P1-TS fuzzy scheduling control system design using local pole placement and interval analysis

J. SMOCZEK*

AGH University of Science and Technology, Faculty of Mechanical Engineering and Robotics, 30 Mickiewicza Ave., 30-059 Kraków, Poland

Abstract. The linear parameter-varying (LPV) discrete-time model based design of a fuzzy scheduling control scheme is developed through incorporating the advantages of P1-TS theory, and applying the local pole placement method and interval analysis of closed-loop system polynomial coefficients. The synthesis of fuzzy scheduling control scheme is proposed in the form of iterative procedure, which enables to find the appropriate number of intervals of a fuzzy interpolator ensuring that a family of local linear controllers places closed-loop polynomial coefficients within a desired range. The computational complexity of multidimensional fuzzy scheduling control scheme synthesis is reduced using a fundamental matrix method and recursive procedure for fuzzy rule-based interpretation. The usability of the proposed method is illustrated by an implementation example and experimental results obtained on a laboratory scaled overhead crane.

Key words: P1-TS fuzzy system, scheduling control, LPV discrete-time system, local pole placement, interval analysis.

1. Introduction

The variation of operating conditions, time varying parameters and external influences of a nonlinear plant involve the robustness of a control scheme. The well-known and popular method is the gain scheduling control which is effectively applied in a wide range of applications where the system dynamic varies within the known interval of scheduling variables. The gain scheduling design is frequently an iterative process, based on the series expansion linearization of a system about its equilibrium points at which a family of local linear controllers is determined to satisfy the system robustness. The survey of scheduled control synthesis is presented in [1, 2], however, the effectiveness of a gain scheduling control system is still the subject of attention in the recent works [3–6]. The numerous research works demonstrate the usefulness of a fuzzy scheduled based control in different applications [7–9], developed especially based on the Takagi-Sugeno (TS) model [10]. Also, numerous authors adopted the interval mathematics [11, 12] for robust control systems synthesis [13–15]. Interval analysis is implemented for modeling interval systems and designing robust controller according to the iterative procedures [16–18], Monte Carlo technique [19], or through applying the soft computing methods, e.g. evolutionary algorithm (EA) [20] and artificial neural network [21].

Although many works concern the problem of synthesis the linear or fuzzy interpolation based scheduled control schemes, the less attention in the literature is addressed to the problem of desired location of nominal operating points associated with scheduled controllers. This problem is solved using interval analysis of closed-loop system poles employed in genetic [22] and iterative [16] algorithms used to find the appropriate number of local controllers linearly interpolated within the bounds of the one exogenous variable. The fuzzy clustering method [23] and EA [24] are proposed to reduce the fuzzy partitions of the TS fuzzy gain scheduling system based on the assessment of control system performance. In [25, 26] the EA is employed to minimize the number of fuzzy sets specified for scheduling variables, rope length and mass of a payload, and design a TS fuzzy controller for an anti-sway crane control system. In [27, 28] the iterative procedure utilizing the interval analysis of closed-loop polynomial coefficients is developed for fuzzy interpolation based discrete-time control scheme implemented on a laboratory scaled overhead crane. However, the mentioned methods does not show directly a general solution for multi-input multi-output (MIMO) fuzzy scheduler design.

In this paper the LPV discrete-time scheduling control scheme design is developed based on a local pole placement method, interval analysis of closed-loop polynomial coefficients and fuzzy interpolator elaborated based on the P1-TS theory proposed in [29–31]. The zero-order P1-TS fuzzy interpolator can be equivalently represented by a zero-order TS fuzzy system with input variables' fuzzy sets associated with the membership functions represented by polynomials of the first-order, and rule conclusions represented by the zero-order polynomials. The zero-order TS system corresponds also to a fuzzy singleton-type reasoning method [32].

The new contribution of this paper and its relation to the previous author’s papers are as follows. The fuzzy model based design of scheduling control scheme proposed in [27, 28] is in this paper presented in more feasibly form through applying the analytical theory and fundamental matrix recursion method proposed in [29–31] for fuzzy rule-based system interpretation, that results in reduction of the computational complexity of multidimensional scheduling control scheme.
synthesis. Thus, incorporating the advantages of PI-TS theory into interval arithmetic and local pole placement based synthesis of scheduling control scheme, the iterative procedure is developed for multi-input fuzzy scheduler design. In a case study, the usefulness of this procedure is verified and compared for two different assumptions for a desired closed-loop poles interval, in the experiments carried out on a laboratory scaled overhead crane.

