplace the throughput didn't increase.

*The Turning point method yield*s good results; nevertheless, it has several disadvantages:

- direct evaluation of the global bottleneck from several local bottlenecks is not possible,
- labor is not considered as a potential bottleneck.

The Arrow-based method based on comparison of the simulation statistics of the starvation and blocking of the neighboring workplaces is able to evaluate multiple bottlenecks. The bottleneck of the largest severity is the workplace with the maximum value of the specific index. A disadvantage of this method is its low reliability to find a bottleneck if it is located at the beginning or at the end of production line.

The Criticality indicators based method allows a direct quantitative identification of the bottleneck workplace, considering machines and labor. The element of the workplace causing congestion is discovered in the second stage after the analysis of statistics. Compared with other methods, this one determines not only the bottlenecks and their severity but also the reserves in the production system (workplaces with unused capacity). This is a good starting point for the automation of the process synchronization when the throughput maximization can be achieved by synchronization of all production process elements to ensure the continuous flow of material.

Comparison of the values of the partial indicators allows detecting bottlenecks without knowledge of the production system structure which makes the algorithm applicable for various types of production processes.

CONCLUSION

The main objective of this paper was to compare several bottleneck detection methods developed in the last decade and to explore the advantages and disadvantages of each approach. The results of the study showed that the Criticality indicators based method gives good results compared with other methods and is prospective to be used for automated synchronization of the production line.

ACKNOWLEDGEMENT

The work has been supported by the Scientific Grant Agency of the Ministry of Education of the Slovak Republic and the Slovak Academy of Sciences under Grant No. 1/0544/09.

REFERENCES

[1] BILLER, S., LI, J., MARIN, S.P., MEERKOV, S. M., ZHANG, L. (2008): Bottlenecks in Production Lines with Rework: A System Approach. In: *Proceedings of the 17th World Congress the Int. Federation of Automatic Control*, Seoul, Korea, July 6-11, 2008, pp.14888-14893

[2] CHIANG S.-Y., KUO C.-T. AND MEERKOV S.M. (2000): C-Bottlenecks in serial production lines: Identification and application, In: *Mathematical Problems in Engineering* 7 (2000), pp. 543–578

[3] GOLDRATT, E.M. AND J. COX (1984): *The Goal. Excellence in Manufacturing.* North River Press, Croton-on-Hudson, N.Y 1984

[4] KRÁLOVÁ, Z., BIELAK, M. (2004): Production process synchronization using simulation in WITNESS. In: *Proceedings of 7th Conference WITNESS,* Kozov, Czech Republic, June 3-4, 2004, pp. 78-84 (in Slovak)

[5] KUO C.-T., LIM J.-T. AND MEERKOV S.M. (1996): Bottlenecks in serial production lines: A system-theoretic approach, *Mathematical Problems in Engineering* **2** (1996), pp. 233–276

[6] MARKT, P.L., MAYER, M.H. (1997), WITNESS simulation software: a flexible suite of simulation tools, *Proceedings of the 1997 Winter Simulation Conference,* pp.711-17.

[7] LAWRENCE, S. R., BUSS, A. H. (1995): Economic analysis of production bottlenecks. *Mathematical Problems in Engineering*, Volume 1 (1995), Issue 4, pp. 341-363

[8] LI, L., CHANG, Q., XIAO, G., BILLER, S. (2007): Bottleneck Detection of Manufacturing Systems Using Data Driven Method. In: *Proceedings of the 2007 IEEE Int. Symposium on Assembly and Manufacturing*, Ann Arbor, Michigan, USA, July 22-25, 2007, pp. 76-81

[9] ROBINSON, S. (1996): *Successful Simulation. A Practical Approach to Simulation Projects.* McGraw Hill Book Co Ltd , 1996, ISBN 9780077076221

[10] ROSER, CH., NAKANO, M., TANAKA, M. (2001): A practical bottleneck detection method. In: *Proceedings of the 2001 Winter Simulation Conference, WSC 2001,* Arlington, VA, USA. ACM, December 9-12, 2001, pp. 949-953

[11] ŠVANČARA, J., KRÁLOVÁ, Z. (2009): Simulation Based Performance Analysis of the Multi-product Manufacturing System. In: *Proceedings of the 17th International Conference on Process Control '09,* Štrbské Pleso, Slovakia, June 9 - 12, 2009, pp. 249–253

Ing. Michal Leporis

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of Control and industrial Informatics
Ilkovičova 3
812 19 Bratislava
michal.leporis@stuba.sk

Doc. Ing. Zdenka Králová, PhD.

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of Control and industrial Informatics
Ilkovičova 3
812 19 Bratislava
zdenka.kralova@stuba.sk

Parallel computing on graphics cards

Slavomir Kajan, Juraj Slačka

Abstract
This paper deals the potential of parallel computing on graphics cards. In many applications we encounter time-consuming mathematical or general computing operations, which handle large amounts of data. One solution to speed up such calculations is to get them done on a graphics card. Graphics cards with CUDA (*Compute Unified Device Architecture*) interface allow programmers to use parallel performance of modern graphics cards. In Matlab we tested the acceleration of chosen mathematical operations on selected graphics cards. As practical example we implemented hand written digit recognition using artificial neural networks.

Key words: parallel computing, graphics processing unit (GPU), compute unified device architecture (CUDA)

Introduction

Currently we can see various software applications with high computation demands on mathematical or general operations that processes large amount of data. Acceleration of such time-consuming calculations is usually done by increasing computational power of a single PC, or by connecting multiple PC's into a computer Grid or Cluster. Such a solution requires a significant investment in the hardware equipment. Another way to speed up mathematical calculations is to compute them on graphics card. Accelerating of general calculations on graphics cards massively expanded in year 2007(February), when *nVidia* released to the public the first official version of application interface for general calculations on graphics cards. This interface is called CUDA - *Compute Unified Device Architecture and* it enables the programmers to use massive parallel performance of modern graphics cards, which have been designed for entertainment and computer games.

1. Principle of parallel computing on GPU

The graphics cards with older architecture used to calculate graphics scenes by vertex and pixel units. In modern games it happened sometimes that at the same moment there were used only the pixel units and the vertex units had nothing to do or vice versa. That's why *nVidia* in the new generation of the graphics cards (family 8000) introduced a new architecture of graphics processing unit. This new graphics card does not use separated vertex and pixel units, but for all types of calculations it uses the stream processors, which are more complex. These stream processors are all identical and they dynamically share the computing tasks as required by actual situation. With release of CUDA interface programmers have the opportunity to work with these processors and they can perform on them other than graphics calculations. Nowadays, the graphics cards witch CUDA support are used in scientific research centers and universities in various sectors as biology, medicine, physics, astrophysics, artificial intelligence, chemistry, etc... [5, 8].

If we compare a stream processor with a central processing unit (CPU) in serial calculation, the CPU will definitely win. The power of graphics cards is in the parallel data processing. In one graphic chip we can commonly find hundreds of stream processors. Their combined performance is many times better than the CPU performance. Graphics processing unit consisting of stream processors and graphics memory is in short known as GPU (see fig. 1)

To determine theoretical computational performance of CPU or GPU we use the number of flop's (floating point operations per second). For example a two-core CPU Core 2 Duo E7400 has the theoretical computational performance of 22,4 Gflop and the graphics card Nvidia Geforce GTS 250 with 128 stream processors has the theoretical computational performance of 700 Gflop. From these numbers it can be seen that this graphics card is theoretically about 28 times faster than Core 2 Duo in single precision calculations [7].

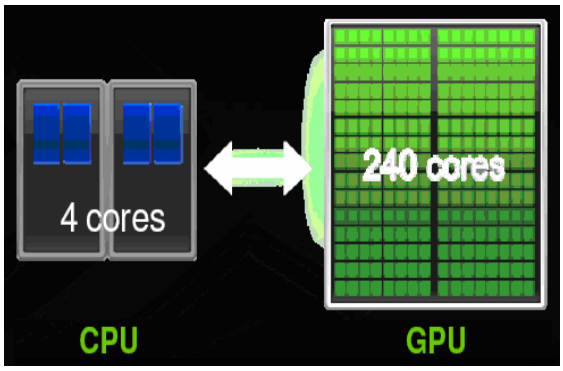


Fig.1 Comparison of CPU and GPU

Of course not every problem can be well parallelized, but there are applications in many fields such as edge detection in images, artificial intelligence and neural networks, where scientists by using powerful graphics cards reached speed up 50-100 times over CPU. Similar acceleration can be seen in the graph in figure 2.

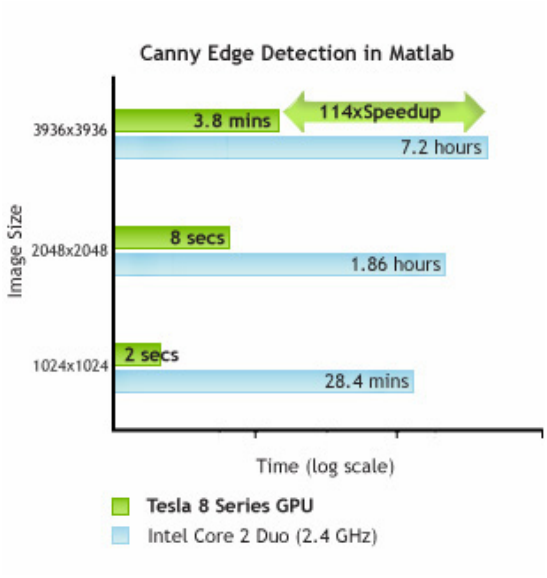


Fig.2 Speedup example of edge detection

2. Parallel computing on GPU for Matlab

During the massive expansion of parallel computing on the GPU's, companies as AccelerEyes and GP-You Group have developed an extension for Matlab, which directly allows users to use in Matlab the performance of graphics cards. The company AccelerEyes offers a library for parallel computing on the GPU called Jacket [1, 3]. Conversely, the company GP-you Group has developed a similar library called GPUmat which is like Jacket and allows parallel computations on the GPU, but it is distributed for free (freeware) [2, 4]. Both libraries provide basic functions for handling calculations on graphics cards. These functions differ only in few commands. The benefit of the Jacket library is that there is existence of other extensions such as application functions for graphics, neural networks and other demonstration examples. The main advantage of the library GPUmat is providing free access to a very good basis for parallel computing on graphics card. Library GPUmat is designed to be easy to integrate with Matlab and other existing programs can be easily converted to run on the GPU. Installation of the GPUmat library is very easy. You just have to copy it to a location on the disk and run GPUstart. The program will check for compatibility with CUDA interface graphics card and add routes to the directory structure of MATLAB. Before installation of *GPUmat* you should have:

- Installed graphics card with driver software
- Installed development tools - CUDA SDK ver. 2.1 or 2.2 for your OS
- Installed software CUDA Toolkit ver. 2.1 or 2.2 for your OS

To use the computing power of the graphics cards we have to create a data variable which will be located in the memory of the graphics card. This variable must be type single, and we can create it by the command GPUsingle. Here is a revealing handicap of GPU calculations that GPU's can work only with a single type of data, rather than the type of double, which reduces the accuracy of the calculation. However GPU's with core G200 or newer can process double data format, but this ability is not implemented in Jacked or GPUmat library. When we create such a variable, it can be applied to any function in Matlab while GPUmat will care that the calculation will run on the graphics card.

In the example of multiplication of two matrices A, B, we will explain how the calculation works. We can write matrices A and B into the graphics memory by using the command GPUsingle. Multiplication of matrices A and B is preformed by multiplying the row of matrix A and the column of matrix B. This computation is carried on each GPU stream processor and the result is written to the position of the matrix C in the graphics memory. (see fig. 3).

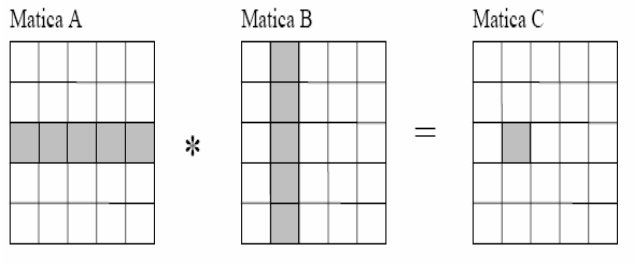


Fig.3 Image of matrix multiplication on GPU

In Matlab we can do this as follows:

```
>> A=GPUsingle(rand(100)); % create matrix A on GPU
>> B=GPUsingle(rand(100)); % create matrix B on GPU
>> C=A*B;          % multiply A and B on GPU
```

Similarly, the GPU can perform various mathematical operations. If we want the data from graphics memory to pass back into the CPU cache we can use the command CPUsingle.

Calculation of mathematical operations on the GPU has two main handicaps. The first one is that not all mathematical operations can be performed equally well in parallel, thereby speeding up the calculation is different and cannot be accurately determined. Another handicap is that the higher speed of calculation is dependent on the number of the processed data. The acceleration can usually be seen when the number of processed data is about 1e4 to 1e5. This threshold depends on the ratio of computational power of GPU and CPU and also on the number of stream processors on the GPU.

3. Examples of parallel computations on chosen graphics cards

The possibilities of accelerating the mathematical calculations on the GPU in Matlab using the library GPUmat are demonstrated on the following examples: matrix multiplication, matrix multiplication by elements and the Fourier transformation. Data used for mathematical operations were randomly generated. Calculations were made on three graphics cards on three PC's with the following parameters:

1. PC 1

- CPU AMD Athlon 64 X2 Dual Core Processor 6000 + 3.02 GHz, 2GB RAM
- GPU nVidia 8600GT (256MB RAM, 32 stream process.)

2. PC 2

- CPU Intel Core2 Duo E6750 Processor+2 GHz, 4GB RAM
- GPU nVidia GTS250 (512MB RAM, 128 stream process.)

3. PC 3

- CPU Intel Core2 Duo E6750 Processor + 2.66 GHz, 4GB RAM

- GPU nVidia GTX275 (896MB RAM, 240 stream process.)

3.1 Examples of matrix multiplication and Fourier transformation

In testing, we monitored the dependence of calculation acceleration on the number of processed data, where speedup of calculation has been calculated as a fraction of CPU time - T_{CPU} and the GPU time - T_{GPU}. Time calculation T_{CPU} and T_{GPU} were calculated as the statistical median of ten consecutive running calculations, thereby eliminating the random error.

$$Speedup = \frac{T_{CPU}}{T_{GPU}} \quad (1)$$

The results of the individual measurements on three different PCs are shown in figures 4 to 6.

In figure 4 you can see the results of the accelerating multiplication of two randomly generated matrixes A and B. In Matlab we calculated it as follows:

```
>> C=A*B;          % matrix multiplication
```

Graphics card GTX275 had the best results thanks to the highest number of the stream processors.

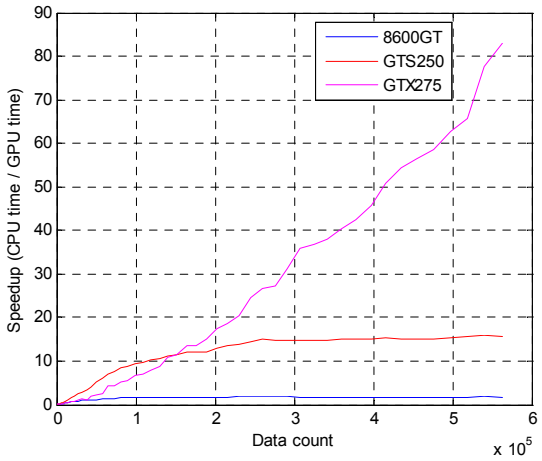


Fig.4 Computation speedup of matrix multiplication

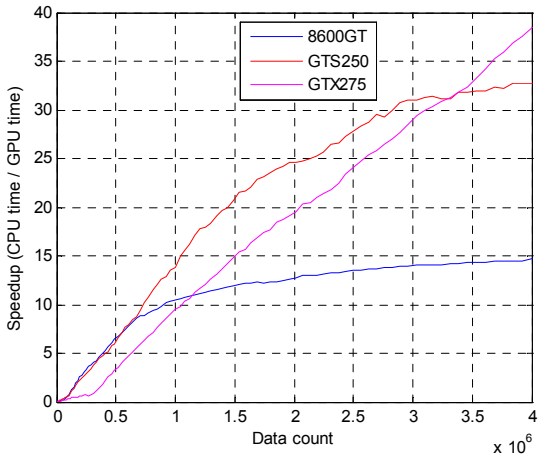


Fig.5 Computation speedup of matrix element multiplication

In figure 5 are shown the results of the speedup of the multiplication (element by element) between two randomly generated matrixes A and B. In Matlab we calculated it as follows:

```
>> C=A.*B;          % matrix multiplication by element
```

In figure 6 you can see the results of the accelerating the Fourier transform calculation of two randomly generated signals A and B:

```
>> C=A+B;           % add two signals
>> D=fft(C);        % calculate FFT
```

In these two last cases, the best results were achieved on the graphics card GTS250, thanks to lover clock rate in 2nd PC with GTS250 and lower capacity utilization of stream processors on GTX275 card.

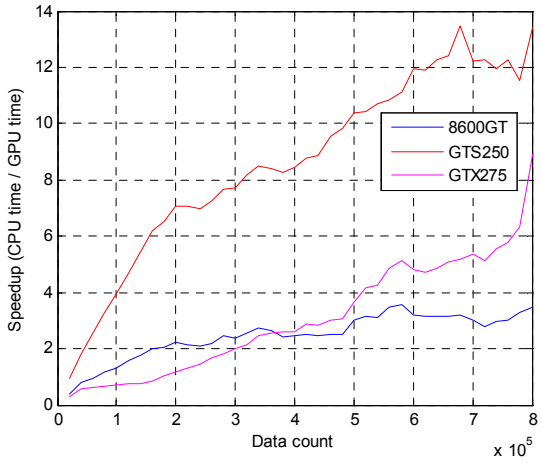


Fig.6 Computation speedup of Fourier transformation

3.2 Example of written digits recognition by neural network

As a practical example of the computation accelerated on the graphics card, we used an application for hand written digit recognition by neural network. We chose a neural network because the learning of the neural network can be easily parallelized and it is appropriate to demonstrate the application of parallel computing on graphics card. For digit recognition we used multi-layer perceptron network (MLP) with one hidden layer. The written digits shown in fig. 7 were scanned into a grid of 30x30 that means there are 900 inputs in neural network in range from 0 to 1, where 0 is white and 1 is black colour. To simulate a different number of processed data, we used a change in the number of neurons in hidden layer from range 500 to 12.000.

Fig.7 Written digits for recognition using NN

For MLP network training we used back-propagation algorithm with momentum. Calculations of the outputs and adjusting the neural network weights can be rewritten in a matrix form [6]. This enables us to use graphics card to accelerate computations. The results of the calculation speed up against CPU are shown in fig. 8. Simulations were computed on PC 3 with graphics card Nvidia GTX275.

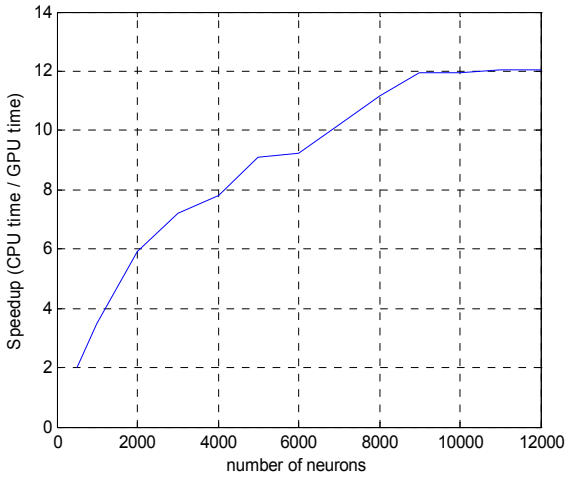


Fig.8 Computation speedup of neural network training

Conclusion

The existence of a CUDA application interface for general computations on graphics cards results in growing number of the applications using parallel computing. The library as a GPUmat or Jacket for MATLAB gives users a relatively simple way to implement intensive mathematical calculations on the graphics cards and use their advantage in parallel data processing. By measuring the performance of three graphics cards we verified, that by using the library GPUmat (and Jacket) in Matlab a speedup (up to ten) of the calculations of mathematical operations can be achieved. The advantage of parallel computing on GPU over CPU is considerably lower financial costs and vice versa, the disadvantage is the lack of support for double precision calculations by these libraries. The main disadvantage, as mentioned in the paper, is that such an approach to accelerate the calculation cannot be applied universally to every application, but only for special applications with large amounts of data such as pattern recognition, artificial intelligence, neural networks, graphics simulations, etc.

Acknowledgement

The work has been supported by the grants agency VEGA no. 1/0690/09. This support is very gratefully acknowledged.

References

[1] The AccelerEyes: Jacket User Guide, jun 2009

[2] The GP-you group: GPUmat User Guide, April 2009

[3] http://www.accelereyes.com – internet page of company AccelerEyes

[4] http://gp-you.org – internet page of company GP-you

[5] http://www.nvidia.com – internet page of company nVidia

[6] JADLOVSKÁ, A.: Modelovanie a riadenie dynamických procesov s využitím neurónových sietí. Edícia vedeckých spisov FEI TU Košice, ISBN 80-8894122-9, 2003

[7] KAJAN, S., SLAČKA, J: Matematické výpočty na grafickej karte v prostredí Matlab, internet portal Posterus, http://www.posterus.sk/?m=200907, jul 2009

[8] VISCONTI, M.: Heterogeneous GPU Computing In Computational Science, http://www.osc.edu/supercomputing/training/customize/Mark_Visconti_slides.pdf

Ing. Slavomír Kajan, PhD.

Slovak University of Technology in Bratislava

Faculty of Electrical Engineering

and Information Technology

Institute of Control and Industrial Informatics

Department of Robotics and Artificial Intelligence

Ilkovičova 3

812 19 Bratislava
E-mail: slavomir.kajan@stuba.sk

Juraj Slačka

Slovak University of Technology in Bratislava

Faculty of Electrical Engineering

and Information Technology

Department of Applied Informatics and Information

Technology

Ilkovičova 3

812 19 Bratislava
E-mail: xslacka@stuba.sk

Controller Design for the Static System with Electric Actuated Valve

Štefan Chamraz, Richard Balogh

Abstract
In our article we deal with the replacement of the pneumatic actuator of the valve with an electric one and its influence on the controller design. In second part of the article we show possible undesirable effects of integral behaviour of the electric actuator on PID controller parameters, namely its D part and its consequence on the whole system.

Keywords: PID controller, valve with electric actuator

Introduction

From time to time appears in the literature some pessimistic reports from the PID controllers applications, especially from the chemical and paper industries [1], [2], [3]. Most of them can be summarized as follows. The most often used and satisfying structure is a PI controller. For simpler processes usually the P controller is used. When there is a PID controller used often with derivative part set to zero. Even when it is non-zero, the regulation circuit behaviour is the same as if the D component was discarded. We were curious why this happens.
Also we observed the similar behaviour of the D-part of the controller when we once replaced the pneumatic valve actuator with the electric one. After this replacement, the system behaviour didn't match to theoretical calculations. Problems appeared also when we changed the operating point of the system and/or the size of the step. For the step change of size Δ the reaction of the system was good, but the step 3Δ was not corresponding to the theoretical one, even when we operated still in the linear region. These problems were more significant for control steps and less intensive for the disturbances at the system input. The original system contained just one major non-linearity – the flow characteristic of the valve. Its impact was compensated with a properly selected valve construction characteristic. Valve controller was designed as a relay with dead zone and hysteresis. For an analysis of its properties we simplify the system as much as possible (removing non-important nonlinearities) so we can observe the influence of the actuator properties on the whole system. The resulting simplified system consists of integrator, sampler and servo actuator only. Our intention was to design the parameters so that the system behaviour will be the same as with pneumatic valve actuator.

Simplified model of the system was created in Matlab/Simulink. Model of the controlled system was based on the heat exchanger **PPV**, control valve **V** (modelled by the linear flow characteristics $q_B = K_V\,\varphi$) and valve actuator **Pv** (modelled by the linear transfer characteristics $\varphi = K_P\,m$). Corresponding block diagram is in Fig. 1, where m_{B0} represents an operating point corresponding to the linearized system with transfer function (2). The model of the system included also a gain of the pneumatic actuator and control valve. Our task was to design a digital control system (controller) and actuator parameters such that we achieve the required quality for steps of the tracking performance and disturbance rejection.

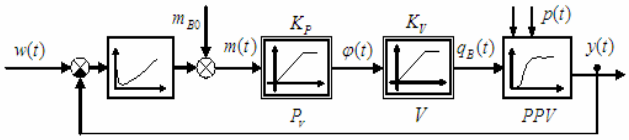


Fig. 1 Block diagram of the system

We required the control process without the overshoot and predefined settling time.

Such a simplified system can be rearranged to the system in Fig. 2. This model assumes the ideal characteristics of an actuator (**AC**) and the measuring element (**MP**).

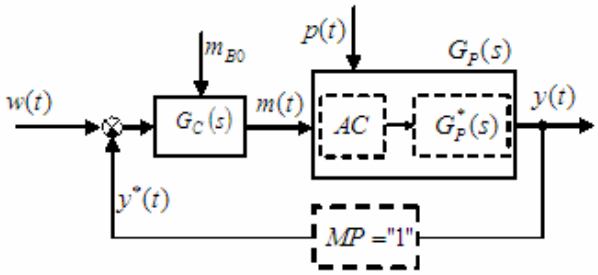


Fig. 2 Simplified diagram of the control system

In such sufficiently simplified example (see Fig. 2), we replaced the pneumatic actuator by the electric motor (AC). Thus it is necessary to replace the transfer function of the pneumatic actuator PP(s) = KP by the transfer function of the electric actuator Φ(s)/M(s) = 1/sTM (see Fig. 3).

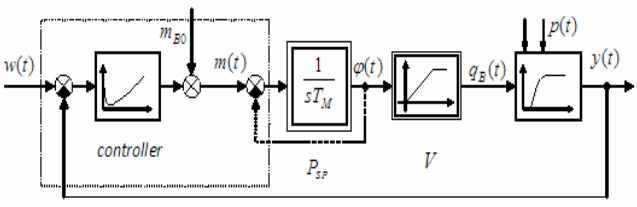


Fig. 3 Diagram of the system with electric actuator

If we complete negative feedback (dashed line in Fig. 3) around the electric motor, we can eliminate the integrating behaviour of the actuator. Its transfer function is then

$$\frac{\Phi(s)}{M(s)} = \frac{1}{(1+sT_M)} \quad (1)$$

If the actuator time constant T_M is small relative to the dynamics of the controlled system, we can consider (1) as the unity transfer function. To determine the integrating time constant we used a similar approach as in [4] to determine the sampling period T_V. Satisfactory results for the T_V were achieved just when we take into the account the whole system. First, the continuous controller was designed so that we achieve the desired transfer characteristics of the system. Considering the transfer properties, we then "recalculated" the T_V such that the delay of the signal due to the sampling was acceptable. When we design T_V considering just the controlled system, the results were not satisfactory. Similar approach will be used here.

1. Controller design

Let us assume, that the sensor and actuator transfer functions are unity and controlled system is

$$G_P(s) = \frac{Y(s)}{M(s)} = \frac{1}{(1+30s)(1+10s)} \quad (2)$$

Structure of the controller designed using the method of inverse dynamics is

$$M(s) = K_C\left((bW(s)-Y(s)) + \frac{1}{T_I s}E(s) + (cW(s)-Y(s))\frac{sT_D}{1+sT_f}\right); \quad (3)$$

$$T_f = \frac{T_D}{N}; \quad N \geq 10$$

Controller parameters are: $K_c = 1$, $T_I = 40$ [s], $T_D = 7.5$ [s], $b=c=1$ and they correspond to the desired dynamics of control $\lambda = 40$ [s]. The image of the controlled variable is

$$Y(s) = \frac{1}{\lambda s+1}W(s) + \frac{\lambda s}{\lambda s+1}G_P(s)P(s) \quad (4)$$

It is evident from (4), that controlled variable reach the zone of 1% insensitivity at the time about $T_{reg} = 5\lambda = 200$ [s]. When we require to reduce the settling time to its half $\lambda = (40/2)$ [s] $= 20$ [s], it is easy in theory. It used to be sufficient just to change the value of controller gain to $K_c = 2$. As follows from (4), the size of the response to the disturbance step will decrease, but the settling time will not. That leads to the "adjusting" the process. Generalized method of desired dynamics [5] allows to specify the dominant time constant of the required model. When retain the previous value $\lambda = 20$ [s], we can use this method to design proper controller for both tracking and disturbances rejection with the structure (3) and following parameters $K_c = 5$, $T_I = 33.3$ [s], $T_D = 7$ [s], $b=0.6$, $c=0.43$. Controller parameters were calculated for the model of the controlled system $G_M(s) = (1+10s)^{-2}$.

Such designed control system will not be too deteriorated with the block with irrelevant time constant. In our case that means the actuator with linear transfer function, immediate tracking of the correction controller value. This requirement is not possible in theory.

The original parameter design was based on the transmission properties of the system. Assuming a system (2), it is sufficient to simply choose an integration time constant of the actuator T_M with respect to the desired dynamics $\lambda = 40$ [s] such that system (1) will have a phase shift circa 5.7° at the frequency $\lambda^{-1} = 0.025$ [rad.s^{-1}]. That means $T_M \approx 0.1\lambda = 4$ [s]. Controller with two degrees of freedom was designed for $\lambda = 20$ [s] with corresponding actuator time constant $T_M = 2$ [s].

These were our theoretical knowledge used to determine actuator parameters. If the system on Fig. 2 was completed with the actuator model (1), resulting processes were not significantly changed. The shape of the signals was not changed nor for different sizes of the control signal nor the disturbances. This fact was contrary to our practical experience. Such a model behaved as a linear circuit. The quality of transitional processes depends only on changes of controller parameters. This leads us to improvement the model of the actuator.

2. Properties of electric drive

Electric actuator is a complicated device (see. Fig.4). In most cases is the rotational movement converted to the translational one. Electromechanical connection provides disconnection of the power when the rod moves out of range. Actuator also contains a potentiometric transmitter, to allow measuring the position of the rod. Movement of the rod has an integrating character and its time constant is usually referred as the time required getting the rod from one limit position to another.

(Dis-)energising of the actuator can be modelled as non-linearity (relay type). Since we have two directions of movement, we have to use two similar control blocks or model them as a symmetric relay with dead zone b.

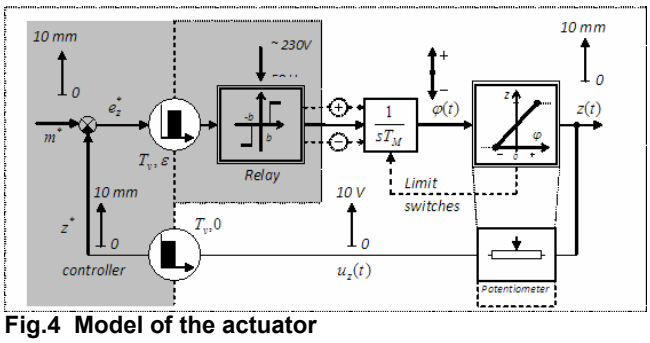


Fig.4 Model of the actuator

For digital implemented controller, the relay control is often implemented in the computer (microprocessor). This part corresponds with the gray framed part in Fig.4. When assuming the sampling period T_v and the dead band b small, the whole system on Fig. 4 can be simplified for the purposes of simulations and determination of T_M as shown on Fig. 5.

When we replace the linear actuator (1) with the structure according to the Fig. 5, we have to realize that we analyze the linearized, but not the linear system. It is not sufficient to consider only the values of parameters, but also the sizes and ranges of signals must be considered. The actuator is not a low pass filter, but a non-linear one. It can pass the signal $m(t)=A\sin(\omega t+\xi)$ without distortion, when the condition

$$\left|\frac{dm(t)}{dt}\right| \leq \frac{1}{T_M} \quad (5)$$

is satisfied. Theory of non-linear filters is described for example in [6]. Their base lies in sliding modes. They can be used to implement the pure derivation and they can suppress uncertainties – to linearize.

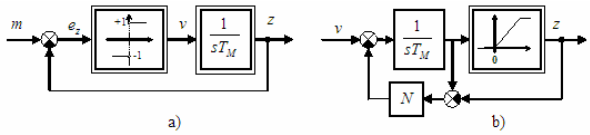


Fig. 5 Improved model of the actuator

System according the Fig. 5a operates in two basic modes – a) tracking the input signal and b) limits derivatives of the input. Affect of such derivative limiter on the quality of transitional processes can be seen on Fig. 6a, corresponding to the unit step change of the control and disturbance. The signal $y_1(t)$ corresponds to the controller (3) with $K_c=b=c=1$, the signal $y_2(t)$ corresponds to the controller (3) with $K_c=2$, $b=c=1$ and finally, the signal $y_{2DOF}(t)$ corresponds to the controller (3) designed as a controller with two degrees of freedom.