The paper is organized as follows. Section two describes a PI-TS fuzzy scheduling control scheme for a LPV discrete-time control system. The iterative synthesis of fuzzy scheduling control system using local pole placement at selected points determined based on the interval analysis of closed-loop system polynomial coefficients is proposed in the form of iterative procedure in section three. Section four presents the case study, which confirms the usability of the proposed method illustrated by experimental results obtained on a laboratory scaled overhead crane. Section five delivers the final conclusions.

2. PI-TS fuzzy scheduling control scheme

Consider the closed-loop control system (Fig. 1) represented by the discrete-time transfer functions of a plant (1) and controller (2) with parameters varying in relation to the vector of exogenous variables $w = [w_1, w_2, ..., w_r]^T$.

$$G_O(z, w) = \frac{b_{n-1}(w)z^{n-1} + b_{n-2}(w)z^{n-2} + ... + b_0(w)}{z^n + a_{n-1}(w)z^{n-1} + ... + a_0(w)},$$

(1)

$$G_C(z, w) = \frac{d_{m-1}(w)z^{m-1} + d_{m-2}(w)z^{m-2} + ... + d_0(w)}{z^m + c_{m-1}(w)z^{m-1} + ... + c_0(w)}.$$  

(2)

\begin{figure} 
\centering 
\includegraphics[width=\textwidth]{fig1.png} 
\caption{Discrete-time closed-loop control system}
\end{figure}

The interval plant operating conditions vary within the bounded intervals of measurable variables $w \in [w_i] = [w_{i,-}, w_{i,+}]$, $i = 1, 2, ..., r$,

\begin{equation}
(3)
\end{equation}

thus, dividing each interval $[w_i]$ into $n_i$ subintervals $[\alpha_{i,j}, \alpha_{i,j+1}]$ (where $\alpha_{i,j} < \alpha_{i,j+1}$, and $j = 1, 2, ..., n_i$) the uncertainty of the model’s parameters and controller coefficients approximation can be based on a zero-order MIMO PI-TS fuzzy system with linear membership functions $N_i,j(w_i)$ and $P_i,j(w_i)$ (Fig. 2) defined as follows:

$$N_i,j(w_i) = \frac{\alpha_{i,j+1} - w_i}{\alpha_{i,j+1} - \alpha_{i,j}},$$

$$P_i,j(w_i) = 1 - N_i,j(w_i).$$

(4)

\begin{figure} 
\centering 
\includegraphics[width=\textwidth]{fig2.png} 
\caption{Linear membership functions specified for the interval $[\alpha_{i,j}, \alpha_{i,j+1}]$ of variable $w_i$}
\end{figure}

The further detailed explanation of PI-TS system theory can be found in [30]. The crisp output vector of a zero-order PI-TS system is a vector containing the parameters of a plant model (1) and controller (2) given by

$$v(w) = g_i^T\Omega_1 Q_k,$$  

(5)

where $g$ and $\Omega$ are called generator vector and fundamental matrix, respectively, which can be determined recursively for $r$ inputs $w_1, w_2, ..., w_r$ according to (6) through assuming the initial generator $g_0 = 1$ and fundamental matrix $\Omega_0 = 1$:

$$g_0 = 1, \quad \Omega_0 = 1,$$

$$g_i = \begin{bmatrix} 1 \\ w_i \end{bmatrix} \otimes g_{i-1},$$

$$\Omega_i = \frac{1}{\alpha_{i,j+1} - \alpha_{i,j}} \begin{bmatrix} \alpha_{i,j+1} - \alpha_{i,j} \\ 1 \end{bmatrix} \otimes \Omega_{i-1},$$

(6)

for $i = 1, 2, ..., r$, where “$\otimes$” denotes the Kronecker product.

The matrix $Q_k$ (where $k = 1, 2, ..., n_1 n_2 ... n_r$) contains the parameters of a plant model and controller at $2^r$ nominal operating points corresponding to the bounds of intervals $[\alpha_{i,j}, \alpha_{i,j+1}]$ (where $i = 1, 2, ..., r$) specified for $r$ inputs:

$$Q_k = \begin{bmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} & ... & a_{1,m-1}^{(1)} \\ a_{0,1}^{(2)} & a_{1,2}^{(2)} & ... & a_{1,m-1}^{(2)} \\ ... & ... & ... & ... \\ a_{0,r}^{(2^r)} & a_{1,r}^{(2^r)} & ... & a_{1,m-1}^{(2^r)} \end{bmatrix}_k,$$  

(7)

The PI-TS system can be equivalently represented by a TS fuzzy system divided into $n_1 n_2 ... n_r$ subsystems, where the $k$th subsystem is represented by the rule base (8) with the rule’s consequents corresponding to the rows of matrix $Q_k$ consisting of closed-loop control system parameters determined for $2^r$ nominal operating points (9).

$$R_1 : \text{IF } w_1 \text{ is } N_1,j \text{ and } w_2 \text{ is } N_2,j \text{ and } ... \text{ and } w_r \text{ is } N_r,j \text{ then } v(w) = \begin{bmatrix} a_{0,1}^{(1)} & a_{1,1}^{(1)} & ... & a_{1,m-1}^{(1)} \end{bmatrix}_k,$$  

(8)

$$R_2 : \text{IF } w_1 \text{ is } P_1,j \text{ and } w_2 \text{ is } N_2,j \text{ and } ... \text{ and } w_r \text{ is } N_r,j \text{ then } v(w) = \begin{bmatrix} a_{0,2}^{(2)} & a_{1,2}^{(2)} & ... & a_{1,m-1}^{(2)} \end{bmatrix}_k,$$

$$\vdots$$

$$R_{2^r} : \text{IF } w_1 \text{ is } P_1,j \text{ and } w_2 \text{ is } P_2,j \text{ and } ... \text{ and } w_r \text{ is } P_r,j \text{ then } v(w) = \begin{bmatrix} a_{0,r}^{(2^r)} & a_{1,r}^{(2^r)} & ... & a_{1,m-1}^{(2^r)} \end{bmatrix}_k,$$
Diophantine equation:

based on the equations system (14) derived from the interval min
satisfies the expected performances if the condition
where
Hence, the
parameters (1) at the nominal operating points correspond-
resulting in the P1-TS system with \( n_t = 1 \) intervals, denoted
\( \alpha_{i,n_1}, \alpha_{i,n_1+1} \) = \( [w_i^-, w_i^+] \) specified for each input variable
\( w_i \) (\( i = 1, 2, ..., r \)). Assuming the desired intervals of stable poles (11) for the nominal operating points (9), and deriving
the controller’s parameters from (14), the P1-TS system is used to interpolate the model and controller parameters at a
given operating point \( w_i \in [w_i] \) to determine according Al-
gorithm 1 the number of nominal points \( \alpha_{i,j} (j = 2, ..., n_i) \)
lying between \( \alpha_{i,1} = w_i^- \) and \( \alpha_{i,n_1+1} = w_i^+ \). Incrementing
\( i \) from 1 to \( r \), the interval \( [w_i^-, w_i^+] \) of currently considered
input variable is divided into the \( n_t \) number of subintervals to obtain \( l = 1, 2, ..., n_l + 1 \) sample points. Starting to
increment \( l \) from \( n_l + 1 \), the current number of intervals \( [\alpha_{i,j}, \alpha_{i,j+1}] \), specified for \( w_i \) variable, is incremented \( n_l = n_l + 1 \),
and each sample point \( w_{l+1} \) is temporally considered as the
upper bound of the interval \( [\alpha_{i,n_i-1}, \alpha_{i,n_i}] \). The desired in-
tervals of closed-loop system characteristic polynomial coeffi-
cients (13) are calculated for nominal operating points (9)
associating with the considered sample point \( \alpha_{i,n_1} = w_{l+1}, \) and
the parameters of controller are derived from (14). The condi-
tion (15) is tested for the most hazardous operating points
(\( w_i \) ) (16) and (17) corresponding to the all possible combi-
nations of the intervals \( [\alpha_{i,j}, \alpha_{i,j+1}] \) midpoints determined
previously and currently considered inputs and the upper
and lower bounds of intervals specified for input variables
\( w_i \) (\( i = c + 1, c + 2, ..., r \), where \( c \) is the number of currently
considered input) which have been not considered yet:

\[
S_i r_i = mid([p_i])
\]

where \( S_i \) is an elementary matrix (see the Appendix).

Assuming the \( S(w) \) and \( r(w) \) consist of closed-loop system
parameters interpolated using a P1-TS system within
\( w_i \in [\alpha_{i,j}, \alpha_{i,j+1}] \) (where \( i = 1, 2, ..., r \)), the control scheme satisfies the expected performances if the condition

\[
S(w)r(w) \in [p_i],
\]

is not violated for at least one interval vector \([p_i]\) (where
\( s = 1, 2, ..., 2^r \)) associated with the rule which has been acti-

ted to interpolate the controller’s parameters \( r(w) \) with the
firing strength factor \( \tau_s(w) > 0 \).