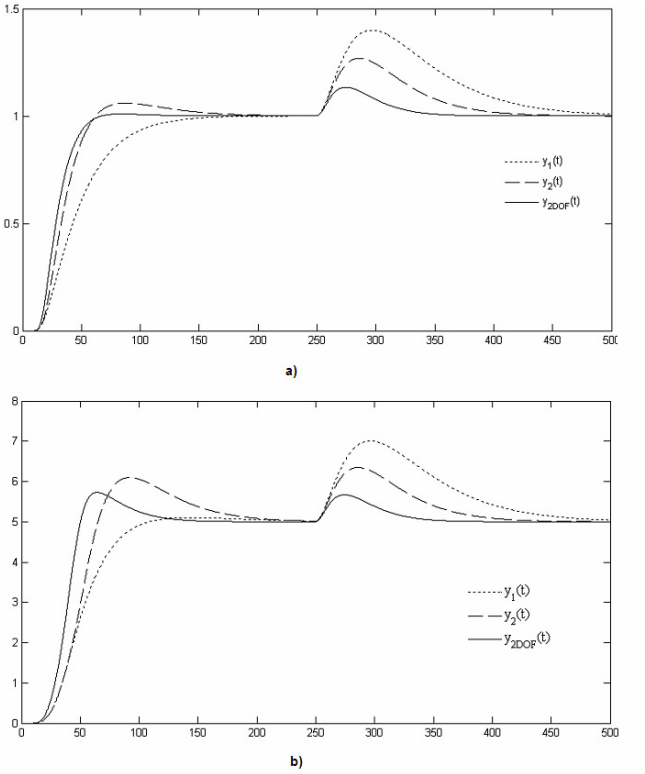


Fig. 6 Effect of the derivation limiter on signals

Effects of the actuator on the system are even more visible, when we change the size of the steps are increased to the value of five (see Fig. 6b – controller parameters are the same as in Fig. 6a). From the above signals it is clear that the dynamics of the system is different for control and for rejecting disturbances. Above is true also when the controller is designed with two degrees of freedom. One of the possible solutions to limit the impact of actuator is to deploy a filter into the control loop.

3 Identification of the actuator time constant

The base for a following analysis is a generalized method of inverse dynamics. Principle of this method was stated as follows [5]: Let the controller behaviour is the same as the non-ideal first order tracker with

$$G_{wy}(s)=\frac{1}{1+s\lambda}, \tag{6}$$

then the transfer function of the disturbance is

$$G_{py}(s)=G_P(s)\frac{s\lambda}{1+s\lambda}. \tag{7}$$

These transfer functions corresponds to the diagram on Fig. 7a. If the controller is designed using the method in [5], instead of the controlled system we use the required model of the controlled system. Simplified diagram of control system is in Fig. 7b. It is evident from (7) that requirements on disturbance transfer function are lower.

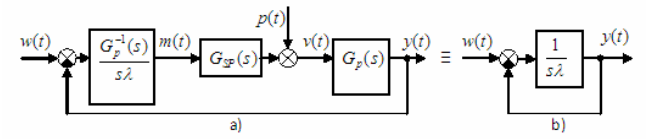


Fig. 7 Alternate system diagram

Assuming that the transfer function of the real servo actuator is $G_{SP}(s)$, then we can model the effects of the disturbance and control signal on controlled system (non-linear system described as linear one)

$$V(s)=\frac{G_C(s)G_{SP}(s)}{G_P(s)G_C(s)G_{SP}(s)+1}W(s)+\frac{1}{G_P(s)G_C(s)G_{SP}(s)+1}P(s) \tag{8}$$

and it can be rewritten using the method of inverse dynamics to the

$$V(s)=\frac{K_C\left(1+\frac{1}{T_I s}+T_D s\right)\frac{1}{T_M s+1}}{1+\frac{1}{\lambda s}\frac{1}{T_M s+1}}W(s)+\frac{1}{1+\frac{1}{\lambda s}\frac{1}{T_M s+1}}P(s) \tag{9}$$

It is clear, from the relationship (9), that if we choose $T_M \approx 0.1\lambda$, transfer function of the disturbance (7) will be changed only minimally. When the amplitude of the signal generated by the D-component of the controller $K_R T_D s$ is large enough, then it will be transferred with its nominator (1).

For linear system we can claim that there will be virtually no change also in the transfer function of the control (6). In particular, we mean the derivation of the set-point step change. When considering also the implementability of the controller D-part, then we are interested in whether it is true $T_f >> T_M$.

When we consider in the numerator of (9) real properties of the actuator instead of the (1), then frequencies over the $\omega > T_M^{-1}$ will be filtered. Furthermore, when considering an ideal Dirac pulse at the input of the system in Fig. 5, it will not be transferred to the output at all. When the D-part will be implemented as the 1st-order filter, then the response of the feedback circuit may be a function of the size of the step, T_f constant and system parameters. From the outside, this may occur that the real actuator will block out the effect of the D-part of the controller in the *control line*. Note, that $T_I \geq 4\,T_D$ and $\lambda = f(T_I)$.

For the system (6) is the response of the controlled value $y(t)$ on the set-point step $w(t)$ given $y(t)=A_0(1-e^{-t/T})$, where A_0 is the step size. When $A_0=1$ or $A_0=7$, the ±1% band is reached at the same time (approx. 5λ). This is the property of linear systems.

Linear system on Fig. 7b remains unchanged when we add a block with transfer function „1". The simplest possible replacement for the actuator is on Fig. 5a. On the Fig. 5a is integrator pictured as a non-linear block. Its implementation is on Fig. 5b. When the tracking block (Fig. 5a) is added to the Fig. 7b, we obtain system on Fig. 8. Outwardly, the both systems should have the same behaviour. While for the

system on the Fig. 7b it doesn't matter if the size of the step is one, five, or ten, for system on Fig. 8, we have to choose the proper value of the time constant T_M. When we choose $T_M \approx \lambda^*$, $\lambda^* = 0.1\lambda$, for the step of size 1, $T_M \approx 0.2\lambda^*$ for the step size 5 and $T_M \approx 0.1\lambda^*$ for the step size 10, responses (see Fig. 7b and Fig. 8) will be almost the same. It is obvious that the T_M for step 10 suits also for step 1.

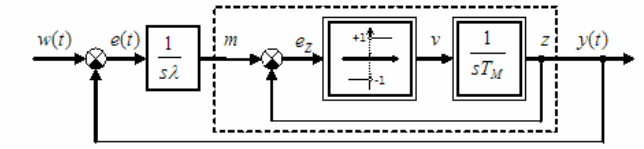


Fig. 8 System modified with tracker block

Signals on Fig. 9 are drawn for $\lambda = 40$ [s]. On this figure, numerical index represents the size of the control step. When the index a is added, actuator time constant corresponds to a step size 10 (maximum). When the index a is missing, the actuator time constant corresponds to a step size 1. Figures a) and b) are not normalized, while c) and d) are.

Relationship (5) can be based on the rewritten in the form

$$T_M \leq \frac{\lambda^*}{A_0} \qquad (10)$$

Similarly to the relationship for the calculation of the sampling period, also there the actuator time constant is not a function of the derivative time constant of the controller. Such designed actuator time constant limits the effect of the D-part of the controller.

4 Impact of sliding modes on properties of the actuator relay

The relationship (10) was derived using the theory of sliding modes. In the system according the Fig. 8 the sliding mode occurs when the condition (5) is satisfied. This is characterized by the zero amplitude of the oscillation frequency and infinite frequency on the input of the non-linearity. Oscillations with infinite frequency could eventually destroy the mechanical parts of the system like the gears, valve packing, etc. Thus it is necessary to limit the frequency of oscillations. This can be achieved by introducing the zone of insensitivity b. Transfer function of the ideal relay is changed to the relay with insensitivity and hysteresis (see Fig. 4).

System in Fig. 4 is a non-linear digital system. Analysis of this circuit is out of the scope of this article. Here we just note, that improper sampling period with respect to the insensitivity zone, then the serial connection of the sampler and relay with the insensitivity zone will show the same behaviour as a relay with hysteresis. For the proper design of the ratio T_v and b we may use the relationship $T_v < 2T_M b$.

Conclusion

In our paper we analyzed the impact of the actuator on the properties of the control system. The result is a simple relationship whereby we can determine the actuator integration time constant so that it acts as an action follower of the disturbance controller designed using the generalized method of inverse dynamics.

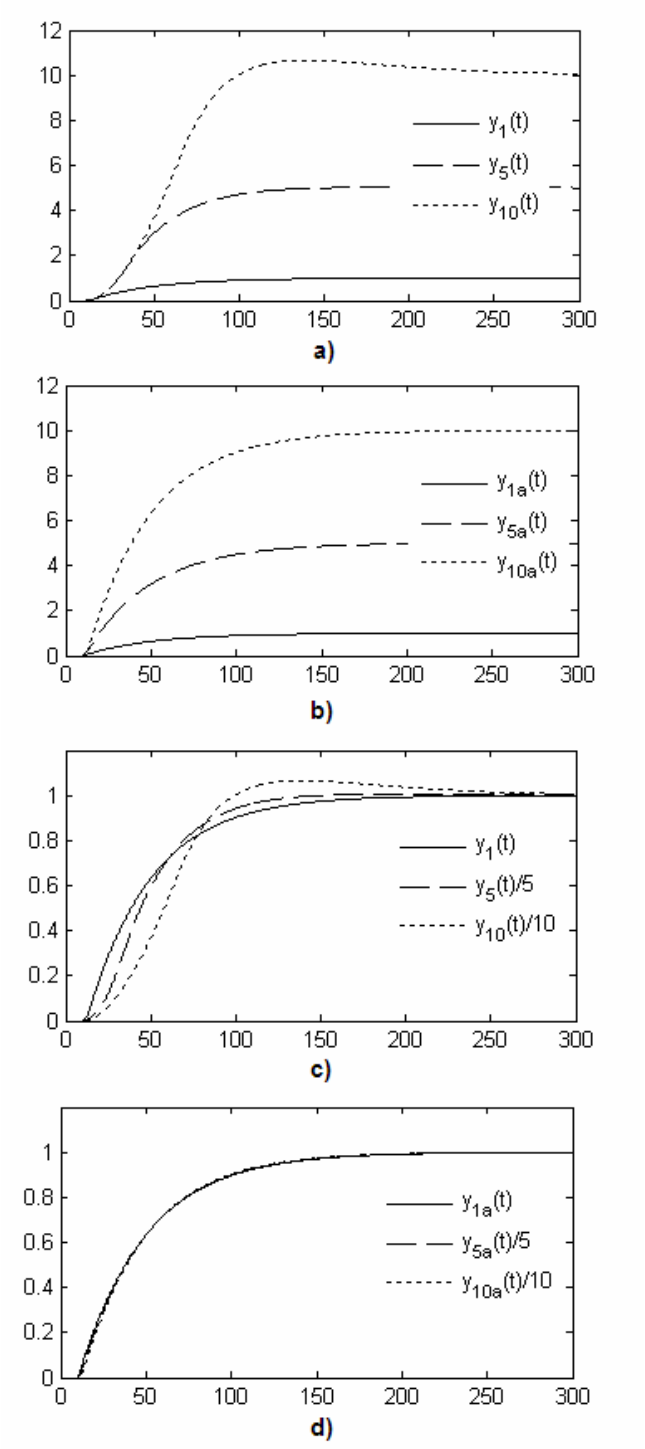


Fig.9 Signals in the system

We didn't confirm the fact that the D-part of the controller can be simply rejected. For such designed controller can the effect of the D-part considered limited and the weighting coefficient in controller (3) can be set to zero. Similar, but much less visible is the actuator effect on the P-part of the controller.

Taking into the account the fact that we are able to measure the flow through the valve, and then the above can also be used for linearization of the valve flow characteristics. Non-linear "amplification" of the valve transfer characteristics is then additional parameter to be considered when selecting the actuator.

The argument that "sometimes it is not important whether the D-part of the controller is on or not," can be found even in systems without electric motor drives. Using (3) and [5] one can show that for certain classes of properly controlled systems and proper practical task there is no difference if we use the PI, or PID controller. This, so-called over sizing controller, quite often occurs in systems with time delay.

Acknowledgement

Publication of this article was supported from the APVV project VMSP-P-0155-09.

References

[1] BURCH, R.: Monitoring and Optimizing PID Loop Performance. Aspen Technology, Inc., Houston ISA Expo conference, 2004.
[2] ANG, K.H. – CHONG, G.C.Y. – LI, Y.: PID control system analysis, design, and technology. IEEE Transactions on Control Systems Technology 13(4) : 2005 : pp. 559-576
[3] VISIOLI, A.: Practical PID control. Advances in Industrial Control series. Springer-Verlag London, 2006, ISSN 1430-9491.
[4] CHAMRAZ, Š.: Voľba periódy vzorkovania (Choice of Sampling Period) [In Slovak], International Konference Cybernetics and Informatics, February 10 - 14, 2008, Ždiar, Slovak Republic, SSKI, STU Bratislava. ISBN 978-80-227-2828-7
[5] CHAMRAZ, Š.: Syntéza regulátora polohy pomocou zovšeobecnenej metódy želanej dynamiky (Controller design using the generalized method of required dynamics) [In Slovak], International Conference February 10 - 13, 2010, Cybernetics and Informatics Vyšná Boca, Slovak Republic,
[6] CYPKIN, Ja. Z.: Relejnyje avtomatičeskije sistemy, Moskva 1974.

Ing. Štefan Chamraz, PhD.

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3
SK-812 19 Bratislava, Slovakia
E-mail: stefan.chamraz@stuba.sk

Ing. Richard Balogh

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3
SK-812 19 Bratislava, Slovakia
E-mail: richard.balogh@stuba.sk

Resolution of robot kinematic redundancy as a problem of adaptive control

Anton Vitko, Ladislav Jurišica, Andrej Babinec, František Duchoň, Marian Kľúčik

Abstract
The redundant robotic arm posses some advantages over the non-redundant ones. The most significant advantages are the great dexterity in tight spaces where the arm motion is strongly restricted and the possibility to optimize the arm motion. The numerical inversion of Jacobi matrix of a redundant manipulator may cause a problem. Therefore the paper presents an approach in which the pseudo-inversion of Jacobi matrix is obtained on–line in a closed loop control scheme. Both the redundancy resolution and motion optimization run simultaneously. Simulation experiments have shown that the designed control scheme exhibits a sufficient robustness with respect to the inexact estimation of the pseudo inverse of Jacobi matrix J^+.

Keywords: closed loop redundancy resolution, optimization

Introduction

Practical interest in redundant industrial robots comes from their numerous applications. The kinematically redundant robotic arms posses higher dexterity in performing technological tasks. Kinematically redundant robotic manipulator is a robotic arm possessing more degrees of freedom (DOF) than are necessary for establishing an arbitrary position and orientation of the end effector in a working space.

The requirements laid the arm dexterity; tool position accuracy, operation velocity and the closely related dynamic performance (mainly the robot's ability to keep the transition tracking error low) have introduced problems which call for a more sophisticated treatment of both kinematic and dynamic problems.

Supposing that each joint has the only one degree of freedom (DOF), the minimum number of joints needed for manipulation in 3D space is six. However, a manipulator possessing only six DOFs could be limited with respect to the need of following an arbitrary (but otherwise permissible) trajectory, for instance to avoid obstacles in the task space or prevent the arm from reaching a singular configuration in which the robot arm may loss its original DOFs. A singular configuration may be characterized by very high (non-realizable) joint velocities or inability to translate or rotate in certain directions. Unfortunately, assets of additional DOFs are often paid off by a more complex control law. The reason is in that the effector's motion is controlled indirectly by controlling the relative position of adjacent links. That gives rise to a fundamental problem, namely to resolve the inverse relation between the joint and Cartesian coordinates. Let us consider that robotic arm consists of *n* joints. Then the direct kinematics model is expressed by eq. (1)

$$x(t) = F[q(t)] \tag{1}$$

or alternatively in terms of velocities

$$\dot{x} = \frac{\partial F}{\partial q}\dot{q}(t) = J[q(t)]\dot{q}(t) \tag{2}$$

where $x \in R^m$, $q \in R^n$ are Cartesian and joint vectors respectively, and J denotes the $m \times n$ Jacobian matrix of partial derivates of F. Then the inverse kinematic model transforms the Cartesian velocity vector $\dot{x}$ to the joint velocity vector $\dot{q}$ in accord of (3)

$$\dot{q}(t) = J^+[q(t)]\dot{x}(t) \tag{3}$$

where J^+ stands for a generalized inverse of the $m \times n$ Jacobian matrix. In the case of a redundant manipulator, the inequality $m < n$ is valid there is (n-m) redundant joints. As a generalized inverse, the Moore-Penrose pseudo inverse of J is frequently used [1], which yields the unique minimum Euclidean norm of the vector of the joint velocity $\dot{q}$. As discussed in [1], the following argument may be raised: although the minimization of the joint velocity is certainly desirable the point of view of a measure of the dynamic performance, a potential drawback consists in using the Moore-Penrose inverse. The Moore-Penrose inverse of a matrix, whose elements are functions of *q,* is leads to high computational burden. If the $m \times n$ Jacobian matrix $J(q)$ has full rank in a certain domain then the pseudo inverse is given by the explicit formulation (4)

$$J^+(q) = J^T(q)[J(q)J^T(q)]^{-1} \tag{4}$$

The matrix multiplication and the calculation of the $m \times n$ matrix inversion can't be avoided if one wants to obtain an explicit expression for $J^+(q)$. With this fact kept in mind, in the authors' paper [2] a control scheme for resolution rate control of redundant manipulator was presented, utilizing an adaptive regulator in which J^+ is updated and used as in the role of a gain matrix. As an extension of this work, this paper presents a generalized approach, being more universal while preserving all advantages of the original one. To be more specific the original scheme is supplemented by an optimization scheme based on the projection of the homogenous part of a general solution of (2) onto the null space of Jacobian matrix J. Let us recall that the general form of direct kinematics at the velocity level is

$$J(q)\dot{q} = \dot{x} \tag{5}$$

The general solution to (5) consists of the two parts as follows

$$\dot{q} = J^{+}(q)\dot{x} \pm [I - J^{+}(q)J(q)]\rho \tag{6}$$

where ρ is arbitrary vector, and

$$\dot{q} = \dot{q}_p + \dot{q}_n \tag{7}$$

The $\dot{q}_p$ is a particular solution satisfying the required motion i.e. a solution from the subspace spanned by the rows of the Jacobian matrix i.e.

$$\dot{q}_p = \dot{q} \in R^n \mid J\dot{q} = \dot{x} \text{ for a given } \dot{x} \tag{8}$$

The second part $\dot{q}_n = [I - J^{+}J]\rho$ is a homogenous solution within the null space of Jacobian i.e.

$$\dot{q}_n = \{\dot{q} \in R^n \mid J\dot{q} = 0\} \tag{9}$$

For any desired Cartesian velocity $\dot{x}$ the homogenous velocity $\dot{q}_n$ can be arbitrary chosen, so that it can be used to meet various optimization criteria. Further specification of the projection idea depends on the choice of the vector ρ. From the viewpoint of reaching the extreme value of a given scalar performance criterion H(q), it is natural to substitute the gradient of a given criterion for the vector ρ. Then (6) will take the form [3, 6]

$$\dot{q} = J^{+}\dot{x} \pm K(I - J^{+}J)gradH(q) \tag{10}$$

In particular, the control scheme was verified for the performance criterion securing that joint angles will not reach given limit values. The criterion has the form (11)

$$H(q) = \sum_{j=1}^{n} \frac{(q_{j\max} - q_{j\min})^2}{(q_{j\max} - q_j)(q_j - q_{j\min})} \tag{11}$$

where q_j is the j-th joint angle, $q_{j\max}$ and $q_{j\min}$ are the upper and lower limits of the joint angle q_j respectively. It is clear that the criterion (11) penalizes the joint angles being near to their limit values (value of H(q) rises as the joint angle approaches its limit) and goes to the infinity at the joint limits. By this action the angles between each couple of consecutive arms are kept as much as possible away of their limits. In other words, during the arm motion the joint angles are forced to be the least possible.

1. Synthesis of control algorithm

On account of brevity here are presented only the basic ideas from which the control algorithm stems. A full derivation of them is found in [5]. The principal idea of the synthesis is based on a conversion of the original inverse kinematic problem, comprising both evaluation of the J^{+} and successive determination of the *q(k+1)* from the previous one, into a control problem. For this purpose let's rewrite (3) into a discrete form:

$$\Delta q(k) = J^{+}[q(k)][x_d(k+1) - x(k)]$$

$$q(k+1) = q(k) + \Delta q(k) \tag{12}$$

where $x_d(k+1)$ stands for the vector of desired values of Cartesian trajectory. That means in essence, that an integral controller with variable gain was included. As a controller gain matrix, the pseudo-inverse J^{+} with continually updated entries is used. The system scheme for the closed loop control is depicted in Fig.1.

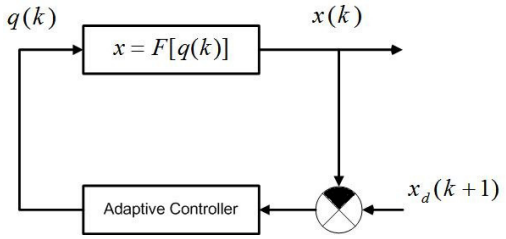


Fig.1 Adaptive control scheme

Let us suppose that the mean of random disturbances acting on the manipulator is equal to zero. Then to estimate J^{+}, i.e. the controller gain matrix, each control step may be done by applying algorithms of stochastic approximation in a deterministic fashion. In order to ensure "sufficiently excited trajectory", the algorithm is augmented with an active experiment [5]. The active experiment is based on the fact that, the "n" estimating steps are embedded between each couple of consecutive control steps i.e. between each couple $[\Delta q(k), \Delta q(k+1)]$. In particular, the series of "n" mutually orthogonal vectors $\Delta q(i)$, i= 1,2….n, is generated by Gramm-Schmidt ortogonalization algorithm. Consequently, the estimation of the J^{+} takes at least theoretically, just "n" steps. Then the estimation algorithm is the form: [2, 5]

$$J^{+}(i+1) = J^{+}(i) + \frac{1}{J_i^T J_i}[\Delta q(i) - J^{+}(i)J_i]J_i^T \tag{13}$$

where J_i, i =1,2…n stands for the i-th column of the Jacobian matrix. The complete closed loop control scheme is shown in Fig. 2, where J(k), $J^{+}(k)$ stands for the $J[q(k)]$, $J^{+}[q(k)]$ respectively.

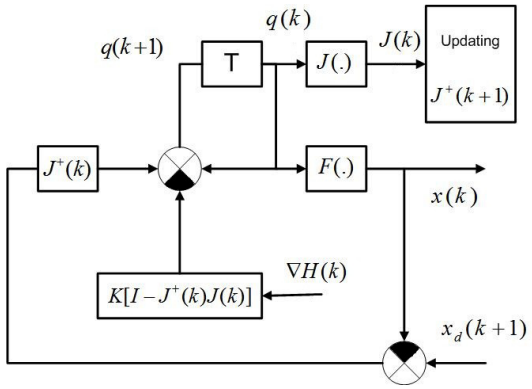


Fig.2 Scheme of the closed loop control

The block of updating the $J^{+}(k+1)$ works in accordance with the algorithm

for k=0 to < number of program steps > do
begin
$J^{+}(1) = J^{+}(k)$;
begin
for i=1 to n-1 do
J_i = *< i-th column of* $J^{+}(1)$ >
$\gamma(i) = (J_i^T J_i)^{-1}$;
$J^{+}(i+1) = J^{+}(i) + \gamma(i)[\Delta q(i) - J^{+}(i)J_i]J_i^T$;
end

$J^+(k+1) = J^+(n)$

end

where $\Delta q(i)$ is a vector having 1 in its i-th entry, while zero elsewhere. It means that within each programming step, counted by the parameter "k", runs just "n" updating steps. The initial value $J^+(k=0)$ may be chosen equal to unit matrix.

As far as the stability of the closed loop is concerned, it could be shown [5], that the stability margin will be great if product JJ^+ approaches the unit matrix. But simulation experiments have shown that the control scheme exhibits sufficient robustness to both uncertainties in determination of the J^+ and size of sampling period.

2. Simulation experiment

Simulations were performed to verify the feasibility of the proposed optimization scheme. A 3 DOF planar manipulator (Fig. 3) was required to trace a straight line starting from various initial conditions and ending in origin.

The manipulator possesses three revolute joints. Each joint is individually actuated and the arm operates in over the horizontal plane. Considering the end-effector position only, the arm is redundant, having *r=1* degree of redundancy

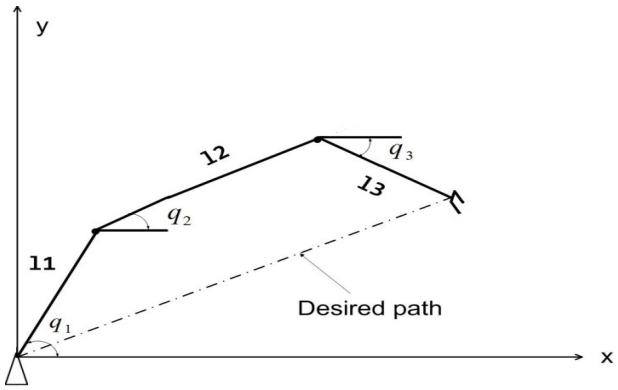


Fig.3 Three DOF planar manipulator

The model of direct kinematics is given by (14)

$$x = \sum_{i=1}^{3} l_i \cos(q_i), \quad y = \sum_{i=1}^{3} l_i \sin(q_i) \tag{14}$$

with the arm lengths, l_1=10, l_2=20, l_3=15. The corresponding Jacobian matrix is:

$$J(q) = \begin{bmatrix} -l1\sin q1, & -l2\sin q2, & -l3\sin q3 \\ l1\cos q1, & l2\cos q2, & l3\cos q3 \end{bmatrix} \tag{15}$$

Optimized movements of the links with are presented in Fig.4. It shows consecutive configurations of the arm if the optimization is switched on. It is clear that in this case, the angle q_1 is kept far from its limits. Its value varies as close to the centre of its permissible range. It can be observed from Fig.4 that range within which varies angle q_1 is from 135° to 75°, hence it is within limits $0 \le q_1 \le 150^\circ$.

An interesting dependence on the number of control steps exhibit tracking errors $e_1 = x_d - x$, and $e_2 = y_d - y$ where *x* and *y* are Cartesians of the end-effector As can be seen in Fig. 5 they are very low what implies high tracking accuracy. Observable errors occur only if the end-effector approaches the origin i.e. if number of control steps becomes greater them 50. Such behavior is caused by the fact that the origin is one of the singular points, therefore the sensitivity of the Cartesian position *x* to the accuracy in setting of the angles q_i is very high. Nevertheless, the maximum tracking error reaches the absolute value of 1×10^{-14} which is negligible with respect to the task space having the dimension of ten units. The high tracking accuracy is reached due to the adaptive scheme illustrated in Fig. 2.

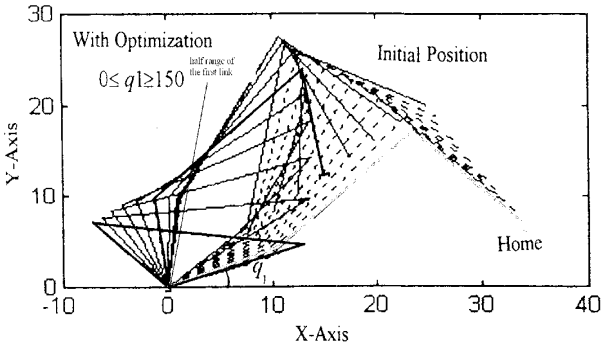


Fig.4 Subsequent positions of the robot arm.

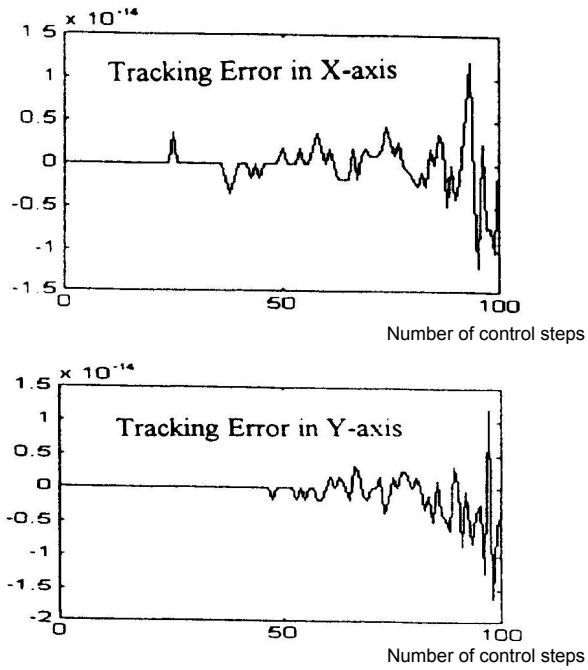


Fig.5 Tracking errors

Conclusions

There was synthesized the adaptive feedback loop allowing updating the pseudo-inverse Jacobian matrix in each control step. To generate a "sufficiently excited trajectory", which was needed for securing the quickest possible adaptation process, the algorithm was augmented with an active experiment, namely "n" auxiliary steps were embedded between each couple of consecutive control steps i.e. between each couple $[\Delta q(k), \Delta q(k+1)]$. In particular, the series of "n" mutually orthogonal vectors $\Delta q(i)$, i= 1,2….n , was generated by Gramm-Schmidt ortogonalization procedure and embedded between each couple of the control steps. Due to this modification the estimation of the J^+ has taken just "n" steps. The algorithm has shown to be robust with respect to the disturbances intentionally inserted into the current values of J^+ and secures very low tracking errors.

References

[1] LOVASS-NAGY, V., SHILLING, R. J.: Control of Kinematically Redundant Robot Using I-Inverses. IEEE Trans. Syst. Man & Cyb., SMC 17 (1987) No. 4, pp. 644-649.

[2] VITKO, A., RODZINÁKOVÁ, O.: Towards the Solution of the Kinematic Problem of Robot Manipulators. 'SMCR' 93, Torino, Italy, pp. CM 19-24.

[3] DUBEY, V. R., EULER, J. A., BABCOCK, S. M.: Real Time Implementation of an Optimization Scheme for Seven-Degree-of Freedom Redundant Manipulators. IEEE Trans. on Robotics and Automation 7, (1991), No. 5, pp. 579-588.

[4] ZGHAL, H., DUBEY, V. R., EULER, J. A.: Collision Avoidance of a Multiple Degree of Redundancy Manipulator Operating Through a Window. J. of Dyn. Systems, Measur. and Control, Vol. 114 (1992), pp. 717-720.

[5] BORŠČ, M., VITKO, A., RODZINÁKOVÁ, O.: Discrete Controller of a Multidimensional Static System with Variable Gain. Elektrotechnický časopis, Vol. 19 (1988), No. 7, pp 516-525, (In Slovak).

[6] SAGLI, J. R., SPANGELO. I., ENGELAND, O.: Resolving Redundancy Through a Weighted Damped Least-Squares Solution. IFAC Robot Control, Viena, Austria, 1991, pp 99-104.

doc. Ing. Anton Vitko, PhD.

Prof. Ing. Ladislav Jurišica, PhD.

Ing. Andrej Babinec

Ing. František Duchoň

Ing. Marian Kľúčik

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering and information technology
Ilkovičova 3
812 19 Bratislava
Slovakia

anton.vitko@stuba.sk

Application of a Visual System for Mobile Robot Navigation (OpenCV)

Peter Pásztó, Peter Hubinský

Abstract
The article is describing the possibility of using visual systems for mobile robot navigation. It is especially intent on a circle and a rectangle sign of specific colour detection in any environment. You can find there computer vision algorithms descriptions (like histograms creating, colour filtering, circle and line detection using Hough transform) which can step by step recognize the signs in the input image scanned by a camera. The created program uses OpenCV functions and therefore is able to recognize signs in real-time. The results of the algorithms in addition to recognized signs are also information about signs (like position of the centre of the circle sign and radius or the area of the rectangle). These data can be used for robot navigation, computing the distance from the sign and so on.

Keywords: RGB model, colour filter, histogram, Hough transform, circle and line detection

Introduction

This article is describing the possibility of using visual systems for mobile robot navigation. The navigated mobile robot will not use any other sensors for scanning the environment in which he is situated. It will use only the visual information from the camera.

There will be shown various image processing algorithms that can obtain the most of visual information to navigate the robot on the basis of them. At the beginning it is necessary to define the goals of the work – to decide what information from the input image will be important.

Functions of OpenCV (**O**pen Source **C**omputer **V**ision Library) are used for the image processing [1]. OpenCV is Intel's optimized library for C/C++ and is designed for real-time applications.

Objectives

The goal is to create an application whose outputs will be enough information useable for robot navigation.

The signs which robot must detect and identify and their meanings are needed to be defined. The shape and the colour of the signs are affecting the choice of used computer vision algorithms. The colour of the signs should vary from the colours abound to make the robot work in a colourful environment.

Let's try to choose two signs at the beginning: a blue circle and a red rectangle. The target is to detect and identify them and calculate the distance from them. These data are useable for mobile robot navigation.

Problem Analysis

The sign recognition is a sequence of computer vision algorithms. There are several ways for shape recognition in an image. In this case it is advantageous to sort out only the important pixels from the image right after it had been scanned. It is possible to do that because the colours of the signs are known. There can remain some unwanted pixels in the image after filtering the red and the blue pixels because there can be other objects with the same colour in the environment. Therefore shape recognizing algorithms are also needed to be used. The sign recognition steps should be the following:

- Image filtering (blue and red pixels)
- Converting to greyscale image
- Circles and lines detection
- Forming a rectangle from the detected lines
- Calculating the distance from the signs
- Determining the location according to detected signs

The Hough transform is used for circle and line detection because this method can detect objects which can be described with an equation [3] [4]. This method has some other advantages which will be mentioned in the next chapters.

Colour filtering

Colour filtering is founded on selecting the pixels of certain colour from the image and removing every other pixel. But the colour of the sign in the scanned image is not the same on all pixels of its area. The real colour of each pixel of the sign in the scanned image depends on many factors (like illumination or camera distortion). So to sort out only the pixels of the sign (but every its pixel) a filter for a range of colours should be applied. This range of colours is also not fixed. It depends on light conditions. If the light intensity is higher the same sign is lighter in the scanned image as the sign scanned in lower light intensity.

Therefore the colour filter must adapt to light conditions and set the range of the colours according to the sign pixels

colour range to keep the sign in the image. Let's see how the colours are represented in computer to know more about how to adapt the filter's colour range.

Colour representation in RGB model and histograms

The colours in RGB model are represented by combination of three basic components (red – R, green – G, blue – B) [2]. The maximal intensity of every component creates white colour and the zero intensity of every component creates black colour. Fig. 1. is showing an RGB cube from different angles of view and with step of colour components intensities of 12.

Let's say that for example the blue sign has to be filtered out (the same will be with the red sign only the position in RGB cube will be different).

On the dependency of the light intensity the colour ranges of the filter will shift on the cube axes and also the ranges can increase or decrease. These variations are mathematically nondescript. We should look at this problem from different view.

After studying the RGB cube we can find out that the intensity of blue component is always higher than the intensity of the other components if we want to stay in the blue part of the cube. It is logical – if the colour has to be blue the blue component must dominate otherwise the colour should be different. Using this knowledge could help to set the colour ranges of the filter.

For setting the ranges of the filter histograms can be used. In our case a histogram is a bar graph which shows the quantity of every pixel of certain intensity in the image. If the image consists of colour pixels of RGB model there will be three histograms for each channel of the image. A histogram example is shown in the fig. 2. The scanned image is divided into RGB channels and a histogram is created for every channel.

The local maxima in the histograms are showing the most often occurrences of pixels of certain intensity in the image. The global maximum does not mean that pixels of this intensity are the pixels of the sign. It depends on the distance of the sign from the camera (how big area of the image the sign fills in). After having a look at the histograms the colour ranges of the filter could be any of their "knolls". By choosing the right "knoll" in each histogram the probability of filtering out the pixels of the sign is very high. The meaning of the word "knoll" is needed to be defined. It is a range in the histogram which begins in the place where the values in the histogram are starting to increase (1). Local maximum is a place where the values are starting to decrease (2). The end of the range is a place where the values are starting to increase again or the value is zero (the new range begins there) (3).

Mathematically:

$$y(i) - y(i+1) < 0 \tag{1}$$

$$y(j) - y(j+1) > 0 \tag{2}$$

$$y(k) - y(k+1) < 0,\ k > j \text{ or } y(k+1) = 0 \tag{3}$$

Using this mathematical description the sign can be filtered out from the image while trying to combine every possible range in the histograms. It can take a long time to set the filter but the knowledge about that the blue sign has its blue component higher than the other components can decrease the time of setting the filter.

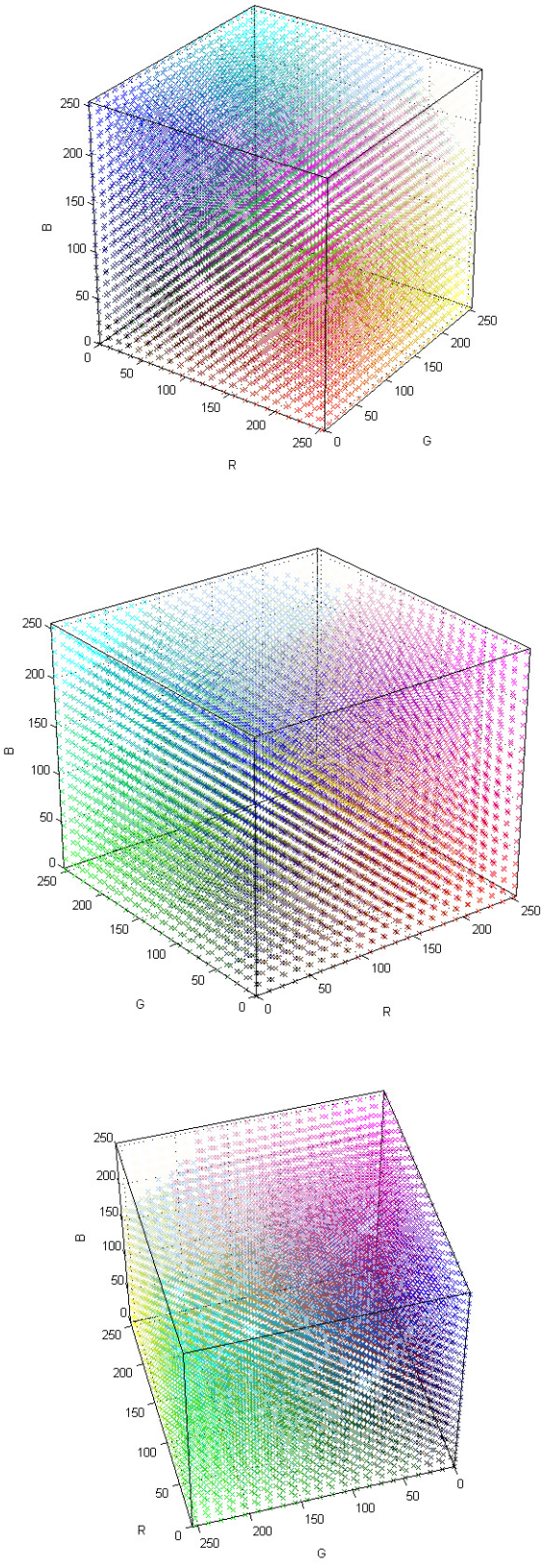


Fig. 1 RGB cube from different angles of view

In OpenCV this kind of colour filter is realisable by using the functions cvCreateHist and cvCalcHist. The output image after using colour filter is converted into greyscale image.
The application of this colour filter is shown in the fig. 3.

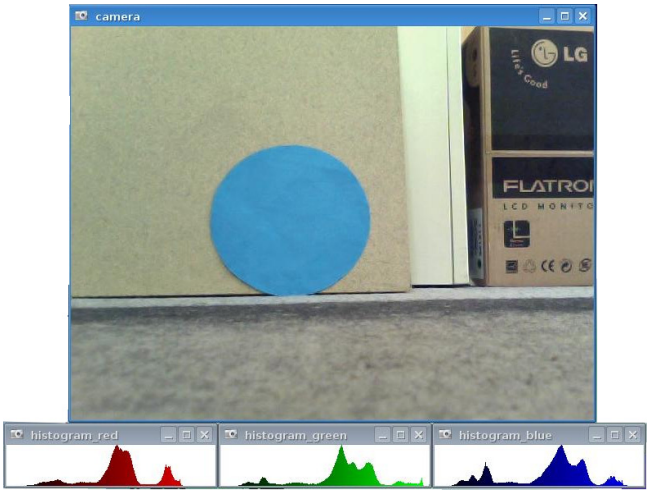


Fig. 2 Histograms for RGB channels of the image

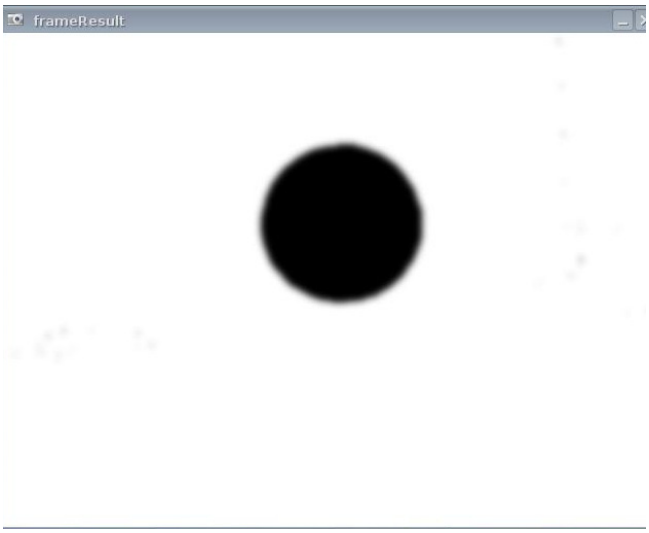


Fig. 3 Demonstration of well set colour filter

Circle and ellipse detection in the image

The colour filtering helps to detect the signs in the image and accelerates the detection because it removes the not needed pixels. The filtered image is an input for other functions for circle detection.

The circle detection is a complicated process which can be solved in several ways. One of the useful ways is edge detection from filtered image and using the Hough transform for circle detection. The advantage of using the Hough transform is that after detecting a circle the information about its location and radius is known. These data can be used for calculating the distance of the robot from the sign.

If the camera is not in the same plane as the circle sign the scanned sign in the image will be an ellipse. Therefore another problem is to recognize the ellipse in the image. After doing that the angle of the robot from the sign can be calculated on basis of the length of the ellipse axes.

In OpenCV is a function cvHoughCircles (has built in edge detection) to detect circles in the image using Hough transform. But there is not a similar function for ellipse detection. Instead of it a different method has to be used. This can be done by finding contours in the image (using function cvFindContours) and approximate the contours with an ellipse (using function cvFitEllipse). The result of circle detection is shown in fig. 4. and the result for ellipse detection is in fig. 5.

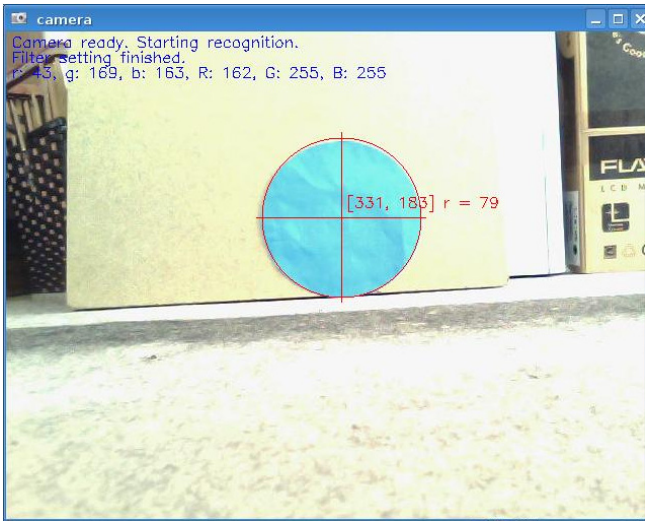


Fig. 4 Circle detection

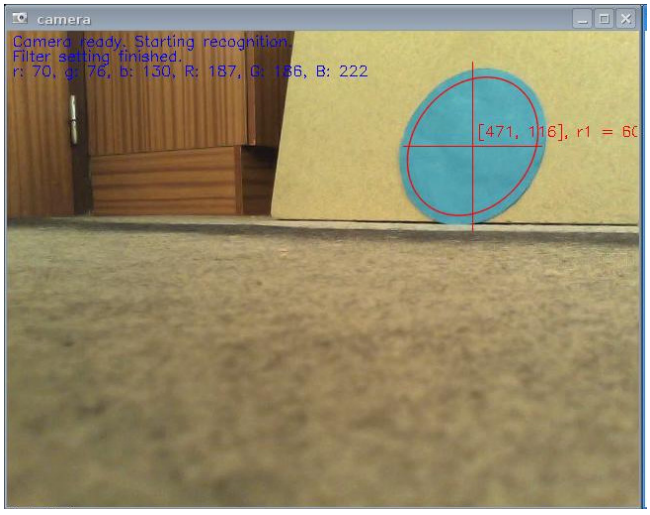


Fig. 5 Ellipse detection

Rectangle detection

The rectangle detection is the final part of this work. The pre-processing of the image is the same as it was at circle detection. So the input image is filtered and edges are detected in it. The Hough transform can also be used for rectangle detection but only for detecting the lines. After line detection in the image the lines that can form a rectangle have to be sorted out.

The mathematical equation of a line can be used for sorting out the needed lines. Every rectangle consists of four lines and each pair of lines contains right-angle. So the function for rectangle detection calculates the intersection of each line using the following equations (the coordinates of the beginning and the end of the lines are known):

$$p_1 : y_{p1} = k_{p1} x_{p1} + q_{p1}$$
$$p_2 : y_{p2} = k_{p2} x_{p2} + q_{p2} \quad (4)$$

If the lines are intersecting then $x_{p1} = x_{p2} = x_s$ and $y_{p1} = y_{p2} = y_s$. The intersection can be calculated:

$$k_{p1} x_s + q_{p1} = k_{p2} x_s + q_{p2}$$
$$x_s = \frac{q_{p2} - q_{p1}}{k_{p1} - k_{p2}} \quad (5)$$
$$y_s = k_{p1} x_s + q_{p1}$$

The coefficients k_{p1}, k_{p2}, q_{p1} and q_{p2} have to be calculated. If the begin and the end coordinates of the lines are $[p_1.x, p_1.y]$ and $[p_2.x, p_2.y]$ the line can be described with equations where these points are substituted to (coefficients k and q are the same):

$$p_1.y = k \cdot p_1.x + q$$
$$p_2.y = k \cdot p_2.x + q \Rightarrow q = p_2.y - k \cdot p_2.x$$

$$p_1.y = k \cdot p_1.x + \left(p_2.y - k \cdot p_2.x\right)$$
$$k = \frac{p_1.y - p_2.y}{p_1.x - p_2.x}$$
$$q = p_2.y - k \cdot p_2.x \quad (6)$$

On the basis of these equations the intersection of two lines can be calculated. After calculating all the intersections the Pythagoras' theorem is used to verify the lengths of the sides of the rectangle.

For line detection the OpenCV's cvHoughLines2 function is used. The result of rectangle detection is shown in fig. 6. and it also shows the circle detection.

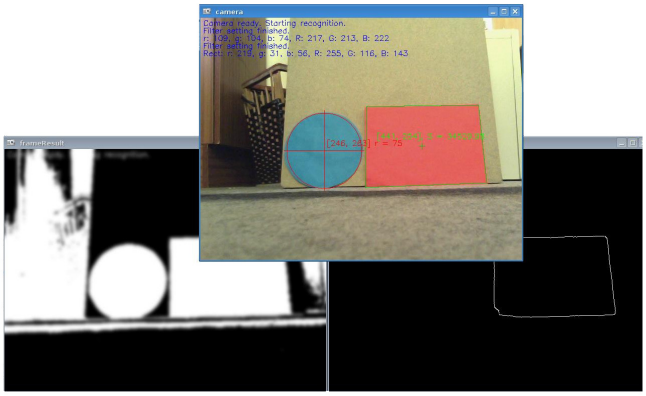


Fig. 6 Circle (red) and rectangle (green) detection

Summary

A program for sign detection was created. It can detect circle and rectangle signs and works reliably. It has an automatic colour filter setting option which uses histograms for setting.

Except of sign detection the program is also supplemented with subtasks that can compute data useful for mobile robot navigation. These data are position of the centre of the sign in the image, radius or area of the detected signs which can be used for distance calculation or angle calculation.

The pictures in this document are also the results of the experiments. They are showing that the detection works well and they contain the listing of the data useful for navigation.

By using OpenCV the program works in real-time. It is able to detect the signs in a very short time even if the camera is moving.

Acknowledgement

This work was supported by Grant Agency of Ministry of Education and Academy of Science of Slovak Republic VEGA under Grant No. 1/0690/09. The authors are pleased to the acknowledge this support.

References

[1] BRADSKI, G., KAEHLER, A., "Learning OpenCV", Sebastopol: O'Reilly Media, Inc., 2008

[2] HARGAŠ, L. – HRIANKA, M. – KONIAR, D.: Image Processing and Analysis. A Practical Approach – Text Book, Žilinská univerzita v Žiline 2008, ISBN 978-80-8070-962-4

[3] HLAVÁČ, V., SEDLÁČEK, M., "Zpracování signalu a obrazu", Praha: Vydavatelství ČVUT, 2001

[4] ŠONKA, M., HLAVÁČ, V., "Počítačové vidění", Praha: Grada, 1992

doc. Ing. Peter Hubinský, PhD.

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Department of Automation and Regulation
Ilkovičova 3
812 19 Bratislava
hubak@elf.stuba.sk

Ing. Peter Pásztó

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Department of Automation and Regulation
Ilkovičova 3
812 19 Bratislava
peterpaszto@gmail.com

Navigational maps based on the GPS data

Jaroslav Hanzel, Marián Kľúčik, Ladislav Jurišica

Abstract
The contribution deals with a problem of the autonomous robot navigation in the unstructured outdoor environment. The robot employs the navigational data from the RNDF map in the form of the GPS coordinates. The global position data is transformed to the local navigational map. The attention is devoted to the conversion of the the global WGS 84 coordinates of the waypoints to the local Cartesian coordinate system.

Keywords: autonomous mobile robot, navigation, RNDF map, GPS coordinates, reference ellipsoid, WGS 84

Introduction

Autonomous mobile robot is an object capable of motion, observation of its environment and decision making. Control system of the robot has to be able to acquire and to maintain a model of the working environment, to localize the robot in that environment, to plan a target and means of its achievement. As a navigation of the robot in the environment is considered its motion along planned trajectory and simultaneous determination of its position. Robots internal representation of the working environment from its perspective is considered as the map of the environment. Robot has to be able to determine its own position in the map, that means to solve the localization problem. It is not a trivial problem because of the uncertainty of an odometry and sensed data. Generally the robots employ two main types of the maps, the global and the local maps, which differ themselves in their properties and their applications. The global as well as the local maps represent only those aspects of the environment, which are needed to resolve the given tasks. This contribution is to be inscribed to the robot navigation in the unstructured outdoor environment with reference to global map in the form of a RNDF map with GPS coordinates. In order to navigate the robot, it is essential to transform the given map to the local robots navigation map.

1. Route Network Definition File (RNDF)

The Defense Advanced Research Projects Agency (DARPA) has been funding the development of many modern technologies with worldwide applications. The DARPA organized prize competition for autonomous vehicles known as a DARPA Grand Challenge. The third competition of the DARPA Grand Challenge, is known as the "Urban Challenge". To meet the objectives of the Urban Challenge, a team's vehicle must complete multiple missions over a so called 'route network' [4]. A 'route network' is defined as the set of roads and areas in which an autonomous vehicle may travel. In order to represent the route networks, the Route Network Definition File (RNDF) was designed.

The RNDF specifies accessible road segments and provides information such as waypoints, stop sign locations, lane widths, checkpoint locations, and parking spot locations. In addition to road segments, the RNDF specifies free-travel 'zones', that have a defined perimeter, but without any waypoints provided. Zones are used to represent parking lots and areas with moving or stationary obstacles or vehicles [5].

1.1 Road Segments

The RNDF includes one or more road segments. Eeach roadsegment comprises from one or more lanes. The basic road segment scheme is shown in Fig. 1.

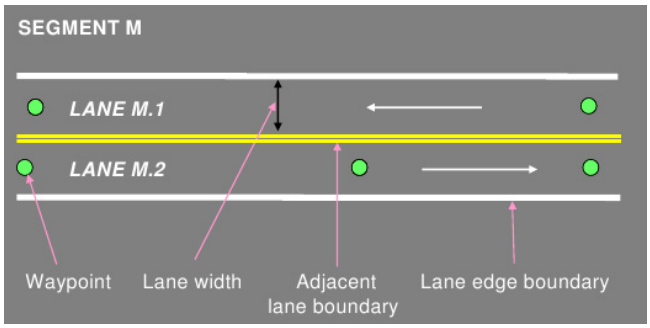


Fig. 1 The road segment M comprised of two lanes [5].

A 'segment' is characterized by the number of lanes and the street name. A 'lane' is characterized by the nominal width of the lane, the lane markings, and the ordered set of waypoints associated with the lane. The 'waypoints' are generally placed at the centre of the lane. The waypoint location in terms of the geodetical latitude and longitude are fixed point numbers with six decimal places in decimal degrees. The travel proceeds from waypoint to successive waypoint in the lane.

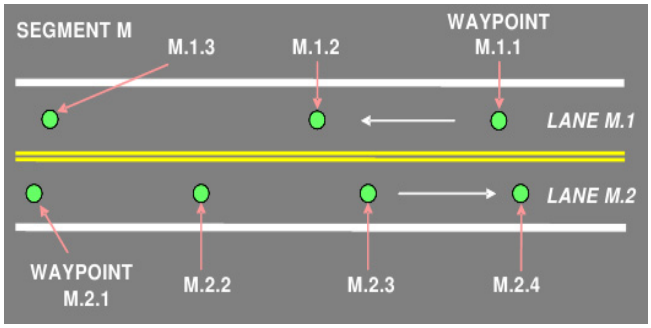


Fig. 2 Waypoint numbering scheme for segment M [5].

The numering scheme of the segments, lanes and waypoints is illustrated in Fig. 2. Every element in the RNDF is specified by a unique identifier in one of the following forms: 'M', 'M.N', or 'M.N.P', where M, N, and P are positive whole numbers. Segments are identified using the form 'M'. The Nth lane of segment M has the identifier 'M.N'. The waypoints of each lane are identified similarly. The Pth waypoint of the lane M.N has the identifier 'M.N.P'.

Entry and exit waypoints may occur at the beginning, in the middle or at the end of the lane. The Fig. 3 shows an intersection in which an exit waypoint (D) at the end of the lane 2.2 is associated with entry waypoints A and E in the lanes 1.1 and 1.2, respectively. In addition, exit waypoints B and F in the middle of the lanes 1.2 and 1.1 are associated with the entry waypoint C at the beginning of the lane 2.1. Along the travel paths in the intersection (D to A, for example), there is an implied travel lane with a width equal to the minimum of the widths of the entry and the exit lanes [5].

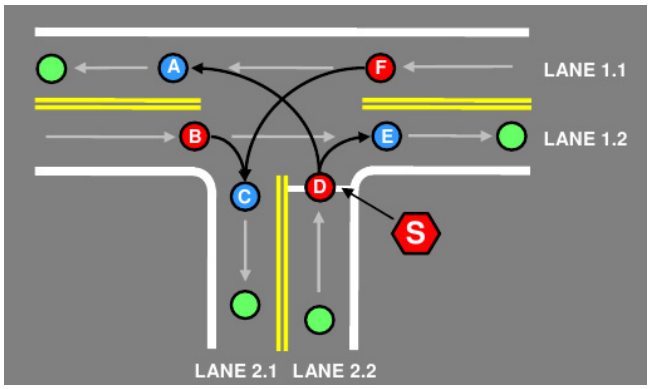


Fig. 3 Red dots are exit waypoints, blue dots are entry waypoints, and green dots are other waypoints. Waypoint D is also the stop sign [5].

In the RNDF there are indicated also 'stop signs', which are associated with the waypoints. In Fig. 3, the stop sign is associated with the waypoint D and is interpreted as a stop line passing through the waypoint D and perpendicular to the direction of travel [5].

An example of the RNDF map used at the mobile robot competition RoboTour 2007 [6] is shown at Fig. 4.

```
RNDF_name,        stromovka-rndf-0.2
num_segments,     22
num_zones,        0
format_version,   1.0
creation_date,    9/29/2006
segment, 1
num_lanes,        2
segment_name,     A
lane,     1.1
num_waypoints,    60
checkpoint,       1.1.30,  1
exit,     1.1.60,  2.1.1
exit,     1.1.60,  5.1.1
1.1.1,    50.105220,       14.427978
1.1.2,    50.105230,       14.427981
1.1.3,    50.105240,       14.427985
1.1.4,    50.105250,       14.427991
1.1.5,    50.105260,       14.427998
1.1.6,    50.105270,       14.428006
1.1.7,    50.105280,       14.428015
1.1.8,    50.105291,       14.428023
1.1.9,    50.105301,       14.428031
1.1.10,   50.105313,       14.428040
1.1.11,   50.105323,       14.428050
1.1.12,   50.105333,       14.428058
1.1.13,   50.105343,       14.428068
```

Fig. 4 A fraction of used RNDF file [6].

2. Local navigational map

For the purpose of the mobile robot navigation in the environment, the control system utilizes a local navigational map. The local navigational map is simplified two-dimensional model of the environment representation. Accordingly it is necessary to transform the GPS navigational data from the RNDF map to the local navigational map.

2.1 World Geodetic System

The GPS data is referred to unified global coordinate system named World Geodetic System 1984 (WGS 84). The WGS 84 geodetic system comprises of a standard coordinate frame for the Earth, a standard spheroidal reference surface (reference ellipsoid), and a gravitational equipotential surface (the geoid) that defines the nominal sea level. It has been in use since 1984 and the last revision took place in 2004. The WGS 84 coordinate system is a right-handed, Earth-fixed orthogonal coordinate system and it is graphically depicted in the Fig. 5.

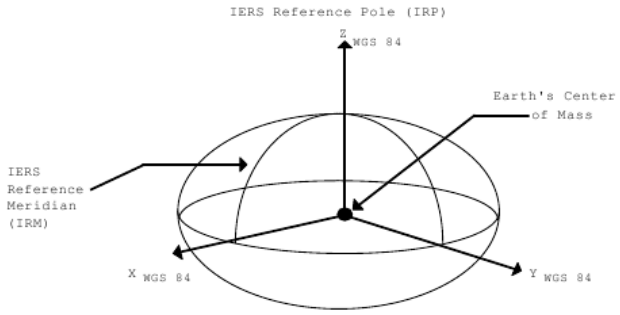


Fig. 5 The WGS 84 coordinate system definition [3].

The coordinate origin of the WGS 84 coordinate system is the Earth' centre of mass. In WGS 84, the meridian of zero longitude is an International Reference Meridian (IRM), which is defined by the International Earth Rotation and Reference Systems Service (IERS). The Z-Axis is defined by the direction of the IERS Reference Pole (IRP). The X-Axis is defined by the intersection of the IERS Reference Meridian (IRM) and the plane passing through the origin and normal to the Z-axis. The Y-Axis completes a right-handed, Earth-Centered Earth-Fixed (ECEF) orthogonal coordinate system, measured in the equator plane, 90° east of the X-axis [3].

Global geodetic applications require clear definition of three different surfaces. The first of these is the Earth's topographic surface, which includes the landmass topography as well as the ocean bottom topography. In addition to topographic surface, a definition for a geometric or mathematical reference surface, the ellipsoid, and an equipotential surface called the geoid is required.

The Earth is not a sphere, but an ellipsoid, slightly flattened at the poles and slightly bulging at the equator, which in mathematics is also called an oblate spheroid. The ellipsoid is used as a surface of reference for the mathematical reduction of geodetic and cartographic data. The World Geodetic System (WGS) represents an ellipsoid of which placement, orientation, and dimensions best fit the Earth's equipotential surface coinciding with the geoid. The system was developed from a worldwide distribution of terrestrial gravity measurements and geodetic satellite observations.

The WGS 84 Reference Ellipsoid is defined by the four defining parameters: semi-major axis (a), flattening (f), angular velocity of the Earth (ω) and Earth's gravitational

constant (GM). The defining parameters values are listed in Tab. 1 [3].

Parameter	Notation	Value
Semi-major Axis	a	$6378137.0m$
Reciprocal value of Flattening	1/f	298.257223563
Angular Velocity of the Earth	ω	$7292115.0\times10^{-11} rad/s$
Earth's Gravitational Constant (Mass of Earth's Atmosphere Included)	GM	$3986004.418\times10^{8} m^3/s^2$

Tab. 1 WGS 84 four defining parameters [3].

For our purposes it will be sufficient to consider geometric parameters only. Mathematically, a reference ellipsoid is an oblate spheroid with two different axes: an equatorial radius (the semi-major axis a), and a polar radius (the semi-minor axis b). The relation between two axes *a* and *b* is given by following formula:

$$f = \frac{a-b}{a} \tag{1}$$

The numerical value of the the semi-minor axis for the WGS 84 reference ellipsoid is *b* = 6356752.3142 m [3].

2.2 The position data transformation

The goal of the transformation process is to calculate position data of the waypoints from the WGS 84 coordinate system to position data of the waypoints referred to the local coordinate system (LCS) utilized by control system of the mobile robot. The relation between the global WGS coordinate system and local coordinate system is depicted in the Fig. 6.

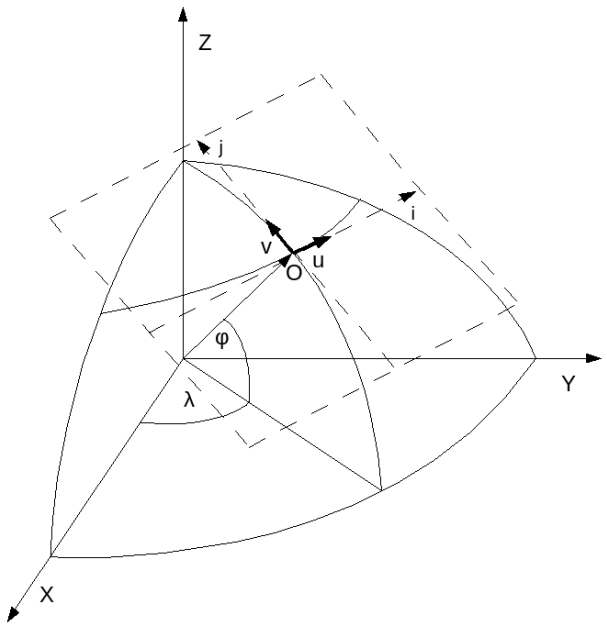


Fig. 6 The WGS 84 coordinate system and the local coordinate system.

The local coordinate system has the form of two dimensional Cartesian coordinate system in the Euclidean plane. This plain is defined as the tangent plane to the surface of the reference ellipsoid at the chosen point. This point also serves as the origin of the local coordinate system (Fig. 6).

The first step in the transformation of the GPS coordinate data from the RNDF map, it is necessary to select the origin of the local coordinate system appropriately according to the given data set. The point named *O* with the coordinates of the mean value of the WGS spherical coordinates (φ,λ) of the all waypoints *WP* was chosen as the convenient origin. Its coordinates are given as follows:

$$O_{(\varphi,\lambda)} = \left(\sum_{1}^{N} (P_\varphi) \Big/ N, \sum_{1}^{N} (P_\lambda) \Big/ N \right), \tag{2}$$

where *N* is number of waypoints, φ is the geodetical latitude and λ is the geodetical longitude.

Now it is necessary to computate the three dimensional geocentric rectangular coordinates (*x*, *y*, *z*) of the origin of the local system in regard to WGS defined at Fig. 6. The equations for the coordinates calculation are given as follows [3]:

$$x = (N+h)\cos\varphi\cos\lambda \tag{3}$$

$$y = (N+h)\cos\varphi\sin\lambda \tag{4}$$

$$z = \left(\left(b^2/a^2\right)N + h\right)\sin\varphi\,, \tag{5}$$

where *a* is semi-major axis and *b* is the semi-minor axis of the reference ellipsoid, the *h* denotes the geodetic height (height relative to the ellipsoid), *N* denotes the radius of curvature in the prime vertical. It is defined by the equation:

$$N = \frac{a}{\sqrt{1-e^2\sin^2\varphi}}, \tag{6}$$

where e^2 denotes the square of the first ellipsoidal eccentricity, which can be computed by the following equation:

$$e^2 = \frac{a^2-b^2}{a^2}. \tag{7}$$

It is also one of the WGS 84 ellipsoid derived geometric constants with value $e^2 = 6.69437999014\times10^{-3}$ [3].

After definition of the local coordinate system origin, the second step is to define the equation of the LCS plane in rectangular geocentric coordinates. The general expression of the plane is given as follows [1]:

$$Ax + By + Cz + D = 0\,, \tag{8}$$

where the constants *A*, *B*, *C* are the coordinates of the *normal vector* of the plane. As the LCS plane is also the tangent plane of the reference ellipsoid, the vector (*A*, *B*, *C*) is perpendicular to surface of the ellipsoid in the origin of the local coordinate system *O*. So the normal vector of the plane can be computed as the gradient of the ellipsoid at the point *O*. The mathematical expression of the WGS 84 reference ellipsoid (oblate spheroid) in the Cartesian coordinate system is given by equation [1, 2]:

$$F(x,y,z): \quad \frac{x^2}{a^2}+\frac{y^2}{a^2}+\frac{z^2}{b^2}=1\,, \tag{9}$$

where *a* is semi-major axis and *b* is the semi-minor axis of the reference ellipsoid. The gradient of the ellipsoid is given:

$$\nabla F(x,y,z)=\left(\frac{2x}{a^2},\frac{2y}{a^2},\frac{2z}{b^2}\right). \tag{10}$$

Finally the equation of the LCS plane is given as follows [2]:

$$\frac{O_x x}{a^2}+\frac{O_y y}{a^2}+\frac{O_z z}{b^2}=1\,. \tag{11}$$

The third step is to define the right-handed orthonormal Cartesian coordinate system in the LCS plane. So it is necessary in the LCS plane to define two reciprocally perpendicular vectors *u*, *v* of the length of 1 (m) at the LCS origin *O*. First there are two points defined at the ellipsoid surface with the positions exactly at east and north directions in regard of origin *O* as follows:

$$UG(\varphi,\lambda)=(O_\varphi,O_\lambda+c), \tag{12}$$

$$VG(\varphi,\lambda)=(O_\varphi+c,O_\lambda)\,. \tag{13}$$

The constant *c* is optional angular distance of the suitable scale. The value of *c* = 0.1 was determined as sufficient. By the application of the equations 3, 4, 5 the WGS Cartesian coordinates (UG_x, UG_y, UG_z) and (VG_x, VG_y, VG_z) of the points *UG* and *VG* are calculated. It is essential to compute the projection of these two points *UG* and *VG* to the LCS plane. The two lines *lu* and *lv* perpendicular to system plane passing through the points *UG* and *VG* are defined. The parametric expressions of the lines *lu* and *lv* with the directional vector (*A*, *B*, *C*) passing through points *UG* and *VG* is given as follows, for $t \in R$:

$$lu(t): \begin{aligned} x &= UG_x + At \\ y &= UG_y + Bt \\ z &= UG_z + Ct \end{aligned}, \tag{14}$$

$$lv(t): \begin{aligned} x &= VG_x + At \\ y &= VG_y + Bt \\ z &= VG_z + Ct \end{aligned}. \tag{15}$$

It is necessary to compute the intersection points of the lines *lu* and *lv* with the LCS plane. We denote these intersects as *UL* and *VL*. The vectors defined by these points and the origin *O* determinate the directions of the positive half–axes of the coordinate system in the LCS plane. To obtain unit vectors, the normalization of these vectors is required. The orthonormal basis vectors *u* and *v* of the LCS in regard to the WGS are now given by following equations:

$$u(x,y,z)=\frac{UL-O}{|UL-O|}, \tag{16}$$

$$v(x,y,z)=\frac{VL-O}{|VL-O|}, \tag{17}$$

where |*x*| denotes the length of the vector *x*. These two vectors *u* and *v* defines the transition from one coordinate system to another and the local coordinate system is completely defined.

Now it is possible to proceed to the last step of the waypoints coordinates transformation process. It is necessary to perform the same procedure as in case of the calculation of the vectors *u* and *v*. For each waypoint *WP* in input data set, by the application of the equations 3, 4, 5, the WGS Cartesian coordinates (WP_x, WP_y, WP_z) are calculated. Then follows the calculation of the coordinates of the perpendicular projections of these points to the LCS. The parametric equation of the line perpendicular to the LCS and passing through the point *WP* for the parameter $t \in R$ is given as follows:

$$lwp(t): \begin{aligned} x &= WP_x + At \\ y &= WP_y + Bt \\ z &= WP_z + Ct \end{aligned}. \tag{18}$$

The coordinates of the point *WPL*, given by the intersection of the LCS plane and the line *lwp* are calculated by application of the formulas 8 and 18. Finally the vector *wp* is computed by the application of the following equation:

$$wp(x,y,z)=WPL-O\,. \tag{19}$$

To determine the LCS 2D coordinates of the point *WPL* lying in the LCS plane, it is necessary to find the expression of the vector *wp*, through the linear combination of the basis vectors *u* and *v*. The vector *wp* can be expressed by means of *u* and *v*:

$$wp(x,y,z)=\alpha u(x,y,z)+\beta v(x,y,z)\,. \tag{20}$$

The local 2D coordinates of the *WPL* point are directly given by the computation of the coefficients *α* and *β*:

$$wp(i,j)=(\alpha,\beta)\,. \tag{21}$$

This procedure is gradually applied to all waypoints in the given RNDF map. The resulting 2-D map of the road network is depicted in Fig. 7.

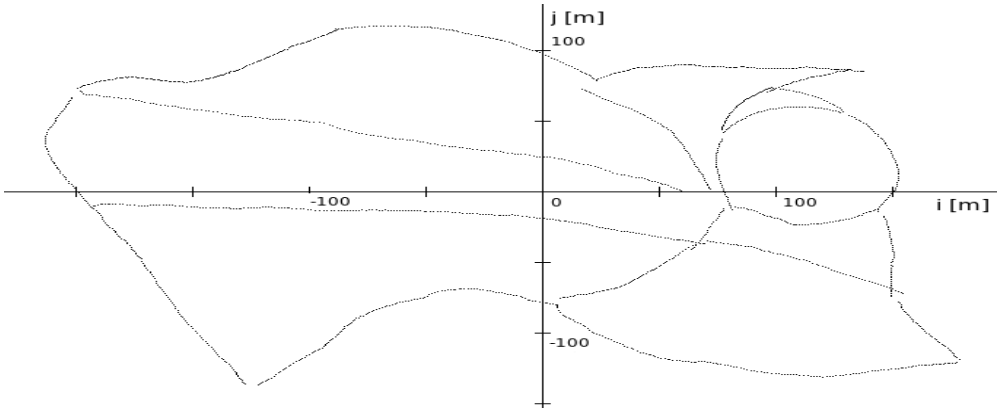


Fig. 7 The local navigation map based on the RNDF data [6] for the contest Robotour 2007.

Conclusion

The aim of the submitted contribution was to make a proposal of the procedure for the transformation of the position data in the form of the GPS coordinates to the local navigational map. This map is used in order to control the motion of the mobile robot along the given trajectory. The navigational map has the form of two-dimensional Cartesian coordinate system. The transformation is based on the mathematical model of the Earth, the WGS 84 reference ellipsoid. The proposed procedure was also successfully verified on the real data.

Acknowledgment

The work has been supported by VEGA grant 1/0690/09 and grant VMSP-P-0004-09. This support is very gratefully acknowledged.

References

[1] Bubeník, F., Pultar, M., Pultarová, I.:Matematické vzorce a metody, ČVUT Praha, 2001, ISBN 80-01-01643-9.

[2] Bartsch, H., J.: Matematické vzorce, Mladá fronta Praha, 1996, ISBN 80-204-0607-7.

[3] NIMA - National Imagery and Mapping Agency, DEPARTMENT OF DEFENSE WORLD GEODETIC SYSTEM 1984, Its Definition and Relationships with Local Geodetic Systems, NIMA TR8350.2, Third Edition, Amendment 2, 2004, http://earth-info.nga.mil/GandG/publications/tr8350.2/tr8350_2.html.

[4] DARPA Urban Challenge, 2007, http://www.darpa.mil/grandchallenge/index.asp.

[5] Route Network Definition File (RNDF) and Mission Data File (MDF) Formats, 2007, http://www.darpa.mil/GRANDCHALLENGE/docs/RNDF_MDF_Formats_031407.pdf.

[6] Resources for Robotour 2007 contest http://robotika.cz/competitions/robotour/2006/src/stromovka-rndf.txt.

Ing. Jaroslav Hanzel, PhD.

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of control and industrial informatics
Ilkovičova 3
812 19 Bratislava
Tel.: +421 (2) 60 291 864
E-mail: jaroslav.hanzel@stuba.sk

Ing. Marián Kľúčik

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of control and industrial informatics
Ilkovičova 3
812 19 Bratislava
Tel.: +421 (2) 60 291 605
E-mail: marian.klucik@stuba.sk

prof. Ing. Ladislav Jurišica, PhD.

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of control and industrial informatics
Ilkovičova 3
812 19 Bratislava

New robust MRAS-based method for sensorless vector control of Induction Machines

Jakub Vonkomer, Milan Žalman

Abstract
Sensorless vector control is an essential feature of every modern AC drive. Most methods are using voltage models for flux estimation, which suffer from pure integration problems. In this paper a novel robust MRAS-based speed estimator based on error between current estimators output and measured current is presented. Theoretical analysis of estimators and design of adaption loop are described as well. Finally, simulation and experimental results are presented.

Keywords: sensorless control, vector control, model reference adaptive system, speed estimator, induction machine

Introduction

As known, induction motors are still the most commonly used motors in the industry, thanks to its good performance and low cost.
Kinds of dynamic control of Induction machines, like Vector Control (Field Oriented Control) or Direct Torque Control (DTC) have become standard feature of industrial AC drives years ago. Original structures require using of mechanical speed sensors, but even at that time, the need for mechanical speed sensor elimination was important. During the last two decades many and many different methods and algorithms of speed estimation for vector control have been published. I will focus on adaptive structures – Model Reference Adaptive Structure (MRAS), which use two different observers. Usually these observers are using known voltages and measured stator currents in less or more complicated mathematical models together with motor parameters to estimate the rotor speed.

1. MRAS structures

MRAS (Model Reference Adaptive System) estimators are usually observing mechanical speed by using two different estimators or observers which one of them is speed dependent. Basic schema of MRAS model is shown on figure 1.
The difference between estimators outputs is used for speed error reduction, commonly by PID controller, but it is worth to mention that in the recent years many papers describing adaptation by Fuzzy Logic or Artificial neural networks have been published.

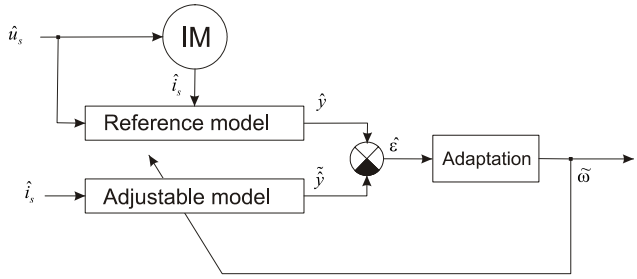


Fig. 1 General schema of MRAS system

In general MRAS models can be divided into three main groups[5]:
1. Models based on error between two models of magnetic flux. They are „classical“ well-known structures. Schauder's MRAS[4]is typical example of this group. Because these structures use voltage flux model, overall quality of the integrator implementation and parameter sensitivity are main issues. Ohtani[6] describes torque based MRAS with better dynamic behaviour and less sensitivity to parameters variation, mostly stator resistance.
2. MRAS based on models of back-emf, error vector is computed between measured and computed back-emf of motor.
3. Structures based on the error of stator current. Stator current vector is observed by stator current observers and then compared to measured values. In MRAS terminology, reference model is the real motor. These structures are not very well-know, therefore a focus on them will be put in this paper.

As mentioned above, main advantage of this structure is absence of back-emf model which is unreliable and therefore unusable at speeds close or equal to zero. Proposed stator current error based MRAS will be later marked as $MRAS^C$ in this paper.

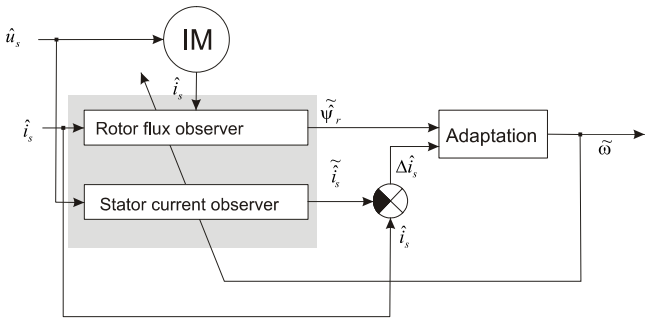


Fig. 2 Simplified schema of proposed MRAS^C system

2. Mathematical model of the estimator

As seen, proposed method uses two main speed dependent observers. First is model of stator current of IM. The second one is model of rotor magnetic flux. Both the models are interconnected - model of current uses the results of flux model.

2.1 Model of stator current of Induction machine

This model is derived from the basic equations of Induction Machine[2]

$$\frac{d\hat{i}_s}{dt} = \frac{1}{\sigma L_s}\left(\hat{u}_s - R_1\hat{i}_s + \frac{L_m}{L_r}\frac{R_r}{L_r}\hat{\psi}_r - j\omega\frac{L_m}{L_r}\hat{\psi}_r\right) \quad (1)$$

,where $R_1 = R_s + R_r\frac{L_m^2}{L_r^2}$ and $T_1 = \frac{\sigma L_s}{R_1}$ and $k_r = \frac{L_m}{L_r}$

Equation (1) can be further rewritten to Laplace transform and simplified:

$$\hat{i}_s = \frac{1}{R_1(T_1 s+1)}\left(\hat{u}_s + \frac{k_r}{T_r}\hat{\psi}_r - j\omega k_r\hat{\psi}_r\right) \quad (2)$$

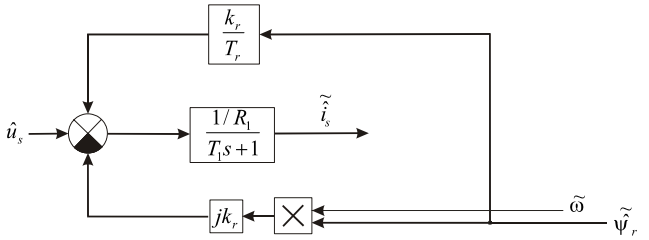


Fig. 3 Model of stator current

2.2 Current model for rotor flux estimation[3]

This model is based on equation originally formulated by Blaschke (often known as Blaschke equation)[1]. In the low speed region, flux components can be obtained more easily with currents and speed signals. This model is mostly used in closed-loop systems where rotor speed is known or at speeds close to zero thanks to its stability. Its main advantage comparing to voltage model of rotor flux is absence of pure integrators.

Current model for flux estimation in stator reference frame (α,β) :

$$s\hat{\psi}_r = -\frac{1}{T_r}\hat{\psi}_r + j\omega\hat{\psi}_r + \frac{L_m}{T_r}i_s \quad (3)$$

In reference frame oriented to rotor flux (1,2):

$$\hat{\psi}_r = \frac{L_m}{T_r s+1}i_s \quad (4)$$

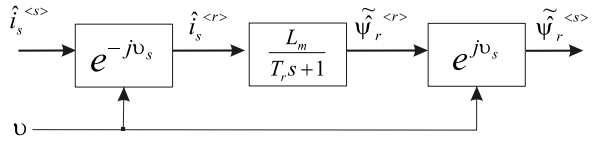


Fig. 4 Same model of flux in reference frame oriented to rotor position (d,q)

Using estimator oriented to rotor position gains several advantages over estimator in stator reference frame. Main advantage is the decoupling of both (α,β) components and processing of signals rotating at the slip frequency only, which is usually not more than couple of Hz. Furthermore park transform is non-dynamical transform.

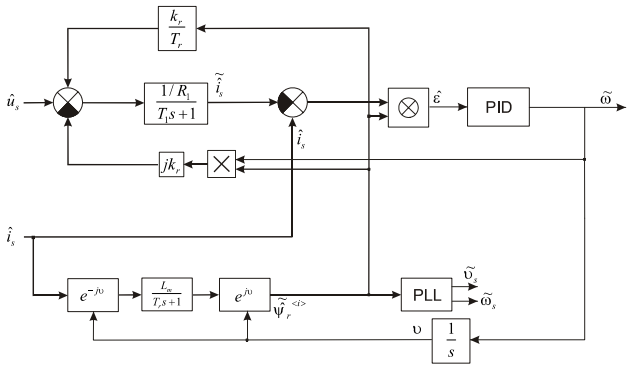


Fig. 5 Schema of final estimator

Error signal is then defined as:

$$\varepsilon = \Im\left(\tilde{\hat{\psi}}^*_r \cdot \hat{e}_{is}\right) = \left(i_{s\alpha} - \tilde{\hat{i}}_{s\alpha}\right)\tilde{\psi}_{r\beta} - \left(i_{s\beta} - \tilde{\hat{i}}_{s\beta}\right)\tilde{\psi}_{r\alpha}$$

(5) $\hat{e}_{is} = \hat{i}_s - \tilde{\hat{i}}_s$

(6)

2.3 Design of the adaptation loop

Adaptation loop has been designed from the linearized mathematical model.

Current model for flux estimation:

$$s\tilde{\hat{\psi}}_r = \frac{L_m}{T_r}\hat{i}_s - \frac{1}{T_r}\tilde{\hat{\psi}}_r - j\omega\tilde{\hat{\psi}}_r \quad (7)$$

$$\Delta\tilde{\hat{\psi}}_r = \frac{jT_r\tilde{\hat{\psi}}_{r0}}{(T_r s+1-jT_r\omega_{00})}\Delta\omega \quad (8)$$

Model of stator current:

$$\tilde{\hat{i}}_s = \frac{1}{R_1(T_1 s+1)}\left(\hat{u}_s + \frac{k_r}{T_r}\tilde{\hat{\psi}}_r - j\tilde{\omega}k_r\tilde{\hat{\psi}}_r\right) \quad (9)$$

$$\Delta\tilde{\hat{i}}_s = \frac{1}{R_1(T_1 s+1)}\left(\frac{k_r}{T_r} - jk_r\omega_{00}\right)\Delta\tilde{\hat{\psi}}_r - jk_r\tilde{\hat{\psi}}_{r0}\Delta\tilde{\omega} \quad (10)$$

By combining together equations (8) and (10) together, following results are obtained:

$$\Delta\hat{i}_s = \frac{1}{R_1(T_1 s+1)}\left(\frac{-T_r s(T_r s+1+jT_r\omega_{00})}{(T_r s+1)^2 + T_r^2\omega_{00}^2}\right)jk_r\tilde{\hat{\psi}}_{r0}\Delta\omega \quad (11)$$

$$\Delta\varepsilon = \Im\left(\tilde{\hat{\psi}}^*_{r0}\cdot\Delta\tilde{\hat{e}}_{is} + \Delta\tilde{\hat{\psi}}^*_r\cdot\tilde{\hat{e}}_{is}\right) \quad (12)$$

$$\Delta\varepsilon = \Im\left(\hat{\tilde{\psi}}^{*}_{r0}\cdot\Delta\hat{\tilde{e}}_{is}\right) = -\Im\left(\Delta\hat{i}_{is}\right)\hat{\tilde{\psi}}^{*}_{r0} \tag{13}$$

$$\Delta\varepsilon = \frac{T_r s\left(T_r s+1\right)}{R_1\left(T_1 s+1\right)\left(\left(T_r s+1\right)^2 + T_r{}^2\tilde{\omega}_{00}{}^2\right)} k_r \tilde{\psi}_{r0}{}^2 \Delta\tilde{\omega} \tag{14}$$

2.3 Adaptation loop controller

PID controller has been chosen as most appropriate controller structure. Parameters have been designed using Inverse Dynamic method.

By assumption in (15), a simple 2nd order transfer function is obtained which can be easily modified to satisfy the form (17):

$$\frac{\left(T_r s+1\right)}{\left(T_1 s+1\right)} \approx 1 \tag{15}$$

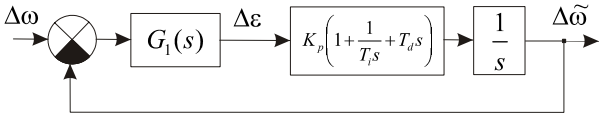


Fig. 6 Control loop for controller design

$$G_1(s) = \frac{\Delta\varepsilon}{\Delta\tilde{\omega}} \tag{16}$$

Following bode plots demonstrate the comparison of original and simplified transfer function after assumption in eq. 15.

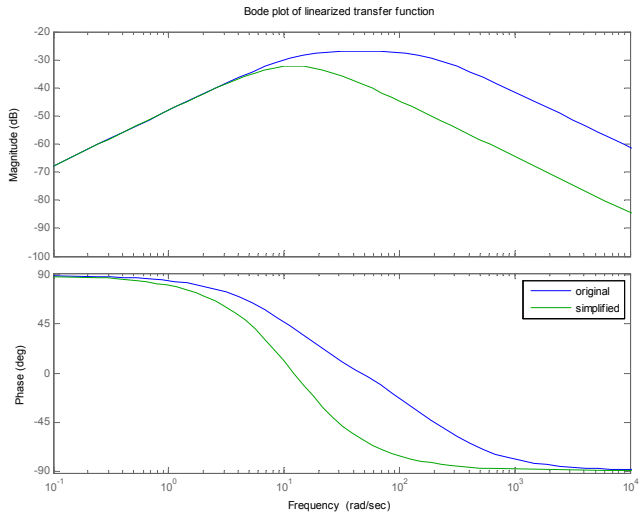


Fig. 7 Bode plot of original transfer function and the simplified transfer function used for controller design

This transfer function derived in (14) can be easily modified to satisfy following form[3]:

$$G(s) = \frac{K}{T^2 s^2 + 2bTs + 1} \tag{17}$$

In this case it is very convenient to use Inverse Dynamics method to design the controller. Inverse Dynamics is a robust method, providing aperiodic step response which is very helpful in this application. Different to Pole-Placement method where designing the poles, but zeroes can add undesirable dynamic to the system.

By using mentioned method following formulas of PID parameters are gained:

$$K_p = \frac{T_i}{K\cdot T_w} \qquad T_i = 2bT \qquad T_d = \frac{T}{2b} \tag{18}$$

T_w – required dynamics (time constant)

Final simplified schema of the adaptation controller is shown on the following figure (Fig. 8).

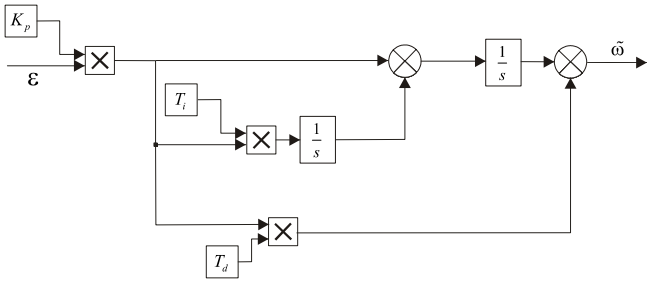


Fig. 8 Designed PID controller

3. Results

Several experiments have been done in Simulink. By building a close-to-reality simulation model in Simulink with intensive using of SimPowerSystems toolbox blocks, experimental test to parameter variation has been done. This test is a complex test of behaviour, due to fact that it does not only include stability analysis of the adaptation control loop, but effects of the parameter variation on the other loops, including current and flux controllers as well.

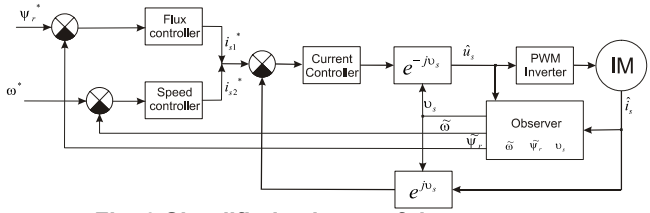


Fig. 9 Simplified schema of the structure of tested system

3.1 Parameter variation effects

Following table demonstrates the overall performance and stability of entire systems due to variation of electrical parameters of the motor.

Tab.1 Experimental analysis of motor's electric parameters variation

		Current-based MRAS, Tsam = 0.4ms										
		Parameter variation										
Par.	Speed	-50%	-40%	-30%	-20%	-10%	0%	10%	20%	30%	40%	50%
Rs	< 100%											
	< 50 %	X								R	R	R
	< 5%	X										
Rr	< 100%										X	X
	< 50 %										X	X
	< 5%											
Lm	< 100%	X	X									
	< 50 %	X	X									
	< 5%											
σ.Ls	< 100%											
	< 50 %	R	R									
	< 5%	R	R									

Legend

	No significant change
	speed error less than 5%
X	speed error less than 5%, torque ripples
R	minimal speed error, not able to fast speed reversation
	speed error higher than 5%
X	speed error higher than 5%, torque ripples
	edge of stability, significant speed error
	unstability

3.2 Benchmark test, simulation

Test consists of a simple speed reference signal containing several steps. First is step to 50Hz, then reverse to -50Hz followed by steps to lower and lower frequency down to zero to explore dynamic behaviour of the system at various speeds.

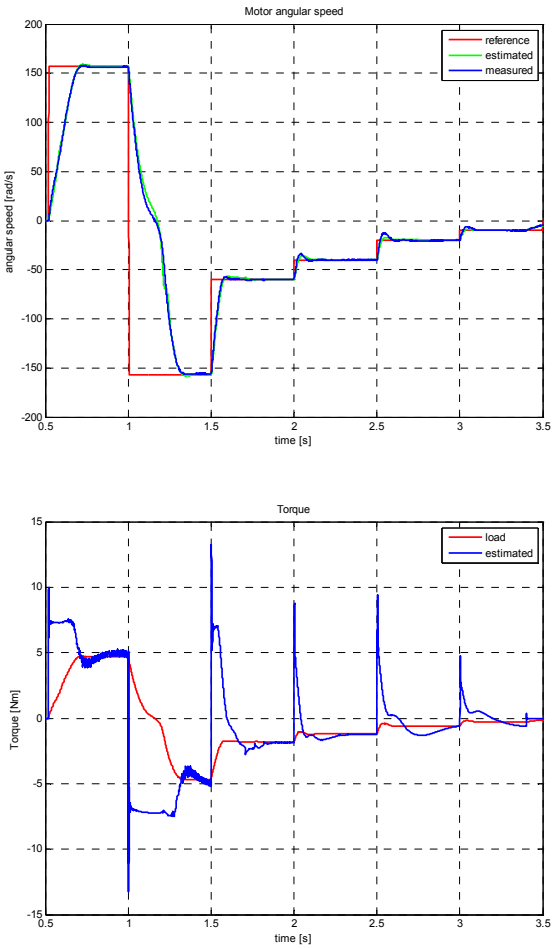
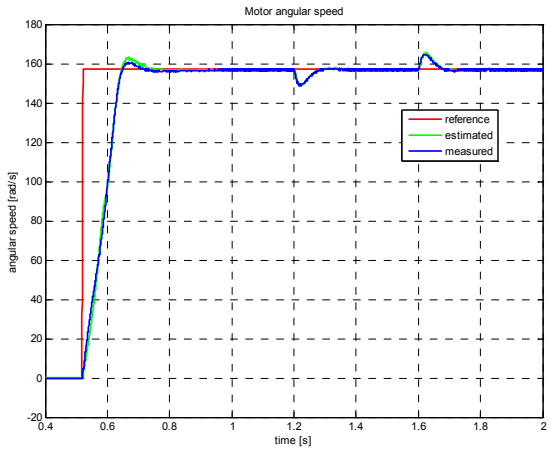


Fig. 10 Benchmark test: speed and torque

This test shows overall performance of proposed method, including speed reverse possibilities and operation in the low speed region at the end.

3.3 Dynamic load change transient test in high speed operation

Test contains high speed operation at 50Hz followed by load change transients in time 1.2 and 1.6 respectively.



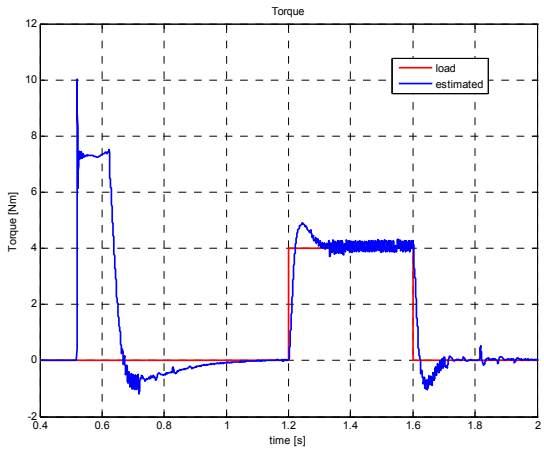


Fig. 11 High speed operation: speed and torque

Figure 11 proves very good dynamic response of MRAS^{C} as well.

3.4 Dynamic load change transient test in low speed operation

Following test signal demonstrates operation at 5 rad/s, then fast reverse to -5 rad/s. Furthermore, there is a load change in time (1.2 s to 1.5 s) of 1.5 Nm in motor mode of operation and generator mode of operation in (2.2 s to 2.5 s)

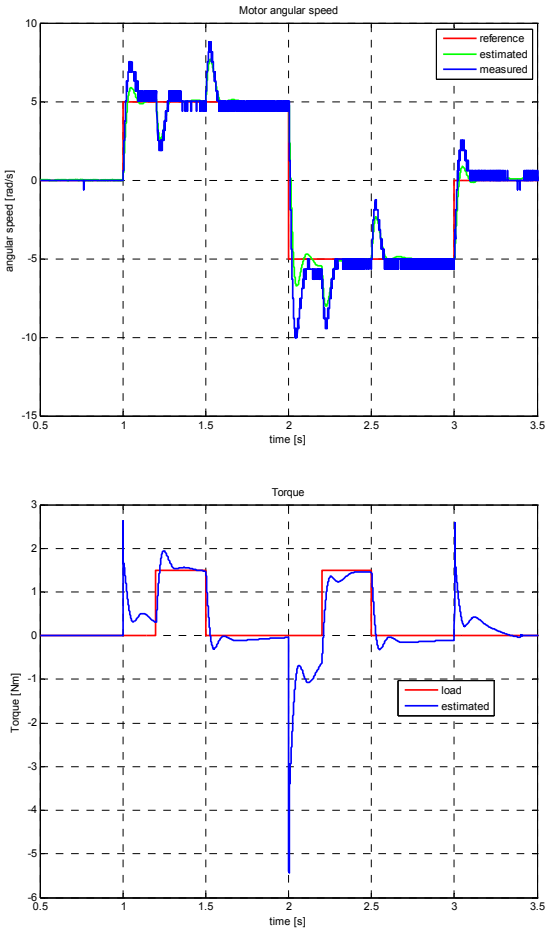


Fig. 12 Low speed operation: speed and torque

Plots show the excellent dynamic response of proposed sensorless speed method in low speed region.

Conclusion

Not very well known approach of MRAS-based sensorless speed control method has been demonstrated. MRAS^C shows very good dynamic response and appears to be very immune to parameters variation, which is necessary in the industry, where stability and robustness against disturbances is essential. The mathematical model of the estimator including the design of adaptation loop controller has been described.

Acknowledgement

This work has received support from the Slovak Research and Development Agency. Project reference number is APVV-0530-07.

Appendix

Motor data (rated values) – wye connection used
P_n = 1.1 kW
U_n = 230/400 V (delta, wye)
I_n = 5 / 2.9 A
N_n = 1380 min^{-1}
Pole pairs = 2

Electric parameters
R_s = 7.66 Ω
R_r = 5.12 Ω
L_m = 0.386 H
L_r = 0.421H
L_s = 0.421 H
J = 0.005 $kg.m^2$

References

[1] Bimal K. Bose, Modern power electronics and AC drives USA, Prentice Hall, 2002

[2] Žalman M.: Akčné členy (Slovak language), STU Bratislava 2003

[3] Milan Žalman – lectures for Intelligent Servosystems, 2007/2008.
http://servo.urpi.fei.stuba.sk/index.php?option=com_content&task=view&id=15&Itemid=26

[4] Schauder, C., "Adaptive speed identification for vector control of induction motors without rotational transducers," Industry Applications, IEEE Transactions on , vol.28, no.5, pp.1054-1061, Sep/Oct 1992
URL:
http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=158829&isnumber=4088Asdas

[5] Dybkowski, M.; Orlowska-Kowalska, T., "Application of the stator current-based MRAS speed estimator in the sensorless induction motor drive," Power Electronics and Motion Control Conference, 2008. EPE-PEMC 2008. 13th , vol., no., pp.2306-2311, 1-3 Sept. 2008
URL:
http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=4635607&isnumber=4635237

[6] Ohtani, T.; Takada, N.; Tanaka, K., "Vector control of induction motor without shaft encoder," Industry Applications, IEEE Transactions on , vol.28, no.1, pp.157-164, Jan/Feb 1992
URL:
http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=120225&isnumber=3434

Ing. Jakub Vonkomer
prof. Ing. Milan Žalman, PhD.

Slovak Technical University, Bratislava
Faculty of electrical engineering and information technology,
Institute of Control and Industrial Informatics
Ilkovičova 3
Bratislava 1
SK - 812 19
Slovak Republic
E-mail: jakub.vonkomer@stuba.sk, milan.zalman@stuba.sk

Master-slave position servo-drive design of aircore linear motor with permanent magnets

Tatiana Radičová, Milan Žalman

Abstract
Besides that establishment of linear motors on market was in reality pretty slow, it can be said that already nowadays there are linear motors relatively extended. Thanks belong to their extraordinary features like high dynamics, high position accuracy and especially during the operation there is no frictional force developed what contribute to increasing the lifetime of the motor, of course. A very important role performs correct control design of the linear drive. That is why we focus on this field in the following lines. Specifically, we will present positive effect of PID algorithm improved with the lead compensator and precorrection constants to the control performance and control accuracy of position servo-drive LMPM.

Key words: PID algorithm, lead compensator, Master slave control, position servo-drive, Luenberger observer

Introduction

Nowadays, linear motors types are still more and more preferred thanks to their matchless features. Although in comparison with rotary motors, they are henceforward financially demanding. It can be mentioned that a high speed trains, such as Maglev or Trans-rapid became famous by implementation these kind of motors in it. Linear motors however occur in various sectors, whether in electrotechnical or electronic production (drives for operating and position engineering, drives into machine tools,…).

In this paper we will pay attention to position servo-drive control design of aircore LMPM in consideration of control performance. Control performance possesses highly important function in servo-drives. To achieve the best control performance precorrection is integrated into the original PID algorithm with lead compensator. Compared to the article [5] our asset is to presume positive effect of precorrection (realized by Master-slave generator) and Luenberger observer.

1. Current state of linear servo-drive

1.1 Design of the motor

A linear motor is substantially a standard synchronous or induction motor developed into the plane as can be seen in the figure below.

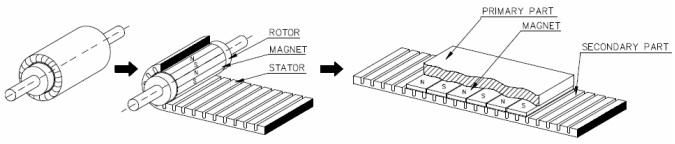


Fig.1 Basic structure of linear motor

Primary part (stator) is formed by ferromagnetic stack composed of laminations and by 3-phase winding inserted in its slots.

Secondary part (rotor) is usually longer part, where in synchronous motors is designed by rare-earth permanent magnets (Nd-Fe-B) and in induction motors is formed by the squirrel cage located either in the slots of the ferromagnetic stack, or at least fixed on the steel base of the driven device.

Most often the primary part is moving along a travel being formed by an arbitrary number of secondary parts[2].

1.2 Advantages of aircore LM PM

- No Attractive Forces - Because the forcer contains no iron, there exist no attractive forces between the forcer and the rails. This means no additional forces on bearings. The motor is easier to handle and install without these attractive forces.
- No Cogging - With its ironless forcer, this style motor has no cogging. This is ideal in applications requiring extreme velocity control. This type of motor is normally used in conjunction with air bearings due to the air-bearing's "frictionless" / ultra smooth characteristics.
- Low Weight Forcer - These forcers have low weight. In applications that have very light payloads, this can be a benefit. Higher acceleration / deceleration may be possible due to this lower weight, which of course results in higher throughput.

1.3 Disadvantages of aircore LM PM

- Heat Dissipation - Since the forcer is made of wound coils and held together with epoxy, the heat must leave the coils by traveling up the coil to the aluminum mount plate and out to a heat-sink. Heat also passes through the air gap and into the magnet rail. Both of these paths have high thermal resistance's and thus make thermal management of the motor difficult.
- Structural Stiffness - The forcer is made of coils and epoxy. The force is generated at the coil. This means that all of the exerted force is on the windings and ep-

oxy. This is a weak structure as compared to the iron-core. This weakness limits the maximum sizes and forces to which these types of motors can be manufactured without additional structural members being added.
- Force per Package Size - Due to the thermal and structural limitations, the force per package size of this type of motor is low. In addition, the double rail design, also takes up additional space.

2. Position servo-drive design

Position servo-drive can be performed by various algorithms. PID algorithm with lead compensator is applied, referring to the article [5]. The main asset is to execute synthesis of PID algorithm and the lead compensator. The entire position servo-drive structure is in the *Fig.2*.

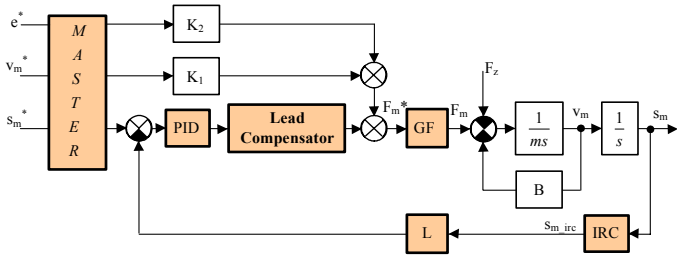


Fig.2 Entire diagram of position servo-drive (L-Luenbergerov observer, IRC-Incremental sensor, GF-Force generator)

2.1 Force Generator

Position servo-drive (*Fig.2*) include force generator LMPM. Force generator is one of the unavoidable blocks of linear servo-drive control structure, which works on a principal of vector frequency-current synchronous motor with PM control (*Fig.3*).

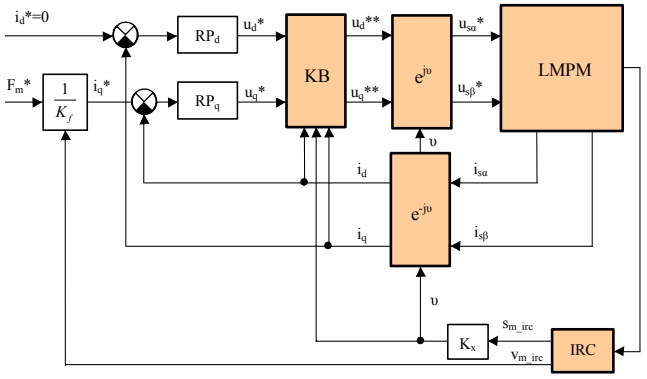


Fig.3 Force Generator LM PM Structure

2.2 PID position control design

PID controller is most used controller in praxis and it contains 3 parallel connected sub-circuits. The first sub-circuit is proportional, which multiply controller input value with adjustable coefficient. The second parallel sub-circuit integrates and the third parallel sub-circuit derivates controller input value.

We do not consider either dynamics of force generator nor lead compensator by synthesis of PID regulator.

It is used pole-placement method, which compares multinominal of close-loop system N(s) with desired multinominal $N_{\check{z}}(s)$ by equal squares [3], [4],

$$N(s) = N_{\check{z}}(s) \tag{1}$$

$$N(s) = s^3 + \left(\frac{K_p T_d + B}{m}\right)s^2 + \frac{K_P}{m}s + \frac{K_P}{T_i m} \tag{2}$$

$$N_{\check{z}}(s) = \left(s^2 + 2\xi\omega_0 s + \omega_0^2\right)\left(s + k\omega_0\right) \tag{3}$$

And final relations are:

$$K_p = m\omega_0^2\left(2\xi k + 1\right) \tag{4}$$

$$T_i = \frac{K_p}{mk\omega_0^3} \tag{5}$$

$$T_d = \frac{\omega_0\left(2\xi + k\right)m - B}{K_p} \tag{6}$$

Parameters setup ξ, k and ω_0 are in the *Tab. 1*.

2.3 Lead Compensator design

Lead compensator is a dynamic system with a common structure, which is often designed for unsuitable transfer poles or zeros of control system. In this case it is **compensating lead compensator** [6].

It will be used lead compensator with derivate character (a>1) and lead phase. Transfer function of the lead compensator

$$G_{LC}(s) = \frac{aT_1 s + 1}{T_1 s + 1} \tag{7}$$

, where parameter *a* is chosen from interval (10÷20), given on the ground of experiment and parameter T_1 can be counted as

$$T_1 = \frac{1}{\omega_r\sqrt{a}} \tag{8}$$

Lead compensator design issued from open-loop system of Bode characteristic. We were modifying parameter α by experimental method to achieve the smallest position error. Position error curve can be seen in the *Fig.4* for these parameters:

$$\omega_r = \omega_0 \qquad a = 20 \tag{9}$$

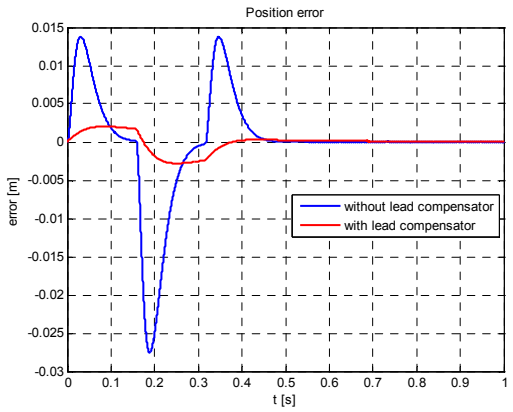


Fig.4 Position error comparison with PID + lead compensator and with just PID controller

In the *Fig.4* it can be seen positive influence of the lead compensator and consequently minimization of position error.

2.4 Master-slave generator

Master serves as a generator of control state value, whereby control vector can be greater than number of measured values. Generator of this control vector is realized on the principal of feedback algorithm, whereby for this realization is needed to know parameters of the control system „rough“model [1].

Slave contains controllers of state values.

Master generator task is to generate curves of desired state values.

2.4.1 Calculation of precorrection parameters

By calculating precorrection coefficients (K_1, K_2) we are starting from condition for feed forward control:

$$G_x(s) = \frac{1}{G_2(s)} \tag{10}$$

And dynamics of force generator is not considered.

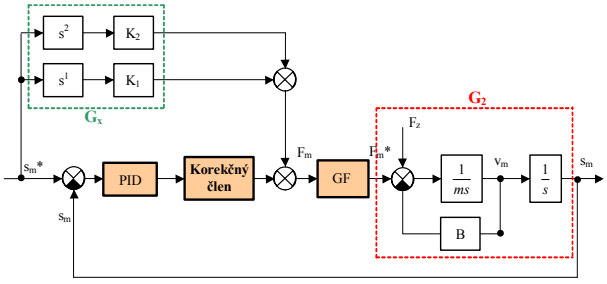


Fig.5 Position servo-drive block diagram with PID structure and marked precorrection Master-3D

From *Fig.5* results

$$G_2(s) = \left(\frac{\frac{1}{ms}}{1 + \frac{B}{ms}} \right) \frac{1}{s} = \frac{1}{ms^2 + Bs} \tag{11}$$

After substitution in the formula (*10*) refer

$$G_x(s) = \frac{1}{G_2(s)} = ms^2 + Bs \quad ; \quad \begin{matrix} K_1 = B \\ K_2 = m \end{matrix} \tag{12}$$

where parameters K_1 a K_2 answer coefficients of precorrection.

3. Precorrection influence on control system

The following simulations are issued from linear continuous design, however the realization is numerical with a sampling period 0.2 ms. Discrete diagram uses each part from chapter 2.

Quality verification is realized for discrete model with IRC sersor.

In the *Fig. 6* it can be seen striking influence of precorrection.

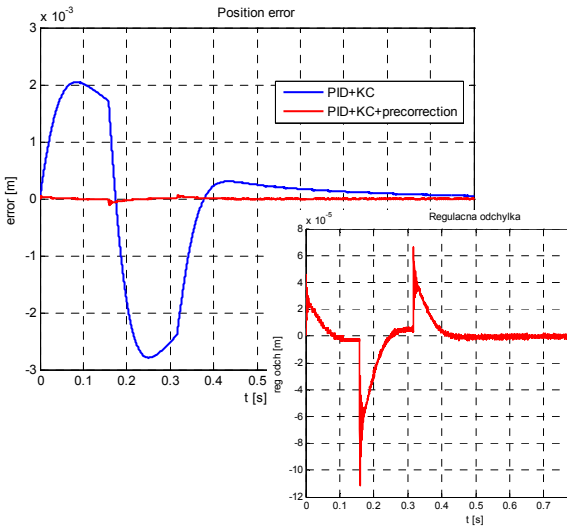


Fig.6 Position error comparison with/without precorrection (plus detail)

4. Luenberger observer design

Luenberger observer is an observer of angular velocity and acceleration. In general it exits different algorithm structures for observing angular velocity and acceleration. In this paper however we elect PID algorithm with the third system order.

Similarly as in the PID algorithm design, we are issuing from pole-placement method and comparing multi-nominal of close-loop system N(s) with desired multi-nominal $N_{\tilde{z}}(s)$ by equal squares.

$$N(s) = N_{\tilde{z}}(s) \tag{13}$$

$$N(s) = s^3 + \left(\frac{K_p T_d}{\tilde{m}} \right) s^2 + \frac{K_P}{\tilde{m}} s + \frac{K_P}{T_i \tilde{m}} \tag{14}$$

$$N_{\tilde{z}}(s) = \left(s^2 + 2\xi_1 \omega_{01} s + \omega_{01}^2 \right) \left(s + k_1 \omega_{01} \right) \tag{15}$$

Parameters setup ξ, k and ω_0 are in the *Tab 1*.

4.1 Luenberger observer influence on control system

Figure below (*Obr. 7*) shows influence of Luenberger observer on control performance.

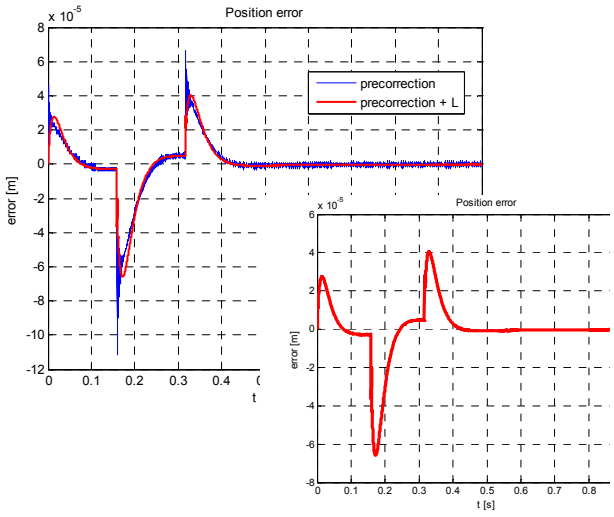


Fig.7 Position error comparison with/without Luenberger observer (plus detail)

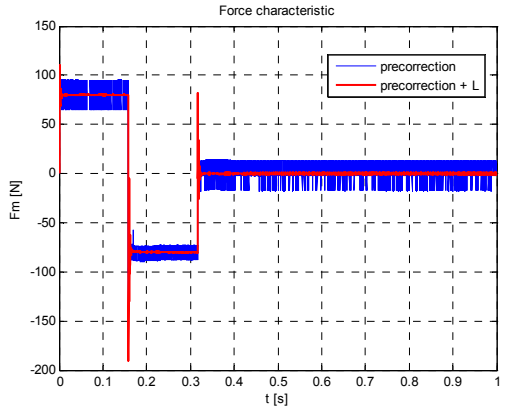


Fig.8 Force comparison with/without Luenberger observer

Luenbergerov observer markedly improves final value of force (*Fig. 8*), so in conclusion it has positive influence on control.

Acronym	Meaning	Value
T	Sampling period	0.2 ms
Parameters for PID controller		
ξ	Damping index	1
k	Shift pole index	1
ω_0	Bandwidth	$2\pi f_0$
f_0	Frequency	10 Hz
Parameters for Luenberger observer		
ξ_1	Damping index	1
k_1	Shift pole index	1
ω_{01}	Bandwidth	$2\pi f_0$
f_{01}	Frequency	10 Hz
Parameters for precorrection		
K_1	Precorrection constant	B = 0.01
K_2	Precorrection constant	m = 0.4 kg
Parameters for IRC sensor		
N	Resolution	2 μm

Tab.1 Table of acronyms and values

Conclusion

Simulation results listed above are gained from simulation diagram elaborated in detail in Matlab-Simulink environment considering discrete algorithms and IRC position sensor.

The aim of this paper was to evaluate results described in the article [5] and consequently design better solution for given problem. It succeeds to find engineering method which helped us to find sophisticated design of parameters for the lead compensator in dependence on PID controller dynamics. As follow new control structure was designed (*Obr. 5*) with add precorrection and Luenberger observer. In the next papers Luenberger observer will be under consideration moreover from point of view disturbance value compensation.

Acknowledgement

This paper was written with support of the project APVV-0530-07.

Literature

[1] ŽALMAN, M.: Akčné členy, STU Bratislava 2003

[2] www.vues.cz, Linear motors– EN_LIN-INFO___020910.PDF

[3] ŽALMAN, M.: Sevopohony lecture published in 2008/2009, available: http://servo.urpi.fei.stuba.sk/index.php?option=com_docman&Itemid=36

[4] ŽALMAN, M.: Inteligentné servopohony lecture published in 2008/2009, available: http://servo.urpi.fei.stuba.sk/index.php?option=com_docman&Itemid=35

[5] Zhou Yun-fei, Song Bao, Chen Xue-dong : Position/force control with a lead compesator for PMLSM drive system, Springer-Verlag London Limited 2005, 18 November 2005

[6] HARSÁNYI, L., MURGAŠ, J., ROSINOVÁ, D., KOZÁKOVÁ, A.: Teória automatického riadenia, STU Bratislava 1998

Ing. Tatiana Radičová, prof. Ing. Milan Žalman, PhD

Slovak Unievrsity of Technology
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3
812 19 Bratislava 1
tatiana.radicova@stuba.sk, milan.zalman@stuba.sk

Control System for Hybrid Electric Drive

Richard Balogh

Abstract
The paper deals with a design of the control system of the hybrid driven military vehicle with combined hybrid electric drive system. Considerations for the harsh environmental conditions and wide operational ranges are covered. Overview of available commercial-off-the-shelf (COTS) systems and experiences from the experimental operation are presented.

Keywords: vehicle computer, vetronics, industrial computer, hybrid drive.

Introduction

During the development of the hybrid driven vehicle [1], [2], [3], there was a requirement to control the cooperation of both slave controllers – the combustion engine ECU and the frequency converter of the electromotor. The purpose of the central control unit was to manage the optimal cooperation of both motors and their optimal operation regarding also the battery control unit.

1. Problem statement

Classic combustion engines offer many advantages to its users. They rose from more than 100 years of evolution and offers reliability, high power and high power density. On the contrary, besides the refuelling and exhalants they cannot offer very high dynamics. Their alternatives – electric motors – are much more efficient, offers high dynamic properties, and noiseless, low emission operation. Unfortunately, the electric drives offers only low energies and power densities. A great advantage is the possibility of reverse operation (regen braking).

As an alternative to the pure electric drive appeared a hybrid drive. Hybrid drive system consists of classic internal combustion engine operating in cooperation with the electric motor. It is supposed to combine the best from both systems. Unfortunately, they are balanced with higher costs, weight and last, but not least – higher complexity of the whole system.

There are two possible configurations for cooperation of combustion and electric machines – parallel and serial [4]. Although serial combination allows more efficient operation of the whole system, this was not our case. We were asked to design system with parallel combination, to increase the maximum available power, especially to improve the dynamic behaviour of the whole vehicle. Configuration of the system is on the Figure 1.

Considered system should offer at least five different operation modes:

- combustion engine drive for normal operation,
- pure electric (quiet) operation,
- combination of electric and combustion engines for increased torque and acceleration,
- idle run with battery charging,
- regen braking.

To make all different operation modes possible, the central

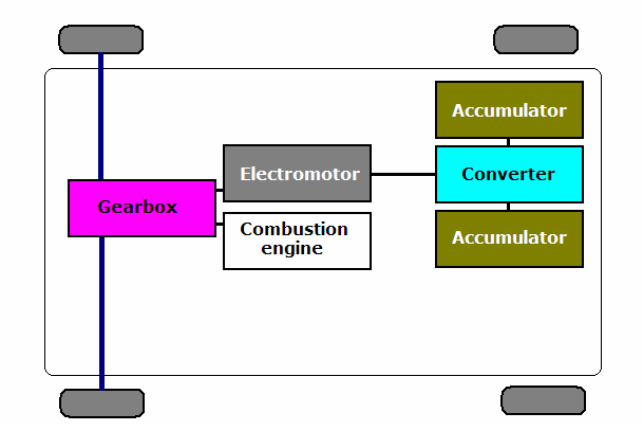


Fig.1 Parallel configuration of the hybrid drives system

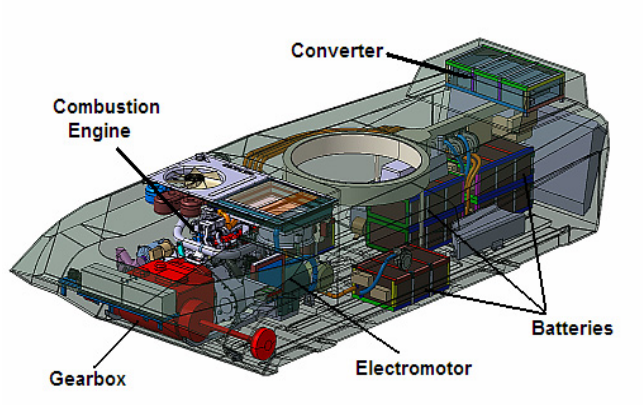


Fig.2 Component placement in a retrofitted vehicle [2]

controller is required to coordinate the operation of all subsystems. For testing the concept in real conditions, a discarded army vehicle OT-90 was used. The vehicle was retrofitted with a new combustion engine. Batteries and electromotor were added and connected to the new common gearbox. The system was designed at the STU in Bratislava and realized in company VOP Trencin.

2. Description of the components

1.1 Combustion engine

Original combustion engine was replaced by the new, air-cooled, emission limit Euro 3 satisfying, Cummins (USA) ISBe 250 30 motor. It has 6 valves and common rail system. This 5.9 litre motor weights of 470 kg and offers 184 kW and 950 Nm at 2500 rpm. As their momentum characteristic was not very good at low speeds, it was supposed that electromotor will add its power, especially during the accelerating phase of the drive. This motor has its own controller unit (ECU) and it is possible to control it remotely using the commands sent over the CAN bus.

1.2 Electric engine and converter

As an electric drive unit was used the Vacon (Finland) TRK 340WS electromotor. It is an asynchronous traction motor with the total power 150 kW and 483 Nm available. Its weight is 180 kg and current consumption is 248 A at 150 kW and 450 V AC power.

The motor was connected to the reversible Vacon NXI 04 605 frequency converter with IGBT power transistors. It can supply power up to 150 kW and 305 A (693 A for 2 sec.). It weights 100 kg and its operation is possible not only autonomously, but also using a remote control commands received from the CAN bus.

1.3 Battery subsystem

It was the most difficult problem to find an appropriate battery subsystem for the vehicle. The design succeeded with the set of six ZEBRA batteries Z37-620-ML3C-32 rated 620 V and 32 Ah at 207 kg each. These batteries operate at 250 °C and utilize molten sodium chloroaluminate ($NaAlCl_4$), which has a melting point of 157 °C, as the electrolyte. The negative electrode is molten sodium. The positive electrode is nickel in the discharged state and nickel chloride in the charged state. Because nickel and nickel chloride are nearly insoluble in neutral and basic melts, intimate contact is allowed, providing little resistance to charge transfer [5].

Fig.3 Zebra battery used in the system

Batteries were connected in parallel and operated using their internal CAN bus from the proprietary Zebra battery controller MBS (Multiple Battery System). Batteries were ready to accept also relatively high charging currents during the regen braking phase. We were able to measure all important parameters of the batteries during the tests using the Zebra monitoring software delivered with batteries.

Unfortunately, these batteries were not a mature product during the time of our project, so we need to replace some of them due to the technology problems of its producer and some of them were damaged during the tests.

1.4 Controller and HMI

It was clear from the beginning, that nobody is able to design one central controller for all complicated subsystems of the vehicle. As all the main components of the system (motors, batteries) already contained the controllers from their producer (incorporating also the producer's know-how), we were asked to design the system for control of the cooperation of the all units, system for communication with the driver and system to select different operation modes.

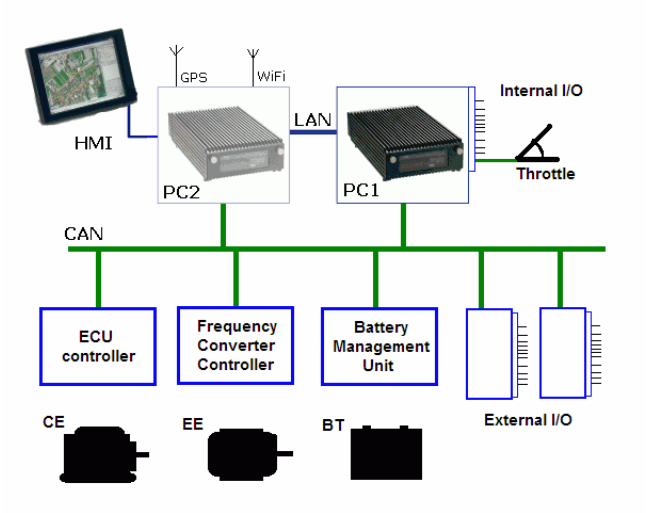


Fig.4 Structure of the control system

We decided to use two rugged versions of the industrial PC for coordination of all subsystems. As a communication system was proposed the existing CAN bus which seemed to be appropriate for such kind of applications. During the tests it makes clear, that even the all subsystems use the same bus, they are not compatible with each other. So in the final version the CAN bus for batteries and motors were spitted. To increase the reliability of the control system we duplicated the control computer. Two identical units (see Fig. 4) were interconnected using the high speed Ethernet connection. In the normal operation, one of the computers is dedicated to communicate with subsystems, whether the second one communicates with crew. The second computer is equipped with the LCD touch panel for entering user inputs and to show status of the system. One of the proposed user screens is shown in Fig. 5.

Design of the controller system was limited especially by the requirements on environmental conditions for its reliable operation. We require high modularity of the system, as we were just in development phase, so nobody was sure if the system requirements will not increase with time. As the system is located in the vehicle with 150 kW electromotor and frequency converter, we assumed a high level of electromagnetic noise. So the high level of EMC protection is a must.

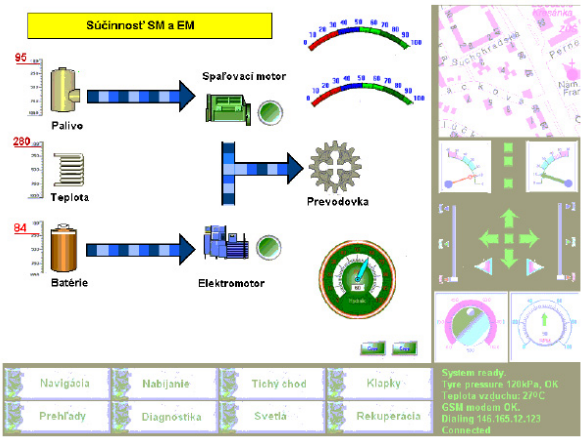


Fig.5 Proposed design of the user screen

Also the temperature range required was impressive: -40 to +60 ºC, dustproof and waterproof case. Also the mechanical robustness (vibrations, shocks) and stability were required. We didn't look for a specific embedded microcontroller system as we didn't know exact requirements for the system. Although a wide range of industrial computers were available, none of them were applicable for our system. Usually the temperature range was not sufficient to operate the vehicle even with low temperatures below the zero. Also the vibrations were more complicated as assumed at the beginning, we had to find a device without any rotating parts (fans, hard disks etc.). The computer we used is partially oversized for its purpose, but it was perfectly suited for additional tasks during the development phase. As we continue with testing and optimization, more and more data used to be acquired and evaluated. So we use the same control system not only for coordination of the subsystems, but also for diagnostics and configuration of subsystems, measurement and data acquisition and their visualisation on the place. It was supposed that the second PC will carry out the HMI interaction and the navigation support of the vehicle. Unfortunately, during the tests, the second PC was completely destroyed so we performed rest of the tests with the one PC unit only. It was now clear, that the backup idea was right.

As the base for the control system we used the MicroSpace MPCX-47 vehicle computer from the Swiss company Digital Logic. It contains Intel Pentium M 738 processor, 1 GB memory and 4GB solid state flash disk. Computer was configured as the system with passive cooling through the case, and with four independent CAN channels. Also the analogue inputs, outputs and some 24 V digital I/O were onboard [6].

Fig.6 Vehicle computer MPCX-47 by the Digital Logic

We used all of them. Two CAN bus subsystems were used for communication with motors and batteries. Third CAN subsystem was used for the diagnostic computer. Analogue inputs were used for measurements of position, digital ones for some safety switches. Digital outputs were used to control the gear switch, pneumatic valves and other auxiliary devices required.

One advantage of this computer is that it is perfectly suited for vehicle applications. It is powered from 24 VDC source, it contain the built-in auto diagnostics system (temperature, power supply) and it offers 8 digital I/O and 8 Analogue Inputs plus 3 counters. The whole system is placed in massive aluminium chassis which creates also a passive cooling system. The computer weights 3.2 kg and is IP65 protected together with some vibrations certificates.

3. Implementation

We build the system incrementally. At first, each component of the system was set upped independently. Then, subsystems were interconnected using the CAN bus and tested. After the initial phase, every device was captured during the time, and its behaviour evaluated. Then the controller program was developed and we started to send commands to the CAN bus. As we mentioned before, the protocols were somehow incompatible, so we split the CAN bus for batteries from the motor controllers. After that all the motors and batteries were moved to the test bed and further optimization of the controller algorithm was performed, based on the data measured under different load conditions. After the successful verification of this system, we use the discarded OT-90 vehicle for field testing. All the systems were mounted inside the vehicle. We needed to save also some space for crew, so the inverter unit was placed on the roof of the vehicle. The biggest space consumers were clearly batteries. Also some safety precautions were necessary due to the high temperature of batteries during their operation. Another issue was high level voltages in the vehicle. Usually the automobile electronic is considered to be intrinsically safe, as the maximum voltage is 12/24 Volts only. That was not the case as we have live 620 Volts inside! So there was necessary to add some safety instrumentation, especially the device for permanent insulation resistance measurement.

First test shows the concept was well designed and vehicle was operating according the assumptions. At the present time, further field tests are performed. On the Figure 7 thera are plotted example data measured and captured by the central control unit via the vehicle CAN bus. Plotted are the battery currents at different speeds of the electromotor.

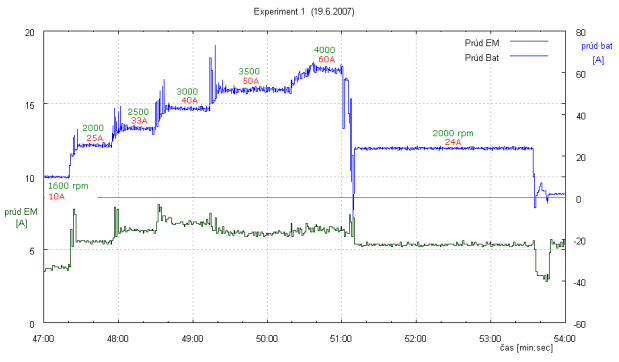


Fig.7 An example experiment on hybrid drive

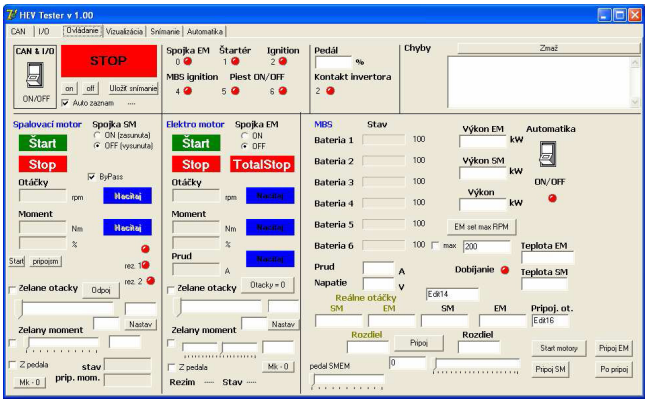


Fig.8 An example of the user screen in final version [3]

Conclusion

We proved the concept of the parallel hybrid electric drive, and now we know all its advantages and drawbacks. During the development we learned many lessons and facts that were not mentioned in datasheets and manuals. Our experience shows that most problematic part of such system are batteries. Their technology should be improved to obtain more reliable and predictable results and operation of the vehicle without need for trained personal.

Acknowledgements

This work was solved for Ministry of Defence of Slovak Republic, in cooperation of DSSI, a.s. and STU in Bratislava. It was also supported by the VEGA project 1/2104/05 *and its publication was supported by the VEGA* 1/0690/09 *project.*

References

[1] KOZÁK, Š. et al.: *The Final Technical Report of the Research Project Control System for the Hybrid Drive Vehicle.* Ev. 2004 UO MO SR / 06. IOSŠ, Bratislava, 2007. [In Slovak]

[2] MATEJ, J., DANKO, J. and FERENCEY, V.: *Toky energií a výkonov v hybridnej pohonnej sústave.* Zborník 8. Medzinárodnej konferencie TRANSFER, Trenčín, 2006. [In Slovak].

[3] CHVOSTEK, T. – KOZÁK, Š.: *Design of communication subsystem for the hybrid vehicles.* In: Kybernetika a informatika : Medzinárodná konferencia SSKI. Ždiar, Slovenská republika, 10.-14. február 2008. - Bratislava : STU, 2008. - ISBN 978-80-227-2828-7.

[4] MATEJ, J. - DANKO, J. - ŠOTNÍK, J.- FERENCEY, V.: *Analýza energetických tokov v hybridných pohonoch motorových vozidiel.* In: KOKA 2006. 37. mezinárodní konference kateder a pracovišť spalovacích motorů českých a slovenských vysokých škol - Praha : Česká zemědělská univerzita v Praze, 2006. - ISBN 80-213-1510-5. [In Slovak]

[5] Wikipedia contributors: *Molten salt battery.* Wikipedia, The Free Encyclopedia. Available online: <http://en.wikipedia.org/w/index.php?title=Molten_salt_battery>

[6] MPCX-47 Technical Users Manual. Digital-Logic AG, Luterbach, 2005. Available online: <http://www.digitallogic.com/index.php?id=manuals&filename=MPCX47Manual.pdf & dir=MPCX47&task=download&mountpoint=45>

Ing. Richard Balogh

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering and
Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3
SK-812 19 Bratislava, Slovakia
E-mail: richard.balogh@stuba.sk

Hybrid control of the mono axial vehicle

Peter Hubinský, Jozef Rodina

Abstract

The Mono Axial Vehicle (MAV) is an interesting platform for small service mobile robots, especially for its robustness. The chassis of the MAV can rotate around the wheels axis, hence we have to use different control techniques. This paper presents the concept of using of different control techniques in the MAV. This hybrid control method was tested in MATLAB SIMULINK environment. The implementation of these control techniques in real MAV is being prepared.

Key words: Inverted pendulum, oscillation damping, input signal shaper, MEMS

Introduction

The most common design for the mobile robots or for the autonomous vehicles is the differential drive. It consists of two independently powered wheels. This kind of chassis uses also stabilizing wheels which ensure stability of the chassis. But in our case we don't have any stabilizing wheel and also the chassis is smaller than the diameter of the wheel. Therefore the chassis is able to rotate around the wheel axis.

This kind of chassis (Fig.1) brings new problems in the motion control of the mobile robots, because it is an oscillating system in lower position (centre of gravity is under wheel ax) and an unstable system in upper position (centre of gravity is above wheel ax).

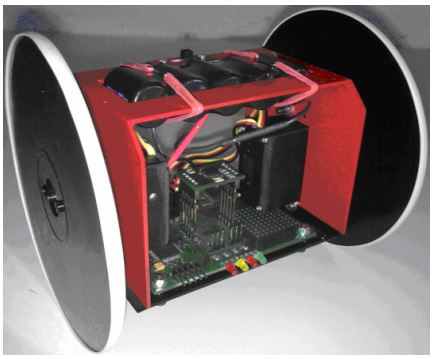


Fig.1 Mono axial vehicle (MAV)

Because of this different behavior different control methods have to be used in upper and in lower position. As mentioned in lower position the system has tendency to oscillate during the acceleration of the vehicle or in case when external forces are affecting the vehicle. Therefore we have to dump this oscillation.

There are various techniques for damping oscillation. We can increase damping ratio of the oscillating system by derivative feedback control. Or we can use feedforward control. We are able to filter input signal so input signal changes do not cause the system oscillation. For this feedforward control we are using signal shapers. We are planning to use LQR controller for stabilizing the MAV in upper position. These control methods were tested in MATLAB SIMULINK environment and simulated results are shown in this article.

1. Dynamic model of the MAV

In order to develop the control system we need to create a mathematical model of MAV. This system behavior is similar to pendulum (inverted) on wheels. First we analyzed the body and the wheels separately and in the end we put it together and get equations of motion of the mono axial vehicle.

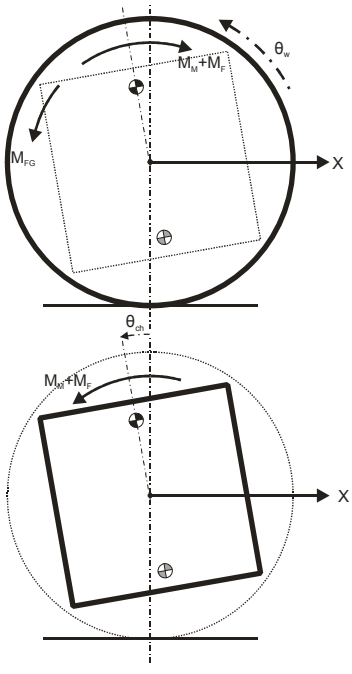


Fig.2 Free body diagram of the mono axial vehicle

According to Fig.2 the following equations for the mono axial vehicle can be defined:

$$J_{CH}\ddot{\varphi} = m_{ch} g L \sin\varphi - (M_J + M_A) \quad (1)$$

$$M_J = b(\dot{\varphi} - \frac{\dot{x}}{R}) \quad (2)$$

$$M_G = b_{wg}\dot{x} \quad (3)$$

$$\ddot{x} = \frac{M_A}{J_W} R \quad (4)$$

$$M_A = M_L + M_R \quad (5)$$

where:

M_A sum of applied torques on the wheels

$M_{L/R}$ applied torque on the left / right wheel

M_G torque of viscous friction of the wheel

M_J torque of viscous friction in the joint

$\ddot{\varphi}, \dot{\varphi}, \varphi$ angular acceleration, velocity, angle of the chassis

g gravitational acceleration

L position of the center of gravity of the chassis

$\ddot{x}, \dot{x}, x$ acceleration, velocity and position of the vehicle

J_{CH} moment of inertia of the chassis

J_W moment of inertia of the wheel

R wheel diameter

m_{ch} chassis mass

By these equations we defined all forces affecting the system of the MAV. Now we have to express a mass of a reduced body yet. We can do this by expressing kinetic energy of the system as follow:

$$E_K = \frac{1}{2} m_{MAV} \dot{x}^2 + \frac{1}{2} 2 J_W \left(\frac{\dot{x}}{R}\right)^2 \quad (6)$$

We get mass of the reduced body:

$$m_{red} = m_{MAV} + 2 J_W \frac{1}{R^2} \quad (7)$$

Now we are easily able to get together the nonlinear simulation scheme of the MAV in MATLAB SIMULINK. This scheme can be seen in Fig.3. There is function of the chassis (pendulum) in the green block, this function is described by equation (1). There are feedbacks between chassis and wheels and between wheels and ground in the blue blocks. These blocks are two because we are using differential drive and thus we have to calculate with two wheels.

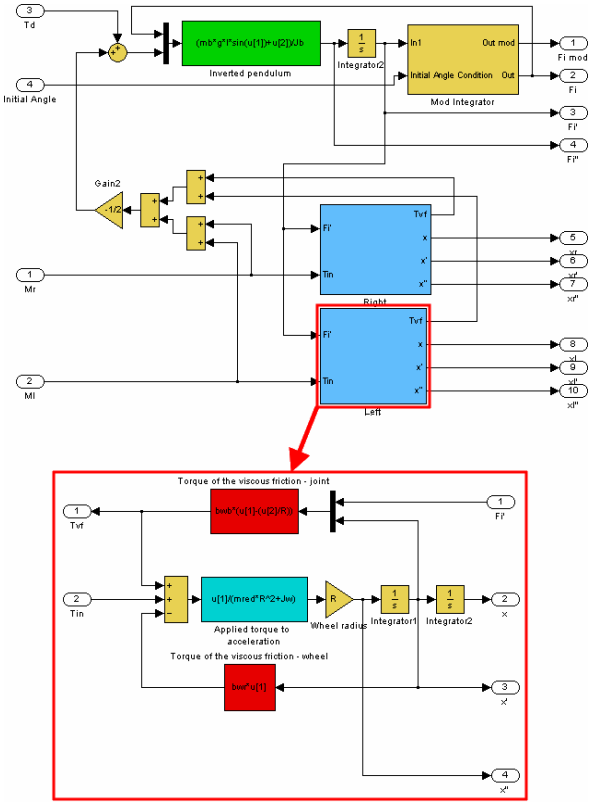


Fig.3 Simulation scheme of the mono axial vehicle

2.1 Dynamic model simulations

We simulate the dynamic model by applying the step of the control signal on the inputs of model. This excites oscillations of the chassis. Also disturbances affected on the MAV excite oscillation. This can be seen in Fig.4 and Fig.5.

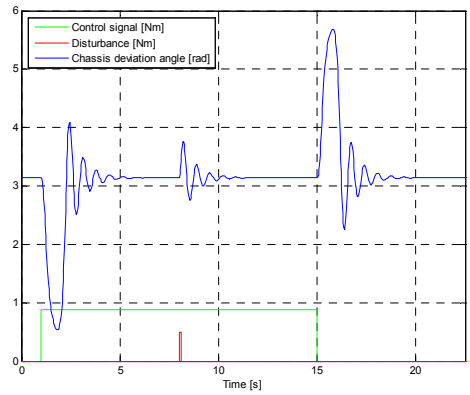


Fig.4 Oscillation of the chassis of the MAV

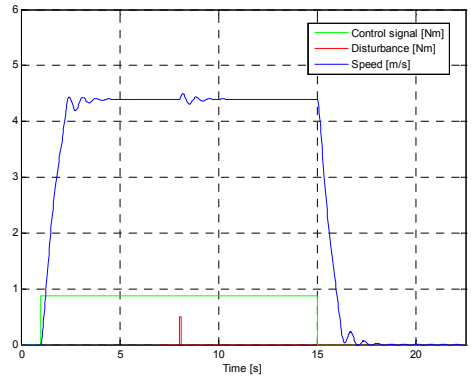


Fig.5 Oscillation of the speed of the MAV

2. Oscillation damping

2.1 Control signal shapers

Changes of control signal (wheels speed is changing) are exciting oscillation (around lower position) in the MAV. This oscillation or this state of the MAV is called residual oscillation. To avoid this oscillation input signal shapers are usually used. As can be seen in Fig.6 the shaped signal is obtained by convolving desired input with the series of Dirac impulses.

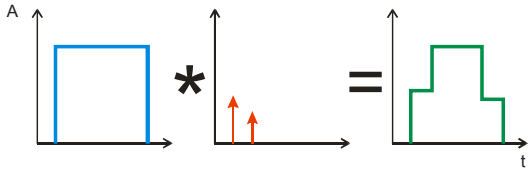


Fig.6 Basic principles of signal shaping methods

Basic type of shapers is Zero Vibration Shaper (ZV shaper). It uses only 2 impulses. The ZV shaper is sensitive to the changes of the system own resonant frequency, therefore in our case we are using Zero Vibration Derivative Shaper (ZVD shaper). The ZVD shaper is less sensitive to changes of the system resonant frequency and its uses 3 impulses. ZVD shaper for system with a natural damping *b* is described by following equation:

$$\begin{bmatrix} A_i \\ t_i \end{bmatrix} = \begin{bmatrix} \frac{1}{(1+k)^2} & \frac{2k}{(1+k)^2} & \frac{k^2}{(1+k)^2} \\ 0 & \frac{T_D}{2} & T_D \end{bmatrix}, k = e^{\frac{b.\pi}{\sqrt{1-b^2}}} \qquad (8)$$

Simulation scheme of the ZVD shaper is shown in Fig.7.

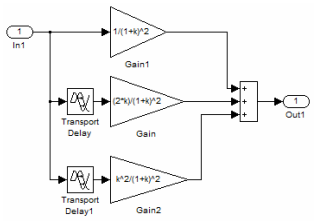


Fig.7 Simulation scheme of the ZVD Shaper

2.1.1 Evaluation of the ZVD shaper

For evaluation we created a simulation model of damping control using ZVD shaper (Fig.8).

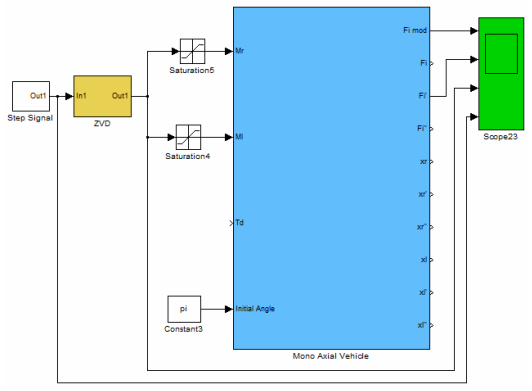


Fig.8 Simulation model of damping control using ZVD shaper

First we need to identify parameters of oscillation of the MAV. Resonant frequency and period of the MAV was calculated as follows:

$$\omega_0 = \sqrt{\frac{m_{ch}.g.L}{J_{ch}}}; T_0 = \frac{2\pi}{\omega_0} \qquad (9)$$

The damping ratio *b* was identified experimentally from the oscillation which was shown in the Fig.5. ZVD shaper was set by using these parameters.

The ZVD shaper was tested with the same input signal as was in the simulations above (Fig.4, Fig.5). Result of these simulations shows next two graphs (Fig.9, Fig.10).

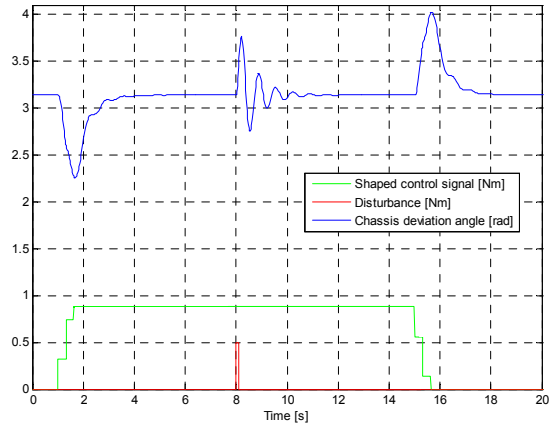


Fig.9 Graph of the chassis angle of the MAV when is used the ZVD shaper

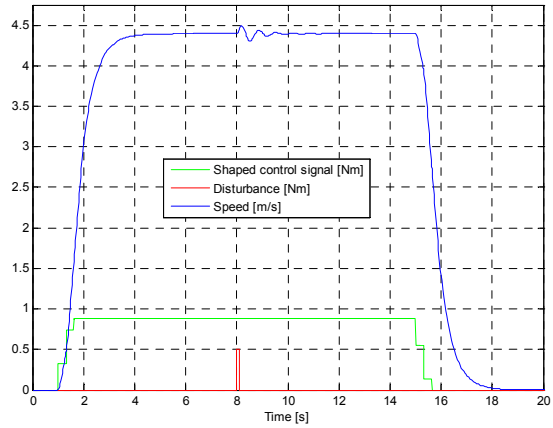


Fig.10 Graph of the velocity of the MAV when is used the ZVD shaper

As can be seen by using of ZVD shaper we are able to eliminate oscillation, which is occurred as response on steps of the input signal. But we are not able to eliminate oscillation caused by disturbances.

2.2 Derivative feedback

Another method to increase the natural damping ratio of systems is the derivative feedback. By using the derivative feedback we can move oscillating poles of the system near to the real axis in the complex plane, so the damping ratio of the system will increase. Scheme of typical derivative feedback is in Fig.11.

We want to damp oscillation of the chassis angle. We will use angular velocity of the chassis for feedback. So derivation of angle output of model is not needed in this case, because it is available from the Mono Axial Vehicle simula-

tion block. Scheme of simulation model can be seen in Fig.12.

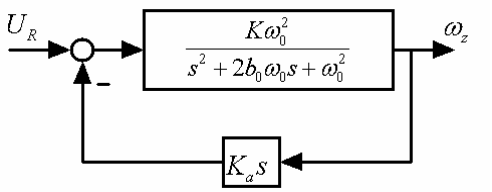


Fig.11 Structure of the control system with derivative feedback

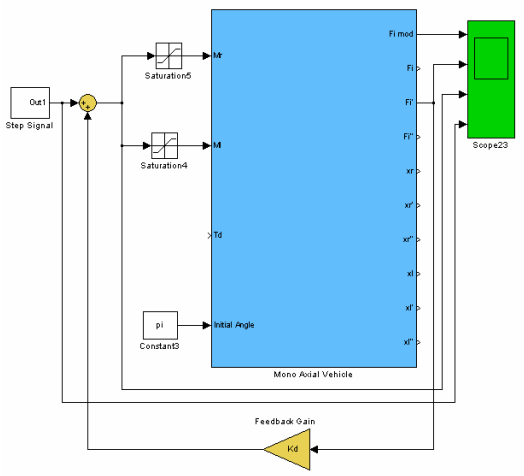


Fig.12 Simulation scheme of the MAV with derivative feedback

Derivative feedback gain was set experimentally. If the system contains another pole pair, we can destabilize it by increasing feedback gain. This might decrease effectiveness of this method.

2.2.1 Derivative feedback evaluation

As mentioned before we set derivative feedback experimentally. The value which gain feedback was finally set is equal to 0.23. As in case of evaluation of the ZVD shaper we evaluated this method by using simulation scheme shown in Fig.12. Results of these simulations are visible in Fig.13 and Fig.14

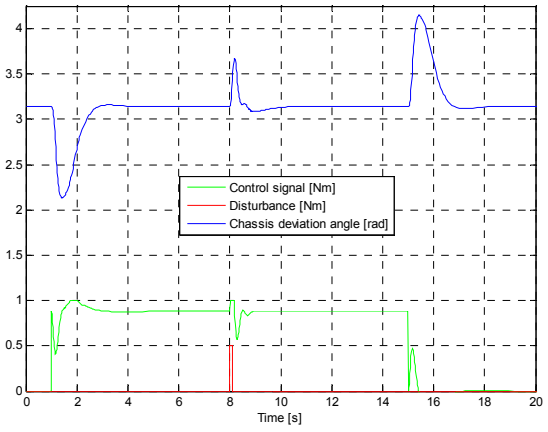


Fig.13 Graph of the chassis angle of the MAV when is used the derivative feedback

Derivative feedback compared to the ZVD shaper is able to avoid oscillation which is caused by disturbances. Also by using this method the damping ration of the system was increased.

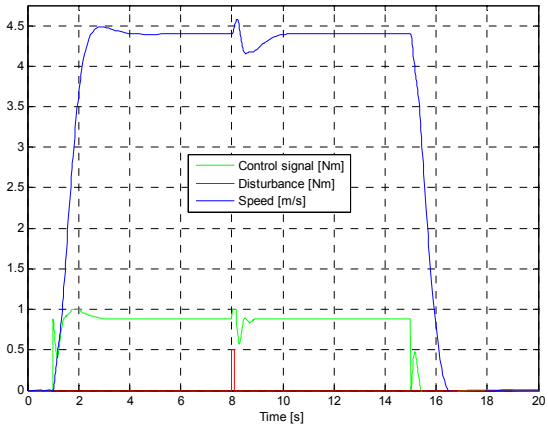


Fig.14 Graph of the velocity of the MAV when is used the derivative feedback

3. The LQR controller for stabilizing the MAV

3.1 LQR controller

Linear Quadratic Regulator Controller is optimal state-feedback controller with good robustness for robotics applications (Fig.15). In fact the LQR controller is optimal pole placement controller.

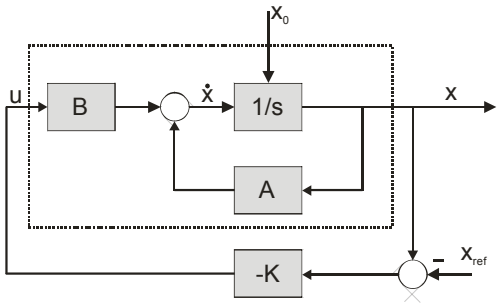


Fig.15 LQR controller

In order to design the LQR controller we need to get the linear state space model of the system as follow:

$$\dot{x} = Ax + Bu$$
$$y = Cx \qquad (10)$$

And the control law of LQR is defined by this equation:

$$u = -Kx \qquad (11)$$

The control gain K (12) is obtained from the infinite horizon performance index J (13) and the solutions of the Riccati equation for infinite horizon is (14).

$$K = R^{-1}B^T P \qquad (12)$$

$$J = \frac{1}{2}\int_{t_0}^{\infty}(x^T Qx + u^T Ru)dt \qquad (13)$$

$$0 = A^T P + PA - PBR^{-1}B^T P + Q \qquad (14)$$

The matrices Q and R should be selected. These matrices are usually diagonal and first choice for these matrices can by given by using the Bryson's rule (15).

$$r_{ii} = \frac{1}{[\max u_i]^2}$$
$$q_{ii} = \frac{1}{[\max x_i]^2} \qquad (15)$$

The LQR is a linear controller, therefore we have to linearise mathematical model of the MAV. Linearization of the dynamic model of the MAV can be done near an equilibrium of the MAV in upper position.

In time of writing this paper we were still working on the LQR controller, therefore we didn't have any simulations available yet.

Conclusion

This paper presents various control techniques to damp oscillation and optimal control of the MAV which can be used as platform for service mobile robots.

First we analyzed oscillation and designed ZVD shaper for damping oscillation. In next step we present derivation feedback, which is used for oscillation dumping. ZVD shaper does not affect stability of the system, but it is also not able to damp oscillation caused by disturbances. The derivation feedback in comparison with the ZVD shaper is able to eliminate oscillation caused by disturbances, but it might destabilize the system. Next advantage of the ZVD shaper can be that it does not require any additional sensors. Also the ZVD shaper can be used if sensors fail. We can also combine these two methods for oscillation damping. For example we will use ZVD shaper during the phase of control signal change, but after this phase we will use derivation feedback.

We also showed basic principle of the LQR controller. Since we are still working on it during the time of writing this paper, the LQR control method is not described and evaluated here.

These control techniques describe just oscillation damping and stabilizing of the MAV. Next step will be to implement this hybrid control into real MAV and to implement methods for navigation in environment to the MAV [4]. Now we are also working on chassis of the MAV and on electronics for it. Chassis will be based on composites materials. The MAV will be driven by MAXON motors and planetary gear. Electronics is based on DSP microcontroller and MEMS sensors.

Acknowledgement

This work was supported by Grant Agency of Ministry of Education and Academy of Science of Slovak Republic VEGA under Grant No. 1/0690/09. The authors are pleased to the acknowledge this support.

References

[1] SINGHOSE W., SINGER N., SEERING W.: Comparison of Command Shaping Methods for Reducing Residual Vibration, Published in 1995. [online] http://www.me.gatech.edu/inputshaping/Papers/ECC95.

[2] KOZÁKOVÁ A.: Design of discrete-time compensator for reference tracking with disturbance rejection. /Transactions of the VŠB - Technical University of Ostrava, Mechanical Series/, 2008, vol. LIV, No.2 (2008), 67-72. ISBN 978-80-248-1727-9

[3] GRASSER F., D'ARRIGO A., COLOMBI S., RUFER A.: JOE: A Mobile, Inverted Pendulum, Published in 2001 [online] http://leiwww.epfl.ch/publications/grasser_darrigo_colombi_rufer_mic_01.pdf

[4] BABINEC A., DUCHOŇ F.: Metódy reaktívnej navigácie mobilného robota v neznámom prostredí, Published in: Nové trendy v kybernetike, automatizácii a informatike : Odborný seminár. Gabčíkovo, Slovenská republika, 7.-9.9.2009. - Bratislava : STU v Bratislave FEI, 2009. - ISBN 978-80-227-3107-2. - CD-Rom

Ing. Jozef Rodina

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of control and industrial informatics
Ilkovičova 3, 812 19 Bratislava
+421/2/60291608
jozef.rodina@stuba.sk

Doc. Ing. Peter Hubinský, Phd.

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering
and Information Technology
Institute of control and industrial informatics
Ilkovičova 3, 812 19 Bratislava
+421/2/60291608
peter.hubinsky@stuba.sk

Estimation of the gantry crane natural frequency using MEMS accelerometer

Peter Hubinský, Jozef Kurilla, Lukáš Palkovič

Abstract
When controlling a gantry crane, residual vibration of the payload occurs. This vibration is ineligible. To eliminate this vibration, feedforward control realized by input signal shapers is often used. This method works with natural frequency of the controlled system that can vary with geometrical parameters of the crane. This paper proposes estimation of the natural frequency from online data obtained from a MEMS accelerometer fixed on the payload.

Keywords: gantry crane, input signal shaper, natural frequency identification, MEMS accelerometer

Introduction

Mechanical dynamic systems containing structures with low friction (low damping ratio) are characterized by residual vibration. It is a vibration still present in the system after the period of varying input signal. Since it influences the quality of the positioning and takes too much time, it is ineligible.

Illustrative cases of such systems are robotic arms with harmonic drives, servo-drives with belt gearings, filling and sealing lines where liquids are transported in opened containers and cranes.

There are various techniques used to reduce this vibration. One of them is to increase the friction in order to damp the vibration quickly. These mechanical modifications can be expensive and other parameters (such as the weight of the device) can get worse. The other way is a derivative feedback control. This method requires an additional sensor of the output variable. It can be sometimes difficult to implement. In more complicated systems, more variables have to be measured.

A feedforward control is a different approach. It is possible to control the system with a signal that does not cause the residual vibration. This signal can be prepared in advance and can be used repeatedly (off-line method) or every signal can be modified in order not to cause the vibration (on-line method).

A drawback of this method is, that externally caused vibration (manual interaction, wind) can not be reduced using this method. The shaper can be implemented by software, hence the cost is reduced. However, both methods require knowing the parameters of the system.

The basic parameters are damping of the system and its natural frequency. The damping does not affect the shaping algorithms significantly and can be often even negligible. The natural frequency is very important parameter and its knowledge has a crucial impact on the vibration reduction. Especially in the cranes, the natural frequency changes significantly and depends on the length of the cable.

When the shaper works with well estimated frequency, there is no residual vibration, or the vibration is very low. However, if the cable length changes, the natural frequency changes too, and the input signal shaping does not reduce the residual vibration enough. The cable length can be measured, bud it does not include the changes of the position of the centre of gravity of the payload. A way, how to use the residual vibration to estimate the natural frequency of the system is presented.

1. Input signal shapers

The payload suspended under a gantry can be mathematically described as a pendulum with moving pivot point. The crane has two horizontal degrees of freedom, while the third degree of freedom is represented by vertical movements of the payload that affects the system parameters. Suppose independent transitions in each horizontal direction. Then, each transition can be considered separately.

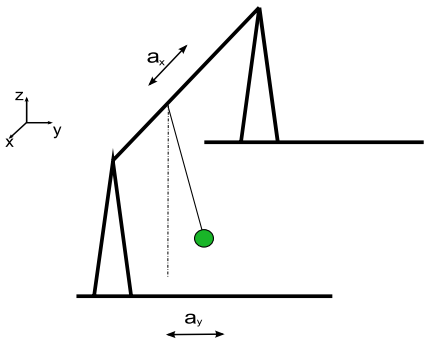


Fig.1 Simplified schematic of gantry crane

After linearization, the pendulum can be described by a transfer function

$$F(s) = \frac{\theta(s)}{u(s)} = \frac{K\omega_0 s}{s^2 + 2b\omega_0 s + \omega_0{}^2}, \qquad (1)$$

where $\theta(s)$ is an angular deflection of the pendulum, u is

the portal velocity, *K* is the gain, ω_0 is the natural frequency of the plant and *b* is its attenuation.

The problem of such crane is following. The pendulum usually does not stop after the subsidence of the input signal. This means, it stays moving around the stabilized position. This state's name is called residual vibration.

There is a way, how to reduce this vibration. It is based on the preconditions that the magnitude of the input signal spectrum on the natural frequency of the pendulum has to be equal to zero.

It is usually made by using of an input signal shaper that processes the input signal as showed in Fig.2. The input signal is multiplied by convolutory product with a signal satisfying the mentioned precondition. The output signal satisfies this precondition too.

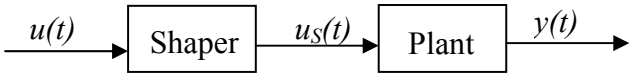


Fig.2 Block diagram of the system with shaper

Typical signal used for this purpose is a series of Dirac impulses

$$u_D(t) = \sum_{i=1}^{N} A_i \delta(t - t_i), \qquad (2)$$

satisfying following conditions:

$$\left| \sum_{i=1}^{N} A_i e^{-j\omega T_{Di}} \right|_{\omega=\omega_0} = 0 \qquad \sum_{i=1}^{N} A_i = 1 \qquad (3)$$

The principle is shown in Fig.3.

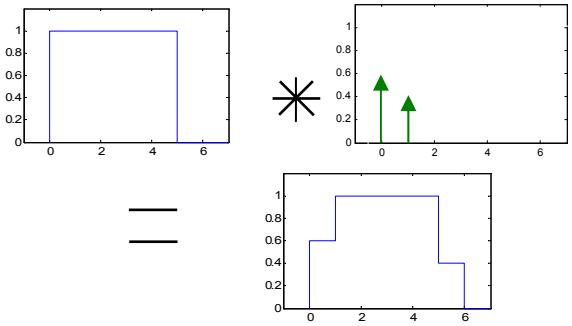


Fig.3 Principle of the signal shaper

The simplest shaper is so-called Zero Vibration Shaper (ZV shaper). It uses only 2 impulses. The delay between them is half of the period derived from the natural frequency of the plant. In there is no attenuation (b = 0), the following constants are used.

$$t_1 = 0 \quad A_1 = 0,5 \qquad t_2 = \frac{\pi}{\omega_0} \quad A_2 = 0,5 \qquad (4)$$

Fig.4 shows the original signal, modified signal and the positions of the pendulum as responses to these signals.

ZV shaper is sensitive to the changes of the natural frequency. There are also less sensitive shapers, for example ZVD shaper characterized by the following constants (again, assume b = 0):

$$t_1 = 0 \qquad t_2 = \frac{\pi}{\omega_0} \qquad t_3 = \frac{2\pi}{\omega_0}$$

$$A_1 = 0.25 \qquad A_2 = 0.5 \qquad A_3 = 0.25\,. \qquad (5)$$

Fig.5 compares the responses of the system with the ZV and ZVD shaper, when the natural of the shaper is 15% higher than the frequency of the shaper.

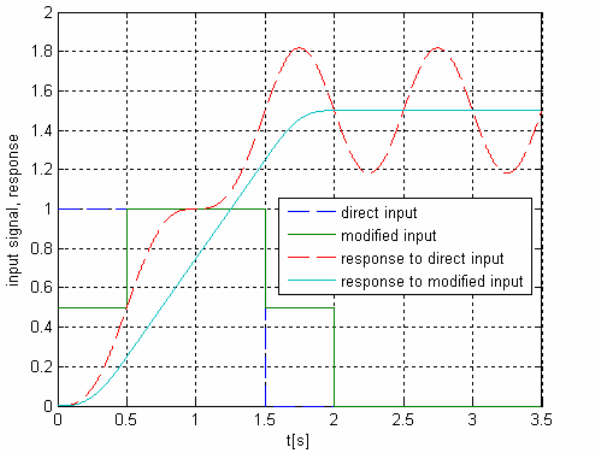


Fig.4 Comparison of the responses to the direct signal and signal from ZV shaper set to the correct natural frequency.

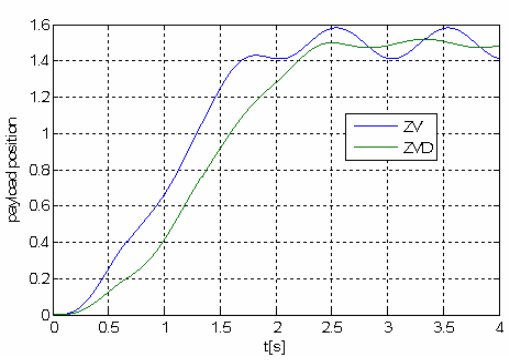


Fig.5 Response of the system with ZV and ZVD shaper, when the natural frequency is 15% higher than the estimated frequency used to set the shapers

So-called sensitivity curves of ZV and ZVD shapers are shown in Fig.6. They tell us what residual vibration will be for the given error in the estimation of the frequency.

It is obvious, that there is more residual vibration with ZV shaper; therefore ZV shaper is more sensitive to the modeling errors than the ZVD shaper. ZVD shaper is more robust.

In the other hand, it can be shown, that more robust shapers have typically higher delay, than the less robust shaper.

However, the natural frequency is basic parameter used to set the shaper delays correctly.

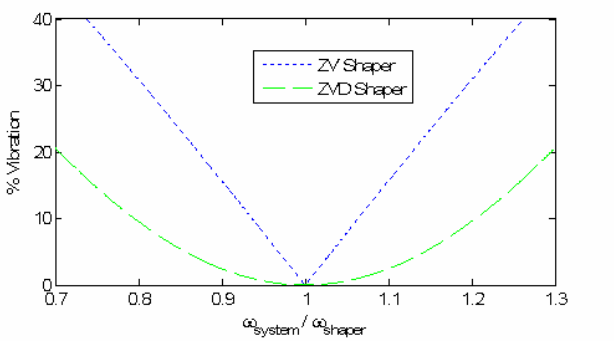


Fig.6 Sensitivity curves of ZV and ZVD shaper

2. Dependence of the natural frequency on the rod length

Generally, physical pendulum is an object allowed to move around its horizontal axis that does not contain the centre of gravity of the object. If the pendulum is unbalanced, it oscillates around its equilibrium with its natural frequency

$$\omega_0 = \sqrt{\frac{mgL}{J}}, \qquad (6)$$

where J is moment of inertia of the object, L is the distance between the pivot point and the centre of the gravity and g is the gravitation constant. Damping ratio is assumed zero, since its values are very low and do not affect the natural frequency considerably.

If the mass is concentrated in one point, the system can be characterized as mathematical pendulum and its natural frequency is

$$\omega_0 = \sqrt{\frac{g}{L}}. \qquad (7)$$

Gantry crane system is often simplified and assumed as a mathematical pendulum. Fig.7 illustrates dependence between the relative natural frequency and relative rod length.

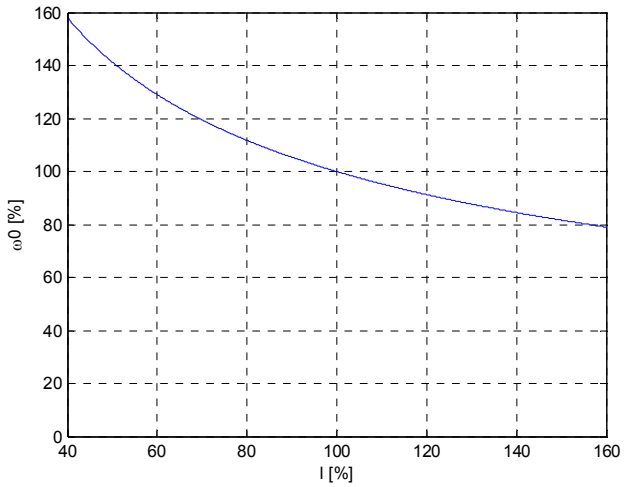


Fig.7 Dependence between relative frequency and the relative rod length

In order to correctly set the shaper, if the length varies considerably, the rod length should be measured. Alternatively, the natural frequency can be measured.

3. Adaptive setting of the shaper

If the frequency varies, insensitive shapers can be used. Their disadvantage is the delay they produce. Other approach is to estimate the natural frequency of the system periodically. Then, the time delays can be adjusted according to the estimated frequency. Such structure is shown in Fig.8.

The frequency estimator analyzes the output signal of the system. It is difficult to measure the position of the payload directly. Therefore, the deflection can be measured. In some cases with a precise sensor, feedback control should be considered instead. If the deflection can not be measured directly, MEMS accelerometer can be used.

The algorithm can work in cycles. In the beginning it stores the output signal samples to the buffer. Then, the signal is analyzed, while another storing cycle starts. Once the frequency is estimated, it can be used to set the shaper, while the new cycle is not affected.

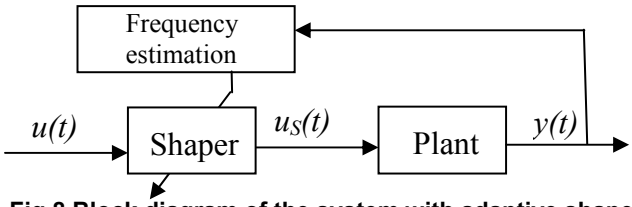


Fig.8 Block diagram of the system with adaptive shaper

4. Accelerations measured on the payload

The idea is to estimate the natural frequency of the system from the accelerations measured on the payload.

Fig. 9 illustrates the principle of such algorithm. Measured acceleration depends on the acceleration of the pendulum and on the gravity projection to the measuring axes of the accelerometer.

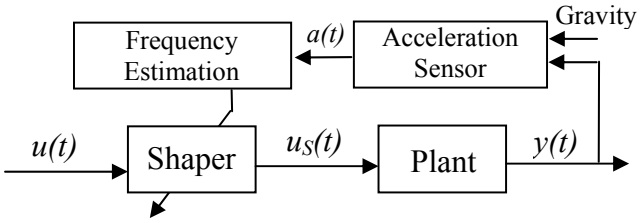


Fig.9 Block diagram of the system with adaptive shaper using accelerometer loop.

Accelerations measured with an accelerometer on the pendulum should be analyzed. Fig.10 illustrates main components of the accelerations that affect the signal.

First, assume a two-axis accelerometer with the axes oriented in the tangential direction and in the radial direction of the pendulum.

The signal in radial direction is

$$a_r(t) = a_{gr}(t) + a_c(t) + a_{ext\_r}(t) + z_r(t) \qquad (8)$$

and in tangential direction is

$$a_t(t) = a_{gt}(t) + a_{pt}(t) + a_{ext\_t}(t) + z_t(t) \qquad (9)$$

Where $a_{gr}(t)$ and $a_{gt}(t)$ represent the projection of gravity to the corresponding axes, $a_{ext\_r}(t)$ and $a_{ext\_t}(t)$ represent external forces (gantry movements, damping, wind etc.), $z_r(t)$ and $z_t(t)$ represent noise, bias and distortions in the signal. $a_c(t)$ is centrifugal acceleration and $a_{pt}(t)$ is the tangential acceleration of the pendulum.

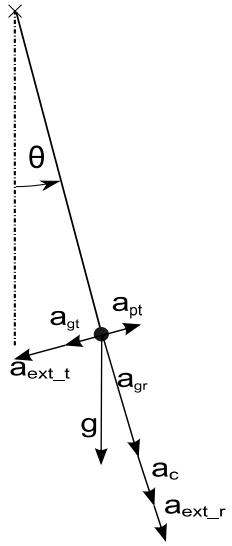


Fig.10 Block diagram of the system with adaptive shaper using accelerometer loop

Movement of free pendulum can be described by equation of free harmonic motion

$$\theta(t) = \theta_0 \cos(\omega_0 t)\,, \tag{10}$$

where θ_0 is the initial angular displacement and $\theta(t)$ is the angular displacement in time t.

The angular velocity is

$$\dot{\theta}(t) = -\theta_0 \omega_0 \cos(\omega_0 t)\,. \tag{11}$$

The tangential angular acceleration is

$$\ddot{\theta}(t) = -\theta_0 \omega_0^{\,2} \cos(\omega_0 t)\,. \tag{12}$$

The projection of gravity to radial direction is

$$a_{gr}(t) = g\cos(\theta(t))\,. \tag{13}$$

The projection of gravity to tangential direction is

$$a_{gt}(t) = g\sin(\theta(t))\,, \tag{14}$$

the centripetal acceleration is

$$a_c(t) = L\dot{\theta}(t)^2 \tag{15}$$

and the tangential acceleration of free pendulum is

$$a_{pt}(t) = L\ddot{\theta}(t)\,. \tag{16}$$

Substituting (7), (10) - (16) into (8) and (9) yields

$$a_r(t) = g\cos\left[\theta_0\cos(\omega_0 t)\right] + \\ + \theta_0^{\,2} g\sin^2(\omega_0 t) + a_{ext\_r}(t) + z_r(t) \tag{17}$$

and

$$a_t(t) = g\sin\left[\theta_0\cos(\omega_0 t)\right] - \\ - g\theta_0\cos(\omega_0 t) + a_{ext\_t}(t) + z_t(t) \tag{18}$$

Fig.11 illustrates these signals for initial angular displacement 0.5 rad and without external forces. Simulated and measured versions are compared. It can be seen, that $a_t(t)$ oscillates with the frequency ω_0 and $a_r(t)$ with frequency $2\omega_0$. The amplitude of $a_t(t)$ is

$$a_{t\max} = g(\theta_0 - \sin\theta_0) \tag{19}$$

and amplitude of $a_r(t)$

$$a_{r\max} = 0.5g(\cos\theta_0 - 1 - \theta_0^{\,2})\,. \tag{20}$$

These dependences are shown in Fig.12. It can be seen, that $a_t(t)$ has a very low amplitude for little angles ($\sin\theta_0$ is almost the same as θ_0). It is complicated to measure so little acceleration with usual MEMS sensors. Its value is near the resolution capability of the sensor and the signal-to-noise ratio is too low. It is much easier to process $a_r(t)$. Its amplitude is also low, but much higher than $a_t(t)$. Since the frequency of $a_r(t)$ is two times higher, sign of the angular deflection can not be derived from this signal. Hence, this signal can not be used for feedback control. In the other hand, the natural frequency can be estimated by analysis of this signal.

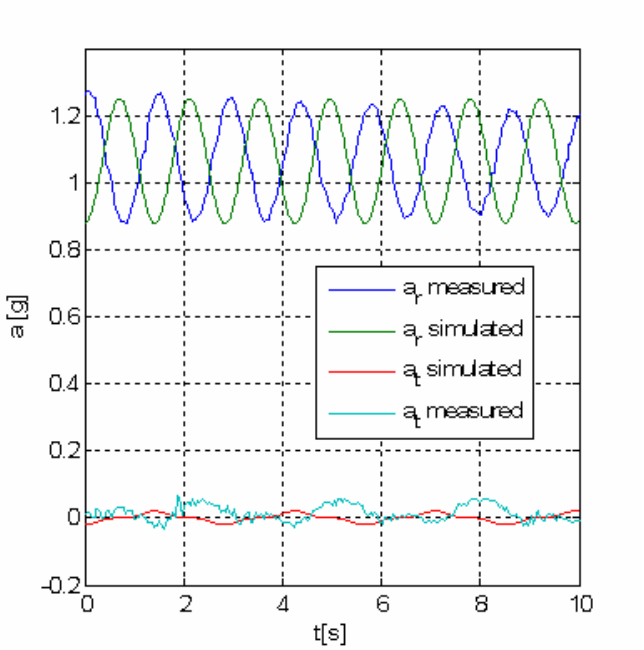


Fig.11 Simulated and measured radial and tangential acceleration for initial angular displacement 0.5rad.

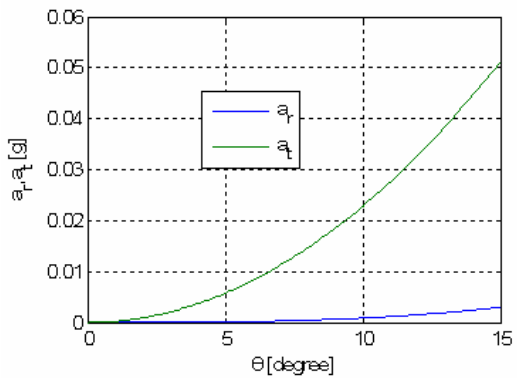


Fig.12 Dependence of the signal amplitude on the initial angular displacement using g as the unit of acceleration

Usually, it is not practical to mount the accelerometer on the payload that its axes correspond with the tangential and radial direction of the movement. Then the strongest signal consists of linear combination of tangential and radial signal. In such case, the radial component dominates and can be used for the analysis.

5. Frequency estimation

Measurement of frequency of moving pendulum with acceptable precision is one of the main steps to design the shaper correctly. There are many approaches to this difficulty. Each of them includes many methods with specific properties. A choice of the method depends on predefined values of accuracy or computational complexity.

One of the simplest estimation methods is based on a direct measurement from the signal. Maximal signal values or medium value crossings can be observed and the frequency can be computed from its time differences.

However, there is a problem because the frequency used for the proposal of algorithms has to be measured from oscillation with small displacements. This means that signal includes a high level of noise. Value of maximum or cross over a medium value cannot be precisely detected.

Frequency analysis and the Fourier transform can be used to solve this problem.

5.1 Spectral analysis of discrete signal

Transformation from time domain to frequency domain by discrete Fourier transformation is one of the most important ways to analyze numeric signal. It is possible to obtain necessary samples of spectrum by finite series of trigonometric functions applied to samples of bounded time flow.

Each nonzero sample of Fourier transformation represents one of the frequency samples that the discrete signal includes. This means that the signal consists of many frequency samples and the task is to find the dominant frequency. A way to calculate each samples of spectrum is given by equation

$$X(k)=\sum_{n=0}^{N-1}x(n)e^{-j\frac{2\pi}{N}nk};k=0,1...N-1, \qquad (21)$$

where N is number of signal samples. A distance between two adjoining samples is $T = t / N$, where t is time length of signal.

One of the most important things is to choose a correct number of samples. If we choose a selection, that does not include integer multiple of period, individual samples of spectrum will not represent corresponding frequency element. This effect caused by non-coherent sampling is called leakage.

So-called Window functions are used to avoid it. Frequency spectrum of signal modified by this special window function is more similar to real spectrum. Window function has similar length as analyzed signal, it is axially symmetrical by y-axis and it falls from maximum value to zero.

Literature [5], [6] introduces many types of windows like Bartlett's, Blackman's, Chebyshev's, Hamming's, Kaiser's window. Each of them has special properties and choice of window depends on type of signal. Window function is applied on signal before Fourier transformation. As signal and window function have the same length, they are simply multiplied per parts. An example of the simplest cosine window (Hann's window) is shown in Fig.13. Hann's window is one of the most used types of window, it belongs to group of cosine window, and its equation is

$$w(n)=0.5\left[1-\cos(2\pi\frac{n}{N})\right];n=0,1...N-1 \qquad (22)$$

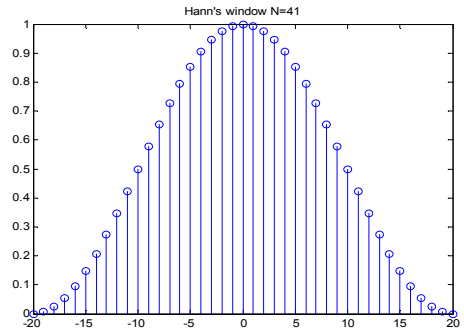


Fig.13 Hann's window

5.2 Non-parametric method of an estimate of power spectral density

More sophisticated method to recognize dominant frequency element of signal is to calculate power spectral density. There are many methods how to find an estimation of this value, some of them try to find this estimate directly from the signal, some of them find linear model of signal, that would approximate origin signal.

One of the simplest methods is direct identification. Calculation of density is given by square of amplitude spectrum of Fourier transformation (21) divided by number of samples.

$$S(k)=\frac{1}{2\pi N}\left[\sum_{n=1}^{N}x(n)e^{-jnk}\right]^2;k=0,1...N-1 \qquad (23)$$

A simple way to find these values is multiplying amplitude spectrum with its conjunction values and divide this result by number of samples. Power spectral density shows the frequency element with the highest power. But this simple method does not solve problem with leakage.

A method that is used for real measurement and uses mentioned window function is called Welch method. Main premise is division of signal to several segments with the same length. Each adjoining segment covers previous segment. It is common to use eight segments and 50% covering. On each segment Hann's or Hamming's window with same length as a segment is applied. Then square of the amplitude spectrum is calculated for each segment and finally the average from these eight blocks of data is computed.

Seeing that we use window function, we have to divide final spectrum by sum of square values of used window. We can recognize dominant frequency much easier from the spectrum made this way.

Method of Welch is able to recognize frequency elements from signal with a high value of noise. A drawback of this method is loosing sensitivity if high value of covering is used. It can make problems if adjoining samples are too close to each other. Measured signal and power spectrum are shown in Fig.14 and Fig.15.

These algorithms are considered to be applied on systems with a low computing, so time optimization of these algorithms should be used.

Once the power spectral density is calculated, a simple algorithm can localize the peak and decide if it is strong enough. Then the estimated frequency can be considered as valid or invalid.

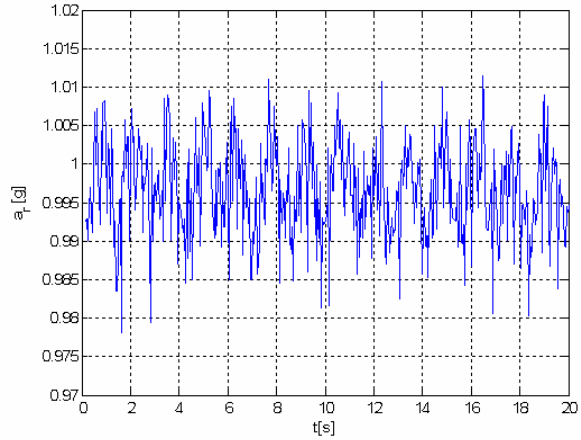


Fig.14 Signal with noise, amplitude of angular displacement is 4º

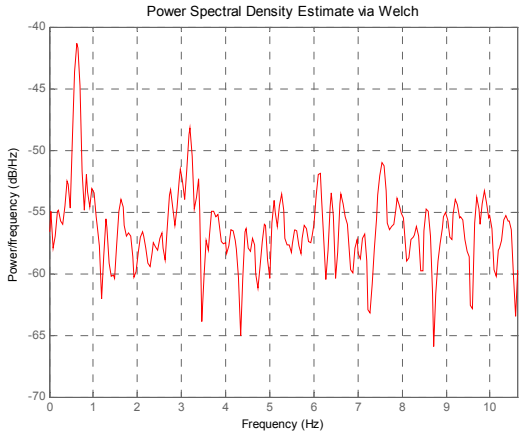


Fig.15 Power spectrum of the signal from Fig.13 using Welch method

Conclusion

A feedforward method used to reduce residual vibration of the payload of the gantry crane was presented. This method requires know system parameters, especially the natural frequency of the controlled system.

The gantry crane is the system, where the natural frequency varies. Various methods can be used to measure this frequency. A technique based on acceleration measurements can be also applied for this purpose.

Mathematical model of the accelerations measured by a MEMS accelerometer mounted on the payload is derived.

There is a signal containing information about the natural frequency. This frequency can be estimated for example by a method explained in this paper.

There are contradictory requirements for this signal. There should be no vibration at all, but in order to estimate the frequency, its amplitude should be as high as possible. Minimal angular displacement to measure the natural frequency is about 3 degrees.

In the other hand, when there is no vibration, the natural frequency is probably well estimated. When the vibration occurs, it has to be estimated again, so the vibration is used for this estimation.

This method can help reduce oscillation in the cranes and increase the labor productivity.

A drawback of this method is that the estimation can be delayed from the real value and it is sensitive to the external sources of vibration.

Acknowledgments

This work was supported by Grant Agency of Ministry of Education and Academy of Science of Slovak Republic VEGA under Grant No. 1/0690/09. The authors are pleased to acknowledge this support.

References

[1] SINGHOSE W., SINGER N., SEERING W.: Comparision of Command Shaping Methods for Reducing Residual Vibration, Published in 1995. [on-line] http://www.me.gatech.edu/inputshaping/Papers/ECC95.pdf

[2] SINGHOSE W.: Command Generation for Flexible Systems, Ph.D. thesis, Massachusetts Institute of Technology, 1997

[3] IVANOV, I., HUBINSKÝ, P.: Crane oscillation reduction, Selected Topics in Modeling and Control, vol. 4, 2004, STU Press, Bratislava, s.20-25

[4] SEDLÁČEK, M., ŠMÍD, R.: Matlab v Měření, Vydavatelství ČVUT, Praha, 2003

[5] KOTULIAKOVÁ, J., ROZINAJ, G.: Číslicové spracovanie signálov, Vydavateľstvo STU, Bratislava, 2001

[6] ONDRÁČEK, O.: Signály a sústavy, SVŠT, Bratislava, 1991

Jozef Kurilla

Slovak University of Technology
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3, 812 19 Bratislava

doc. Ing. Peter Hubinský, Phd.

Slovak University of Technology
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3, 812 19 Bratislava
Tel.: +421 2 60291608
E-mail peter.hubinsky@stuba.sk

Ing. Lukáš Palkovič

Slovak University of Technology
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3, 812 19 Bratislava
Tel.: +421 2 60291599
E-mail: lukas.palkovic@stuba.

Robust controller design for the 3Dcrane process

Thuan Nguyen Quang, Ivan Holič

Abstract
The subject of this paper is to design a robust decentralized PID controller for the 3Dcrane to stabilize motion of the cart along axes-x, axes-y using the Small Gain Theorem, and to design a robust optimal decentralized PD controller to reduce the angular deviation of payload. The obtained results were evaluated and verified in the Matlab simulink and on the real model of the 3DCrane.

Keywords: Decentralized control, Small Gain Theorem, PID controller

Introduction

The industrial crane model 3DCrane is one of the real processes built for control education and research at Department of Information Engineering and Process Control. The 3DCrane is a nonlinear electromechanical MIMO system having a complex dynamic behavior and creating challenging control problems as nonlinear, interactions between subsystems corresponding to motion of cart along the axes-x and axes-y, length of the payload lift-line dynamic changed … The technical equipments allow us to realize control crane by classical and advanced control method (see manual setup [1]).

The main aim of paper is to use knowledge of multivariable (Multi Input and Multi Output) system and stabilization of decentralized control systems [2], knowledge about robust control of linear systems in the frequency domain [3] and in the time domain to design robust PID/PD decentralized controllers stabilizing the cart motion process of 3DCrane along the both axis x/y with the different length of the payload lift-line (robust stability). Furthermore, the resulting feedback control system with designed controllers must satisfy robust performance conditions for tracking the desired position of the cart and quickly to suppress the angular deviation of the payload.

1. Overview of the 3DCrane system with artificial interaction

3DCrane system is a nonlinear electromechanical system having a complex dynamic behavior and creating challenging control problem. It is controlled from PC. Therefore it is delivered with hardware and software which can be easily mounted and installed in a laboratory. You obtain the mechanical unit together with the power supply and interface to the PC and the dedicated digital board configured in the Xilinx technology. The software operates under MS Windows using MATLAB and RTW toolbox package.

The 3DCrane setup (see Fig.1) consists of a payload hanging on a pendulum-like lift-line wound by a motor mounted on a cart. The payload is lifted and lowered in the z direction. Both the rail and the cart are capable of horizontal motion in the x direction. The cart is capable of horizontal motion along the rail in the y direction. Therefore the payload attached to the end of the lift-line can move freely in three dimensions.

The 3DCrane is driven by three DC motors. There are five identical measuring encoders measuring five state variables: the cart coordinates on the horizontal plane, the lift-line length, and two deviation angles of the payload.

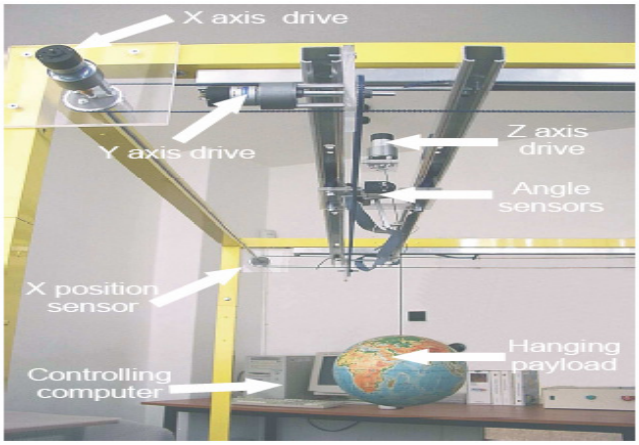


Fig.1 The 3DCrane setup

In the original model, there is no interaction between subsystems, and thus design robust controller for the motion of this 3DCrane system corresponds to design two independent robust controllers for two subsystems. The lift-line R is considered as an uncertainty.

To research the affect of the interaction between subsystems in MIMO system, we consider an artificial interaction between the control signals $U_x(s)$, $U_y(s)$ (to control movement of the crane along the x-axes and y-axes). The resulting control signals $U_{Nx}(s)$, $U_{Ny}(s)$ with interaction corresponding to subsystems are:

$$\begin{aligned} U_{Nx}(s) &= U_x(s) + \frac{0.5s}{0.3s+1}U_y(s) \\ U_{Ny}(s) &= U_y(s) + \frac{0.3s}{0.3s+1}U_x(s) \end{aligned} \qquad (1)$$

The main aim of paper is to design robust decentralized controllers for the 3DCrane system with interaction (1) satisfying two tasks: tracking a desired position of the cart and stabilizing the angular deviation of the payload on the both

case of movement of crane along axes-x and axes-y. Note that the crane is always controlled in such a way that the swinging of the payload is suppressed.

There are two following procedures to design robust decentralized controllers satisfying two above tasks.

The first procedure is to design a robust decentralized PID controller to stabilize motion of the crane along axes-x and axes-y (position (x, y) of cart). There are five sequential steps as:

- Choose a suitable control configuration, and then identify motion process of the 3DCrane along axes-x and axes-y at three operating points with lift-line z equals 0[m], 0.25[m], 0.5[m] resp. P_1, P_2, P_3
- Check the selection of the control configuration for this system (the Relative Gain Array (RGA) and Neitherlinski index (NI)).
- In the case of succeeding selection of the control configuration, create unstructured model uncertainty for motion process. Otherwise, return the first step.
- Design a robust decentralized PID controller for this process using Small Gain Theorem algorithm.
- Verify obtained result by simulating in Matlab and on the real model.

The second procedure is sequentially executed after the first procedure. In this procedure, a robust decentralized PD controller is designed to stabilize and attenuate the payload deviation of the crane. There are four sequential steps as:

- Identify and create the payload deviation model of the 3DCrane at three operating points with lift-line z equals 0[m], 0.25[m], 0.5[m] resp. P_1, P_2, P_3
- Create polytopic model of this process in the time domain.
- Design a robust decentralized PD controller which stabilizes and attenuates the payload deviation angles using BMI.
- Verify obtained result by simulating in Matlab and on the real model.

2. Design of robust decentralized PID controller stabilizing motion of the cart along axes-x and axes-y

2.1 Identification of positioning process

Result of process identification is described by the following transfer function matrices at three operating points.

Transfer function matrix of system at P1 is:

$$G_1(s)=\frac{1}{s}.\begin{pmatrix}\dfrac{2.057}{s+6.392} & \dfrac{5.815\,s-0.264}{s^2+10.57s+31.95}\\ \dfrac{6.856\,s+1.585}{s^2+24.01s+77.12} & \dfrac{3.16}{s+9.308}\end{pmatrix} \quad (2)$$

Transfer function matrix of system at P2 is:

$$G_2(s)=\frac{1}{s}.\begin{pmatrix}\dfrac{1.99}{s+6.473} & \dfrac{5.88\,s+0.475}{s^2+12.7s+32.18}\\ \dfrac{7.869\,s+2.567}{s^2+24.98s+88.74} & \dfrac{5.61}{s+16.65}\end{pmatrix} \quad (3)$$

Transfer function matrix of system at P3 is:

$$G_3(s)=\frac{1}{s}.\begin{pmatrix}\dfrac{2.739}{s+9.193} & \dfrac{10.32\,s+4.497}{s^2+16.6s+83.04}\\ \dfrac{7.341s+2.089}{s^2+25.44s+83.09} & \dfrac{5.263}{s+15.71}\end{pmatrix} \quad (4)$$

2.2 Check the selection of control configuration

In this section, we test the given selection of control configuration with nominal model $G_0(s)$ by using the *Relative Gain Array* (RGA) and *Neitherlinski index* (NI). But there is problem that, the identified process is a first-order astatic system, thus the selection of control configuration can be checked with velocity nominal model $s.G_0(s)$

The relative gain array matrix (Bristol matrix) is calculated as $\Lambda = K \times \left(K^T\right)^{-1}$ where K is the steady-state gain of the LTI model [4].

$$K = dcgain\left(s.G_o(s)\right) = \begin{bmatrix}0.3076 & 0.0320\\ 0.0251 & 0.3368\end{bmatrix} \quad (5)$$

$$\Lambda = K \times \left(K^T\right)^{-1} = \begin{bmatrix}1.0078 & -0.0078\\ -0.0078 & 1.0078\end{bmatrix} \quad (6)$$

From the resulting Bristol matrix (6), we can state that, the affect of interactions between subsystems is small. Off-diagonal elements of Bristol matrix indicate that closing the loop will change the sign of the effective gain. Then the input-output pairing is correctly selected.

The Neitherlinski index (NI) is calculated by equation

$$NI = \frac{\det K}{\prod_{i=1}^{M=2} k_{ii}} = 0.9923 > 0 \quad (7)$$

The positive value of Neitherlinski index indicates that, system is structurally stable.

The given selection of control configuration is correct.

2.3 Design of robust decentralized PID controller using small gain theorem

The following input multiplicative model of uncertainty is selected to design a robust decentralized PID controller R_i:

$$G(s) = G_0(s)\left(1 + W_i(s)\Delta_i(s)\right), i = 1,2,3 \quad (8)$$

Where $\sigma_{\mathrm{M}}\left(\Delta(j\omega)\right) = \sqrt{\lambda_M\left(\Delta^T(j\omega)\Delta(j\omega)\right)}$ is the maximum singular number of the uncertainty matrix $\Delta(j\omega)$ and $W_i(s)$ is the scalar transfer function, which guarantees normalization of uncertainties.

$$l_i(\omega) = \max_k \sigma_M\left(G^{-1}{}_0(j\omega)\left(G_0(j\omega) - G_k(j\omega)\right)\right), i = 1,2,3 \quad (9)$$

Transfer function $W_i(s)$ is selected to satisfy the following inequality:

$$\left|W_i(j\omega)\right| \geq l_i(\omega), \forall\omega;\ i = 1,2,3 \quad (10)$$

The first, controller R_i which guarantees stability and performance for the nominal plant $G_0(s)$, is designed using D-curve method (nominal stability). And system will be robust stable if the following condition is satisfied [3]:

$$\sigma_M\left(M_i(j\omega)\right) < \frac{1}{l_i(\omega)}, \forall\omega; i = 1,2,3 \quad (11)$$

where $M_i(s) = -\left(I + R_iG_0\right)^{-1} R_iG_0$

With zero degree of stability, the controller R_i is calculated as follow:

$$R_i = \begin{bmatrix} 2.33 + \frac{2.33}{s} + 4.665s & 0 \\ 0 & 5 + \frac{2}{s} + 5s \end{bmatrix} \quad (12)$$

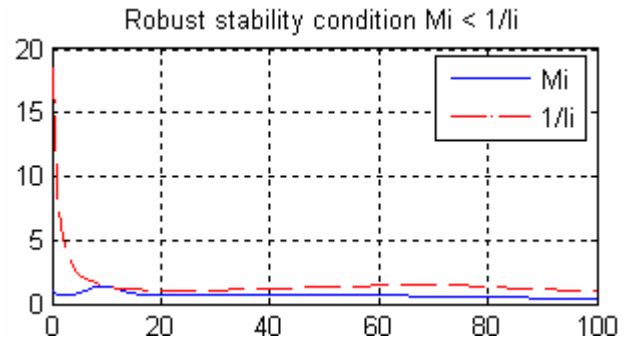


Fig.2 Verifying robust stability condition for input multiplicative model of uncertainty

From Fig.2, we can state that, the closed-loop feedback system with the PID controller R_i is robust stable. And now we verify the obtained result in Matlab simulation (see Fig.3) and on the real model (see Fig.4)

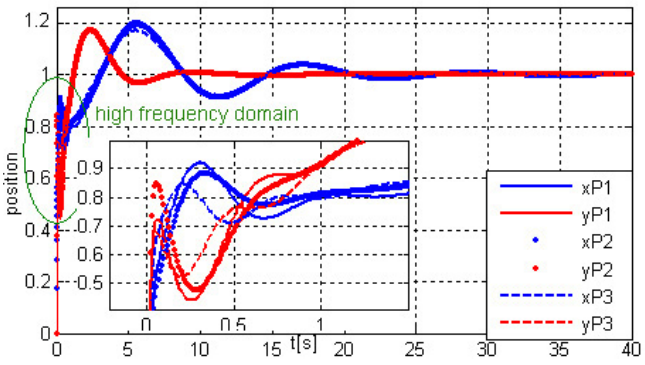


Fig.3 Position output signals with PID controller at three operating points on Matlab

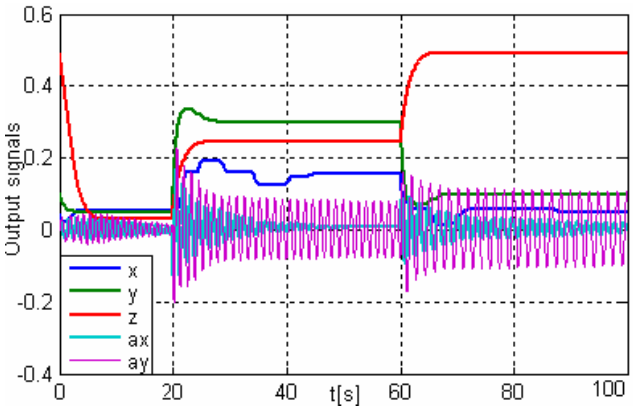


Fig.4 Output signals of system with controller without stabilizing payload deviation ax/ay on real model (z is uncertainty)

From results of simulation in Matlab and on the real model we can show that, feedback system with designed PID controllers is robust stable. However, problem is now that, the angular deviation of payload (ax and ay) has been not yet suppressed. This problem will be solved in the next section.

3. Design of robust decentralized PD controller to reduce the swinging of the payload deviation-BMI method

3.1 Identification of the angular deviation of the payload

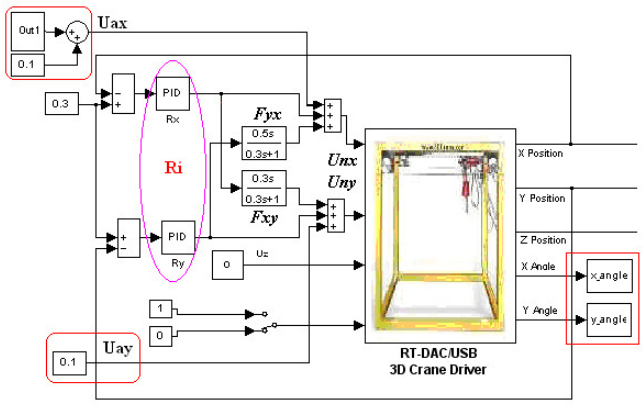


Fig.5 Identification scheme in Matlab

Result of process identification is described by the following transfer function matrices at three operating points:

$$G_{a1}(s) = \begin{pmatrix} \frac{-2.168s - 0.03114}{s^2 + 0.6812s + 49.88} & \frac{0.3381s - 0.08839}{s^2 + 0.896s + 32.48} \\ \frac{-0.2084s + 0.08203}{s^2 + 0.2387s + 47.32} & \frac{-1.678s - 0.6721}{s^2 + 0.3335s + 53.39} \end{pmatrix} \quad (13)$$

$$G_{a2}(s) = \begin{pmatrix} \frac{-1.548s - 0.08906}{s^2 + 0.3721s + 34.13} & \frac{0.2578s - 0.06426}{s^2 + 0.876s + 17.39} \\ \frac{-0.1885s + 0.02379}{s^2 + 0.1219s + 31.1} & \frac{-1.521s + 0.4889}{s^2 + 0.2426s + 34.2} \end{pmatrix} \quad (14)$$

$$G_{a3}(s) = \begin{pmatrix} \frac{-0.9902s + 0.02326}{s^2 + 0.2614s + 26.67} & \frac{-0.1969s + 0.595}{s^2 + 0.5075s + 29.71} \\ \frac{-0.1095s + 0.1075}{s^2 + 0.2331s + 27.19} & \frac{-1.169s + 0.2218}{s^2 + 0.1367s + 26.4} \end{pmatrix} \quad (15)$$

3.2 Polytopic model for the angular deviation of payload process in the time domain

Each of the transfer function matrices (13), (14) and (15) can be transformed to linear time invariant continuous system $(A_i, B_i, C, D = 0)$; $i = 1,2,3$, which are considered three vertices of the polytopic system. Our task is to find remain vertex (the fourth vertex) of this polytopic system.

We shall consider the following affine linear time invariant continuous time uncertain system

$$\begin{aligned} \dot{x} &= \left(A + \theta_1 \tilde{A}_1 + \theta_2 \tilde{A}_2\right)x + \left(B + \theta_1 \tilde{B}_1 + \theta_2 \tilde{B}_2\right)u \\ y &= Cx \end{aligned} \quad (16)$$

where $\underline{\theta}_j \le \theta_j \le \overline{\theta}_j$; $j = 1,2$; $C = \begin{bmatrix} 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \end{bmatrix}$

Polytopic model is defined as follow:

$$\begin{aligned} \dot{x} &= A(\xi)x + + B(\xi)u \\ y &= Cx \end{aligned} \quad (17)$$

where

$$A(\xi) = \sum_{i=1}^{4} \xi_i A_i \;,\; B(\xi) = \sum_{i=1}^{4} \xi_i B_i \;,\; \sum_{i=1}^{4} \xi_i = 1 \;,\; \xi_i \ge 0; i = 1 \ldots N = 4$$

Vertices of polytopic system are created by the combination of extreme values of θ_j.

$$\begin{aligned} A_i &= A + \theta_1\tilde{A}_1 + \theta_2\tilde{A}_2 \\ B_i &= B + \theta_1\tilde{B}_1 + \theta_2\tilde{B}_2 \quad , \quad i = 1...N = 4 \end{aligned} \tag{18}$$

We suppose that, the extreme values of $\overline{\theta}_j = -\underline{\theta}_j = 1$, $j = 1,2$. Polytopic system will be obtained if for $N! = 24$ combinations of extreme value θ_j, by solving equation system $A_i = A + \theta_1\tilde{A}_1 + \theta_2\tilde{A}_2$, $i = 1...(p+1) = 3$ we have matrices $A, \tilde{A}_1, \tilde{A}_2$ for which the maximal eigenvalue of respective matrix A_4 will be minimal.

In the case of the payload angular deviation process, the best combination of θ_1, θ_2 is as follow:

$$\begin{bmatrix} 1 & \underline{\theta}_1 & \underline{\theta}_2 \\ 1 & \overline{\theta}_1 & \underline{\theta}_2 \\ 1 & \overline{\theta}_1 & \overline{\theta}_2 \\ 1 & \underline{\theta}_1 & \overline{\theta}_2 \end{bmatrix} \tag{19}$$

Consider the uncertain system (18) for the payload deviation process, where

$$B = \begin{bmatrix} -0.0039 & -1.5789 & 0 & 0 & \dots \\ 0 & 0 & 0.2533 & 0.0706 & \dots \\ 0.0948 & -0.1589 & 0 & 0 & \\ 0 & 0 & -0.2251 & -1.4236 & \end{bmatrix}^T$$

$$\tilde{B}_1 = \begin{bmatrix} 0.0601 & 0.3098 & 0 & 0 & \dots \\ 0 & 0 & 0.0121 & -0.0402 & \dots \\ -0.0291 & 0.0099 & 0 & 0 & \\ 0 & 0 & 0.5805 & 0.0783 & \end{bmatrix}^T$$

$$\tilde{B}_2 = \begin{bmatrix} -0.0329 & 0.2789 & 0 & 0 & \dots \\ 0 & 0 & 0.3296 & -0.2274 & \dots \\ 0.0419 & 0.0395 & 0 & 0 & \\ 0 & 0 & -0.1336 & 0.1759 & \end{bmatrix}^T$$

$$A = \begin{bmatrix} 0 & -38.272 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -0.4713 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -31.0945 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -0.7018 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -37.2527 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -0.2359 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -39.8975 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -0.2351 \end{bmatrix}$$

$$\tilde{A}_1 = \begin{bmatrix} 0 & 7.8717 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.1546 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 7.5432 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.01 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 8.1087 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.0584 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 9.5972 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.0454 \end{bmatrix}$$

$$\tilde{A}_2 = \begin{bmatrix} 0 & 3.7335 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.0553 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -6.1602 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0.1842 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1.9542 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -0.0556 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 3.8992 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.053 \end{bmatrix}$$

3.3 Robust decentralized optimal PD controller design using BMI

Consider PD control law as follow:

$$u = Fy + F_d\frac{dy}{dt} = FCx + F_dC_d\dot{x} \tag{20}$$

Closed-loop feedback system with PD controller (19) is:

$$M_d(\xi)\dot{x} = A_c(\xi)x \tag{21}$$

where $M_d(\xi) = I - B(\xi)F_dC_d$, $A_c(\xi) = A(\xi) + B(\xi)FC$

Consider cost function as follow

$$J = \int_0^\infty \left(x^TQx + u^TRu + \dot{x}^TS\dot{x}\right)dt \tag{22}$$

The closed-loop feedback system (21) with the PD controller (20) is robust stable and guarantees the cost function (22) if and only if there exist matrices $P = \sum_{i=1}^{4}\xi_iP_i$, $P_i > 0$; $i = 1...N = 4$, H , G , F and F_d then the following inequality is satisfied [5]

$$\begin{bmatrix} A_{ci}^TH^T + HA_{ci} + Q + C^TF^TRFC & \left(P_i - M_{di}H + G^TA_{ci}\right)^T \\ P_i - M_{di}H + G^TA_{ci} & -M_{di}^TG - G^TM_{di} + S \end{bmatrix} < 0 \tag{23}$$

For given cost function with $R = 1.2*I$, $Q = S = 10^{-5}*I$ by using BMI to solve (23), the PD controller is obtained as follow:

$$F = \begin{bmatrix} 2.8397 & 0 \\ 0 & 3.0639 \end{bmatrix}, F_d = \begin{bmatrix} 0.1671 & 0 \\ 0 & 0.3468 \end{bmatrix} \tag{24}$$

And now we verify the obtained result in Matlab simulation (see Fig.6) and on the real model (see Fig.7 and Fig.8)

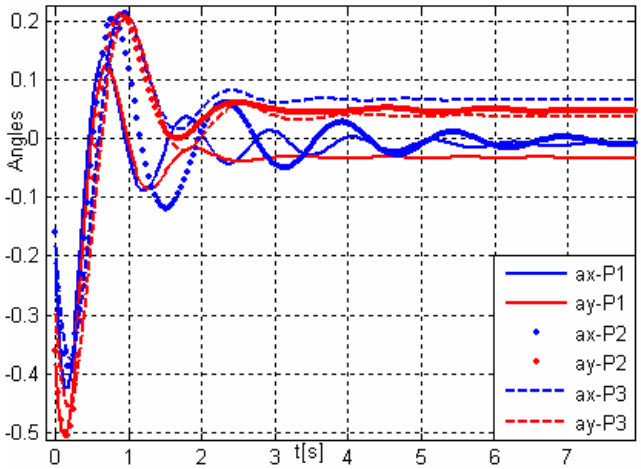


Fig.6 Verify robustness of feedback system with PD controller at three operating points in Matlab

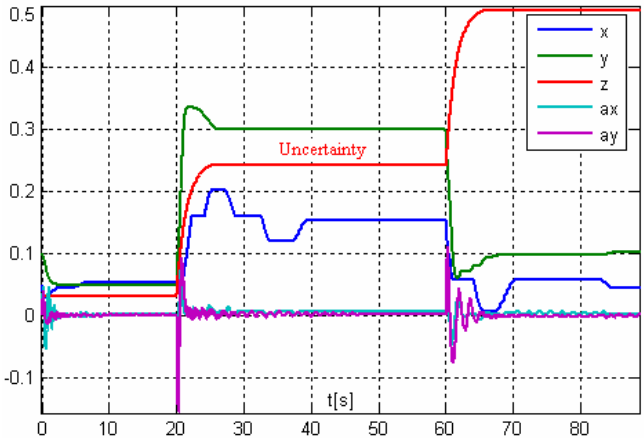


Fig.7 Verify robustness of feedback system with time varying uncertainty

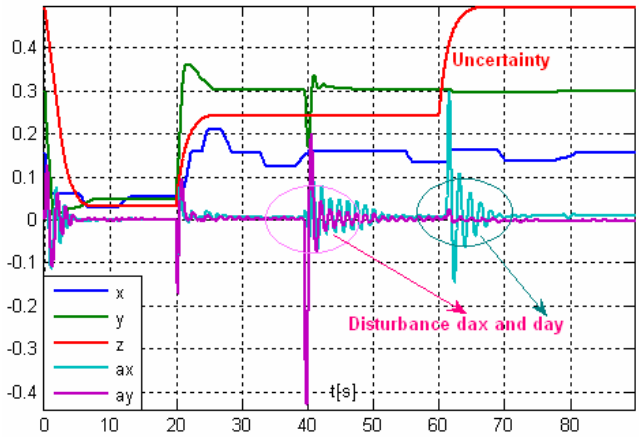


Fig.8 Verify robustness of feedback system with time varying uncertainty and disturbances

From simulating results on real model, we can conclude that, the cart of the crane tracks a desired position and the angular deviation of the payload is suppressed.

Conclusion

In this paper, we have researched and applied successfully the knowledge of multivariable system, stabilization of decentralized control systems and the knowledge of robust control theory in the frequency domain and also in the time domain to control the 3DCrane system.

There were two sequential procedures to design robust decentralized controllers for the 3DCrane system.

In the first procedure, we have identified process of the cart motion along axis x and y. The identification was executed at three operating points corresponding to following length of the payload lift-line: 0, 0.25 and 0.5 [m]. From resulting identified three transfer function matrices, I created model uncertainties in the frequency domain. With input multiplicative model of uncertainties, we have designed robust decentralized PID controller to stabilize the cart motion and track the desired position. The resulting PID controllers are verified in the Matlab simulink and on the real model. The cart was at the desired position, but the payload deviation angles were so big. This problem was solved in the second procedure.

With the designed PID controller in the first procedure, we identified payload angular deviation process at the above three operating points. The results of process identification are three matrix transfer functions. They were transformed to state space form of system and then the polytopic system in the time domain were created. Next, we designed robust optimal decentralized PD controller via BMI to reduce payload angular deviation. The results are verified in the Matlab simulink and on the real model.

Thus, the complete feedback control system with two designed PID/PD controllers was robust stable and achieved robust performance. The robustness of the complete feedback control system was verified (see Fig.8 and Fig.9) when the length of the lift-line was dynamically changed and there were disturbances to angular deviation. The cart of the crane was tracking the desired position and the angular deviation of the payload is suppressed. The main aim of paper has been obtained.

Acknowledgment
The work has been supported by Grant N1/0544/09 of the Slovak Scientific Grant Agency.

References

[1] INTECO Ltd.: *3DCrane User's manual. RT-CON with USB board version*. [on line]. INTECO Ltd.. URL: < http://users.utcluj.ro/~tbuzdugan/App/inteco.pdf>.

[2] Zhining, G., Mahamed, A.: 1997. Stabilization of decentralized control systems. *Journal of Mathematical Systems, Estimation, and Control,* 1997, Vol. 7, No.1, pp. 1-16.

[3] Veselý, V., Kozáková, A., Grman, Ľ.: 2006. Robust control of linear systems in the frequency domain. *Journal of ELECTRICAL ENGINEERING*, 2006, Vol. 57, No. 6, pp. 338-346.

[4] Tham, M. T.: 1999. Multivariable control: An Introduction to decoupling control. Department of Chemical and Process Engineering, University of Newcastle upon Tyne.

[5] Rosinová, D., Veselý, V.: (2007). Robust PID decentralized controller design using LMI. *International Journal of Computers, Communications & Control,* 2007, Vol. II, No. 2, pp. 195-204.

Ing. Thuan Nguyen Quang
Ing. Ivan Holič

Slovak University of Technology in Bratislava
Faculty of Electrical Engineering and Information Technology
Institute of control and industrial informatics
Ilkovičova 3
812 19 Bratislava
Tel.: +421 2 60291 544
E-mail: thuan.quang@stuba.sk, ivan.holic@stuba.sk

Robust controller design for magnetic levitation model

Mária Hypiusová, Jakub Osuský

Abstract
The paper deals with design of robust PID controller for unstable SISO system in the frequency domain. The method considers affine aditive uncertainty and performance specification in terms of phase margin and the modification of Neimark D-partition method which ensures the desired phase margin. The practical application is illustrated by the robust PID controller design for the magnetic levitation model.

Keywords: robust control, D-partition, phase margin, unstable system

Introduction

For many real processes a controller design has to cope with the effect of uncertainties, which very often cause a poor performance or even instability of closed-loop systems. The reason for that is a perpetual time change of parameters (due to aging, influence of environment, working point changes *etc.*), as well as unmodelled dynamics. The former uncertainty type is denoted as the parametric uncertainty and the latter one the dynamic uncertainty. A controller ensuring closed-loop stability under both of these uncertainty types is called a robust controller. A lot of robust controller design methods are known from the literature [1, 3] in the time- as well as in the frequency domains.

From the point of view of control engineering, magnetic levitation (maglev) systems are challenging because of the nonlinear nature of the plant dynamics, the very small degree of natural damping in the process, the strict positioning specifications often required by the application and the system dynamics are open-loop unstable.

Maglev technology has a wide range of applications, for instance, high-speed transportation systems [7], seismic attenuators for gravitational wave antennas [13], self-bearing blood pumps [10] for use in artificial hearts, haptic interfaces [2], photolithography devices for semiconductor manufacturing [9] and microrobots [8].

The design method presented in this paper is a graphical approach based on the D-partition method [11]. To achieve the desired phase margin, controllers are usually designed using Bode characteristics [5, 6]. The modification consists in ensuring the desired phase margin using D-partition method. The method is applied for a nominal model. The designed controller is verified using robust stability conditions for inverse additive uncertainty.

1. Magnetic levitation model

Levitation is the stable equilibrium of an object without contact and can be achieved using electric or magnetic forces. In a magnetic levitation, or maglev, system a ferromagnetic object is suspended in air using electromagnetic forces. These forces cancel the effect of gravity, effectively levitating the object and achieving stable equilibrium.

The model of magnetic levitation shown in Fig. 1 consists of a coil levitating a steel ball in magnetic field [4]. The position of the steel ball is sensed by an inductive linear position sensor connected to A/D converter. The coil is driven by a power amplifier connected to D/A converter. The model is connected to the PC via an universal data acquisition card, like the HUMUSOFT AD512 or MF614.

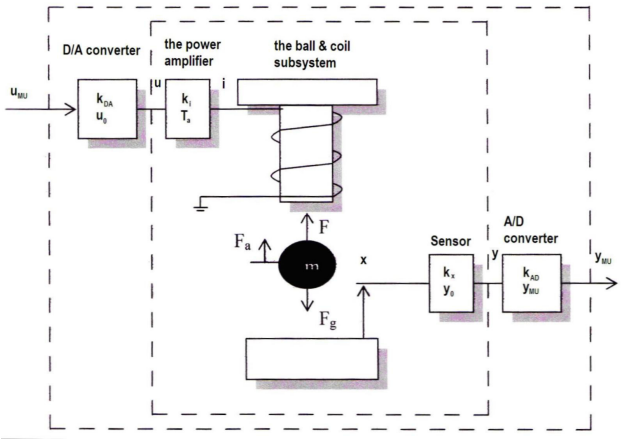


Fig. 1 Scheme of the magnetic levitation model

Fig. 2 The magnetic levitation model

The scheme shows that the model interface can be considered at two different levels:

- physical level – input and output voltage to the coil power amplifier and from the ball position sensors
- logical level – voltage converted by the data acquisition card and scaled to +-1 machine unit [MU].

The basic control task is to control the position of the ball freely levitation in the magnetic field of the coil. The real magnetic levitation model is shown in Fig. 2.

2. Robust PID controller design with desired phase margin

Consider the closed-loop feedback system shown in Fig. 3, where G_R(s) is transfer function of a PID controller; $G_P(s)$ is a transfer function of the real plant; w, e, u and y are the reference, control error, manipulated variable and output of the plant, respectively.

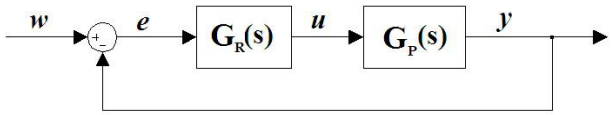


Fig. 3 Classical feedback system

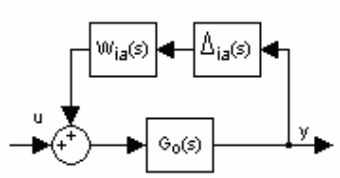


Fig. 4 Inverse additive uncertainty

Consider a perturbed plant with unstructured inverse additive uncertainties (Fig. 4) in the form

$$G_p(s) = G_0(s)(I + w_{ia}(s)\Delta_{ia}(s)G_0(s))^{-1} \quad (1)$$

where $G_0(s)$ is the nominal model; $w_{ia}(s)$ is weighting stable scalar transfer function ($|w_{ia}(j\omega)| \geq l_{ia}(\omega)$, for $\forall\omega$) and $\Delta_{ia}(s)$ is normalized uncertainty ($\Delta_{ia}(s) \leq 1$).

For this uncertainty type it is possible to calculate weighting function $l_{ia}(\omega)$ as follows:

$$l_{ia}(\omega) = \max_k \sigma_M(G_k(j\omega)^{-1} - G_0(j\omega)^{-1}) \quad (2)$$

The robust stability condition derived using M-delta structure [12] will be used in the following form

$$\sigma_M(M_0(s)) < \frac{1}{l_{ia}(\omega)} \quad (3)$$

where $M_0(s) = \dfrac{G_0(s)}{1 + G_R(s)G_0(s)}$ (4)

The condition (3) is verified graphically.

The nominal model can be obtained e.g. by *N* identifications of the plant (in *N* working points) by taking mean values of the nominator and denominator coefficients, respectively:

$$G_0(s) = \frac{(B_1(s) + \ldots + B_N(s))/N}{(A_1(s) + \ldots + A_N(s))/N} \quad (5)$$

Consider the transfer functions of a PID controller

$$G_R(s) = k + \frac{k_i}{s} + k_d s \quad (6)$$

The problem studied in this paper can be formulated as follows. For perturbed system $G_P(s)$ described by (1) a robust PID controller G_R(s) is to be designed for nominal system using Neimark D-partition so that not only stability will be ensured but performance in term of phase margin too. If the designed controller satisfies condition (3), then PID controller is robust and stabilise a perturbed plant with unstructured additive uncertainties.

The characteristic equation for the nominal model is

$$1 + G_R(s)G_0(s) = 0 \quad (7)$$

A small modification of (7) yields

$$k + \frac{k_i}{s} + k_d s = -\frac{A(s)}{B(s)} \quad (8)$$

Using substitution $s = j\omega$ it is possible to obtain real and imaginary parts:

$$RE: k = -\frac{A(j\omega)}{B(j\omega)}$$

$$IM: -\frac{k_i}{\omega} j + k_d j\omega = -\frac{A(\omega)}{B(\omega)} \quad (9)$$

If ω is changing step by step in the interval $\omega \in (0, \infty)$, it is possible to calculate from real part of (9) a frequency dependent vector of complex numbers, which plotted in complex plane creates D-curve for parameter k. Similarly it is with imaginary part of (9), from which k_i or k_d can be obtained but not both at once. In one step it is possible to plot D-curve for parameters k and k_i (PI controller) or k and k_d (PD controller).

A small modification of characteristic equation yields

$$1 + G_R(s)G_0(s)e^{-j\varphi} = 0 \quad (10)$$

It is possible to rotate the frequency characteristics of a system, where φ is the angle of desired rotation in radians. Then the D-curves calculated from (10) are

$$RE: k = -\frac{A(j\omega)}{B(j\omega)e^{-j\varphi}}$$

$$IM: -\frac{k_i j}{\omega} + k_d j\omega = -\frac{A(j\omega)}{B(j\omega)e^{-j\varphi}} \quad (11)$$

and controller parameters are chosen directly from the D-curves. The designed controller will ensure phase margin equal to the angle φ.

The PID controller design consists of two steps: in the first step, PD controller can be designed and in the second step, PI controller design can by applied for the plant with the PD controller. The final PID controller is than calculated as follows

$$G_R(s)=(k_1+k_d s)(k_2+\frac{k_i}{s})= \\ =(k_1k_2+k_dk_i)+\frac{k_ik_1}{s}+k_dk_is \quad (12)$$

In this way, a controller for unstable plant can be designed, if this plant can be stabilized by a PD controller. Hence in the first step, a PD controller is used for stabilization and a PI controller ensures desired phase margin and eliminates steady state offset.

3. Closed-loop identification of magnetic levitation model

The classical black box identification of magnetic levitation model is not possible, because this system is unstable. Therefore it can be encompassed closed-loop identification. From Fig. 3

$$\frac{Y(s)}{W(s)}=\frac{G_R(s)G_p(s)}{1+G_R(s)G_p(s)} \quad (13)$$

$$\frac{U(s)}{W(s)}=\frac{G_R(s)}{1+G_R(s)G_p(s)} \quad (14)$$

After dividing (13) by (14) we obtain transfer function $G_P(s)$.

We have chosen three working points, which are given by the output variable y_{MU} which corresponds to the position $x\,[mm]$:

WP1: $y_{MU}=0{,}3$ $x=1{,}89[mm]$

WP2: $y_{MU}=0{,}4$ $x=2{,}52[mm]$

WP3: $y_{MU}=0{,}5$ $x=3{,}15[mm]$

The transfer functions of a position of the magnetic levitation model in three working points by the closed-loop identification are:

$$G_{p_1}=\frac{-3{,}088s^3+3435s^2+7{,}875.10^5 s+1{,}003.10^6}{s^4+64{,}74s^3+1{,}321.10^4 s^2-3{,}633.10^5 s-4{,}125.10^5}$$

$$G_{p_2}=\frac{-5{,}108s^3+'1710s^2+9{,}39.10^5 s+2{,}288.10^6}{s^4+89{,}69s^3+1{,}521.10^4 s^2-2{,}351.10^5 s-7{,}954.10^5}$$

$$G_{p_3}=\frac{-25{,}82s^3+885{,}3s^2+2{,}001.10^6 s+3{,}149.10^6}{s^4+174{,}6s^3+2{,}132.10^4 s^2-2{,}317.10^5 s-6{,}436.10^5} \quad (15)$$

Each system has one unstable real root.

4. Design of robust PID controller for magnetic levitation model

Let the required phase margin be $\varphi_z=60^\circ$. The design was applied on the nominal model obtained by (5)

$$G_0(s)=\frac{-11{,}34s^3+2010s^2+1{,}24.10^6 s+2{,}15.10^6}{s^4+109{,}7s^3+16550s^2-2{,}43.10^5 s-6{,}17.10^5} \quad (16)$$

The D-curves for k_1 and k_d are calculated and depicted in Fig. 5.

In the first step it is not necessary to choose parameters from blue line because we just want to stabilize the system. System will be stable if the controller parameters will be chosen from a stable area. The red line represents the stability bound. From Fig. 5 we have chosen PD controller parameters $k_1=6$ and $k_d=0{,}04$. The poles of characteristic equation with this PD controller are

-606,9±356,5i -100,26 and -1,71

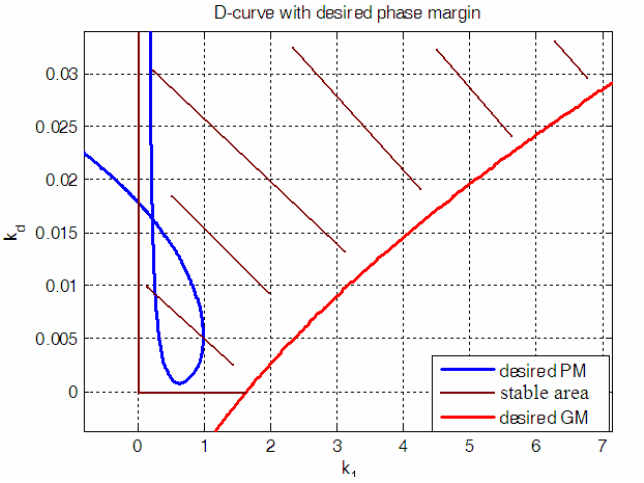


Fig. 5 D-curve for parameters k_1 and k_d

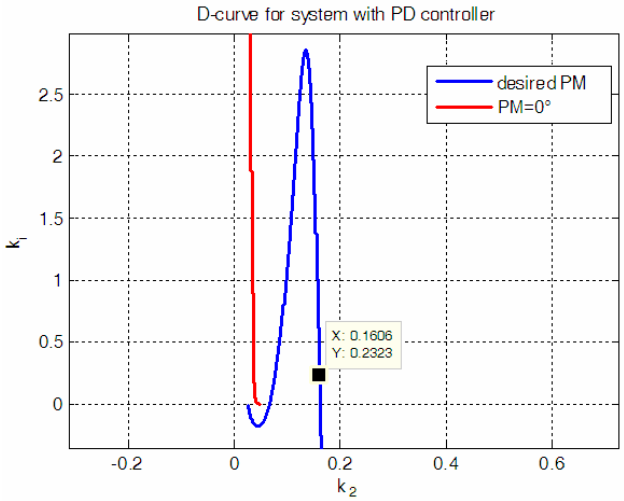


Fig. 6 D-curve for parameters k_2 and k_i

In the second step PI controller is designed for system consisting from plant and PD controller. The D-curves for k_2 and k_i are calculated and together depicted in Fig. 6.

From Fig. 6 we have chosen parameters from blue line which ensures the desired phase margin $\varphi=60^\circ$. The designed PI controller has following parameters: $k_2=0{,}1606$ and $k_i=0{,}2323$. The final controller calculated according to (12) is:

$$G_R(s)=0{,}9729+\frac{1{,}392}{s}+0{,}006424s \quad (17)$$

In Fig. 7 there are depicted Bode characteristics which show that the desired phase margin has been achieved.

The poles of characteristic equation with the designed PID controller are:

-39,31 ± 153,6i -1,75 ± 0,55i and -38,22

Fig. 8 shows that robust stability condition (3) is satisfied.

If the system with the designed controller does not fulfill robust stability condition than it is necessary to increase desired phase margin and repeat controller design.

Fig. 9, 10 and 11 show the results of experiments on the real model with the designed robust controller. Step responses were made in three working points.

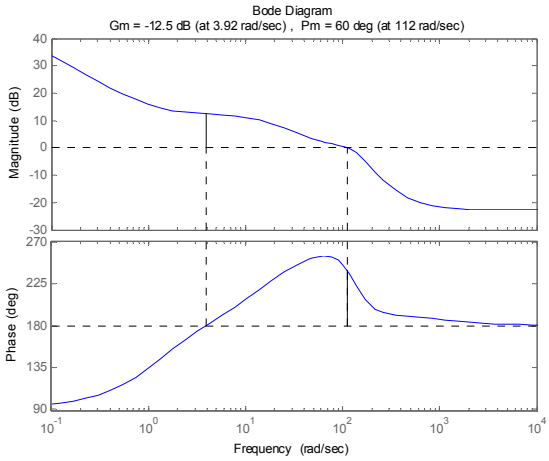


Fig. 7 Bode characteristics for plant of magnetic levitation with the designed controller

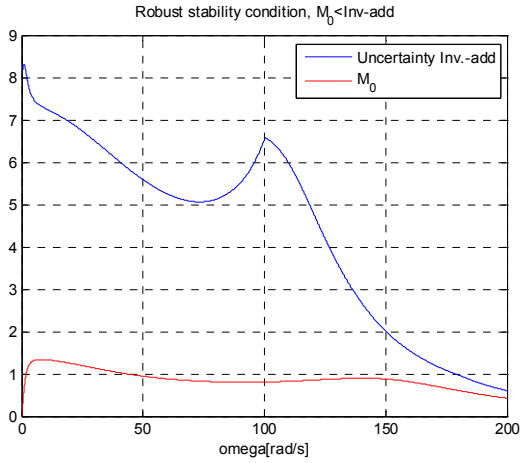


Fig. 8 Robust stability condition

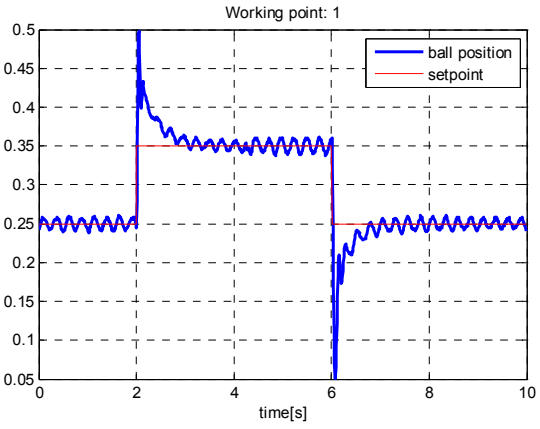


Fig. 9 Step responses in the first working point

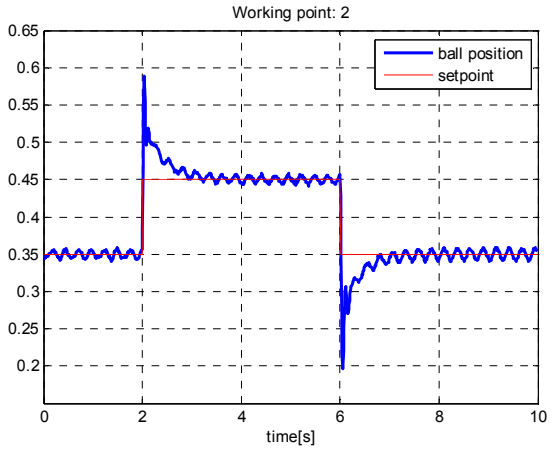


Fig. 10 Step responses in the second working point

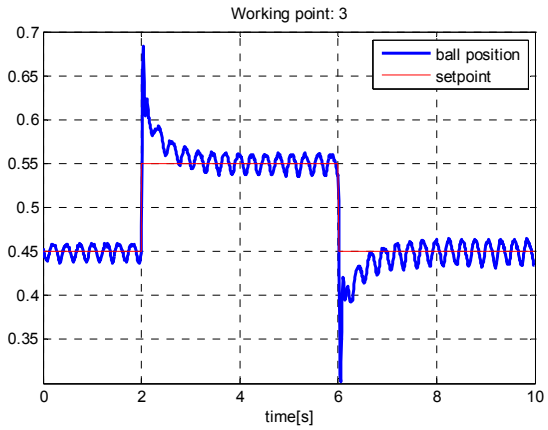


Fig. 11 Step responses in the third working point

Conclusion

In this paper a modification of the Neimark D-partition method for the unstable system of magnetic levitation model has been presented. System was identified in closed loop and described using unstructured uncertainty model. The proposed method was applied on the nominal model which ensures not only stability but performance in terms of phase margin, too. Robust stability condition derived using M-delta structure was verified. The designed robust controller was applied on real magnetic levitation model. The proposed method is graphical, interactive and insightful and it is useful for stable and unstable perturbed systems.

Acknowledgments

The work on this paper has been supported by the VEGA Grant No. 1/0544/09.

References

[1] ACKERMAN, J. 1997. Robust Control - Systems with Uncertain Physical Parameters. *Springer - Verlag London Limited*, 406 s. ISBN 0-387-19843-1, 1997.

[2] BERKELMAN, P.J., R.L. HOLLIS: Lorentz magnetic levitation for haptic interaction: Device design, performance, and integration with physical simulations, *International Journal of Robotics Research*, Vol. 19, No. 7, pp. 644-667, 2000.

[3] BHATTACHARYYA, S. P.; CHAPELLAT, H.; KEEL, L. H.: Robust Control: The parametric Approach. *Prentice Hall,* 647 s. ISBN 0-13-781576-X, 1995.

[4] CE 152 Magnetic levitation model – education manual. *Humusoft s.r.o.* 2002.

[5] FUNG, H.W., Q.G. WANG, T.H. LEE: PI Tuning in terms of gain and phase margins, *Automatica*, 34, pp. 1145-1149, 1998.

[6] HO, W.K., C.C. HANG, L. CAO: Tuning of PID controllers based on gain and phase margin specifications, *Automatica*, 31, pp. 497-502, 1995.

[7] HOLMER, P.: Faster than a speeding bullet train, *IEEE Spectrum*, Vol. 40, No. 8, pp. 30-34, 2003.

[8] KHAMESEE, M.B., N. KATO, Y. NOMURA, T. NAKAMURA: Design and control of a microrobotic system using magnetic levitation, *EEE-ASME Transactions on Mechatronics*, Vol. 7, No. 1, pp. 1-14, 2002.

[9] KIM, W.J., D.L. TRUMPER: High-precision magnetic levitation stage for photolithography, *Precision Engineering*, Vol. 22, No. 2, pp. 66-77, 1998.

[10] MASUZAWA, T., S. EZOE, T. KATO, Y. OKADA: Magnetically suspended centrifugal blood pump with an axially levitated motor, *Artificial Organs*, Vol. 27, No. 7. pp. 631-638, 2003.

[11] NEIMARK, Y.I.: Robust stability and D-partition, *Automation and Remote Control 53* (7), pp. 957–965, 1992.

[12] SKOGESTAD, S., POSTLETWAITE, I.: Multivariable feedback control: analysis and design, *John Wiley & Sons*, 1996.

[13] VARVELLA, M., E. CALLONI, L. DI FIORE, L. MILANO, N. ARNAUD: Feasibility of magnetic suspension for second generation gravitational wave interferometers, Astroparticle Physics, Vol. 21, No. 3, pp. 325-335, 2004.

Ing. Mária Hypiusová, PhD.
Ing. Jakub Osuský

Slovak University of Technology
Faculty of Electrical Engineering
and Information Technology
Institute of Control and Industrial Informatics
Ilkovičova 3
812 19 Bratislava
E-mail: maria.hypiusova@stuba.sk, jakub.osusky@stuba.sk

Archív Archive

2001 AT&P journal PLUS1: Adaptívne a nelineárne riadenie systémov (tlačená verzia)
Adaptive and nonlinear control systems (printed version)

AT&P journal PLUS2: Robotika, mechatronika, diskrétne výrobné systémy (tlačená verzia)
Robotics, mechatronics, discrete manufacturing systems (printed version)

2002 AT&P journal PLUS3: Robustné systémy riadenia (tlačená verzia)
Robust control systems (printed version)

2003 AT&P journal PLUS4: Samonastavujúce sa systémy v riadení procesov (tlačená verzia)
Selftuning systems in process control (printed version)

2004 AT&P journal PLUS5: Robotické systémy (elektronická – CD verzia)
Robotics systems (electronic – CD version)

2005 AT&P journal PLUS6: Mechatronika (elektronická – CD verzia)
Mechatronics (electronic – CD version)

AT&P journal PLUS7: Umelá inteligencia v praxi (elektronická – CD verzia)
Artifical intelligence in Practise (electronic – CD version)

2006 AT&P journal PLUS 1: Mechatronické systémy (elektronická – CD verzia)
Mechatronic systems (electronic – CD version)

AT&P journal PLUS 2: Inteligentné meracie systémy (elektronická – CD verzia)
Intelligent measurement systems (electronic – CD version)

2007 AT&P journal PLUS 1: MMaMS'2007 (elektronická – CD verzia)
MMaMS'2007 (electronic – CD version)

AT&P journal PLUS 2: Riadenie procesov (elektronická – CD verzia)
Process Control (electronic – CD version)

2008 AT&P journal PLUS 1: Mobilné robotické systémy (elektronická – CD verzia)
Mobile robotic systems (electronic – CD version)

AT&P journal PLUS 2: Riadenie v energetike (elektronická – CD verzia)
Control of Power Systems (electronic – CD version)

2009 AT&P journal PLUS 1: Inteligentné pohybové systémy (elektronická – CD verzia)
Intelligent motion control systems (electronic – CD version)

AT&P journal PLUS 2: Riadenie procesov (elektronická – CD verzia)
Process control (electronic – CD version)