The objective function (15) can be used in the iterative
procedure of designing the zero-order P1-TS fuzzy system.
The control scheme synthesis involves to identify the model’s
parameters (1) at the nominal operating points correspond-
ing to the bounds of the scheduling variables intervals (3)
resulting in the P1-TS system with \( n_t = 1 \) intervals, denoted
\( \alpha_{i,n_1}, \alpha_{i,n_1+1} \) = \( [w_i^-, w_i^+] \) specified for each input variable
\( w_i \) (\( i = 1, 2, ..., r \)). Assuming the desired intervals of stable poles (11) for the nominal operating points (9), and deriving
the controller’s parameters from (14), the P1-TS system is used to interpolate the model and controller parameters at a
given operating point \( w_i \in [w_i] \) to determine according Al-
gorithm 1 the number of nominal points \( \alpha_{i,j} (j = 2, ..., n_i) \)
lying between \( \alpha_{i,1} = w_i^- \) and \( \alpha_{i,n_1+1} = w_i^+ \). Incrementing
\( i \) from 1 to \( r \), the interval \( [w_i^-, w_i^+] \) of currently considered
input variable is divided into the \( n_t \) number of subintervals to obtain \( l = 1, 2, ..., n_l + 1 \) sample points. Starting to
increment \( l \) from \( n_l + 1 \), the current number of intervals \( [\alpha_{i,j}, \alpha_{i,j+1}] \), specified for \( w_i \) variable, is incremented \( n_l = n_l + 1 \),
and each sample point \( w_{l+1} \) is temporally considered as the
upper bound of the interval \( [\alpha_{i,n_i-1}, \alpha_{i,n_i}] \). The desired in-
tervals of closed-loop system characteristic polynomial coeffi-
cients (13) are calculated for nominal operating points (9)
associating with the considered sample point \( \alpha_{i,n_1} = w_{l+1}, \) and
the parameters of controller are derived from (14). The condi-
tion (15) is tested for the most hazardous operating points
(\( w_i \) ) (16) and (17) corresponding to the all possible combi-
nations of the intervals \( [\alpha_{i,j}, \alpha_{i,j+1}] \) midpoints determined
previously and currently considered inputs and the upper
and lower bounds of intervals specified for input variables
\( w_i \) (\( i = c + 1, c + 2, ..., r \), where \( c \) is the number of currently
considered input) which have been not considered yet:

\[
\frac{\{\alpha_{i,j+1}-\alpha_{i,j}\}}{2}, \quad \text{for } i=1, 2, ..., c-1,
\]

\[
\{\{\alpha_{i,j+1}-\alpha_{i,j}\}/2, \{\alpha_{i,j+1}\}, \text{ for } i=c, j=n_i-1 ,
\]

\[
\{\alpha_{i,j}, \{\alpha_{i,j+1}\}, \text{ for } i=c+1, c+2, ..., r, j=n_i
\]

(16)

\[
\frac{\{\alpha_{i,j}, \alpha_{i,j+1}\}}{2}, \text{ for } i=1, 2, ..., c-1, j=1, 2, ..., n_i
\]

\[
\{\alpha_{i,j-1}-\alpha_{i,j}\}/2, \text{ for } i=c, j=n_i-1 ,
\]

\[
\{\alpha_{i,j}, \alpha_{i,j+1}\}, \text{ for } i=c+1, c+2, ..., r, j=n_i
\]

(17)

If the condition (15) is satisfied for all operating points (16)
and (17) the next sample point \( \alpha_{i,n_1} = w_{l+1} \) is considered
as the upper bound of the interval \( [\alpha_{i,n_1-1}, \alpha_{i,n_1}] \). If
the condition (15) is violated, the upper bound of the interval
\( [\alpha_{i,n_1-1}, \alpha_{i,n_1}] \) is set to the sample point \( \alpha_{i,n_1} = w_{l+1} \)
which has been tested successfully in the previous iteration.
However, if the condition (15) is violated at the first sample
point \( l = 2 \), the number of sample points should be double
increased \( n_l = 2n_l \).

The P1-TS fuzzy design process is two-stage procedure,
which in the first step results in determining the nominal op-
erating points for which the parameters of a system’s model
should be identified. In the next step, the iterative procedure
(Algorithm 1) should be repeated for initial number of inter-
vals \( n_l \) determined in the previous step to validate a control
system with reduced interpolation errors.
4. Case study and experimental results

In this section the proposed method is addressed to the anti-sway crane control problem. Reduction of the vibration is a serious concern for industrial cranes which are extensively used for shifting goods in building sites, shipping yards, container terminals and many manufacturing segments [33–35]. The system under consideration is the laboratory scaled overhead traveling crane with lifting capacity of 150 kg and motion mechanisms driven by DC motors. The experimental setup is presented in Fig. 3. The measurement equipment is based mainly on the incremental encoders used in the open-loop identification experiments and control for sensing the crane position and speed, rope length and payload deviation. Also, the vision based measurement techniques implemented on the laboratory stand are detailed described in [36, 37].

The identification experiments were conducted using a PC with I/O board (control-measurement card) and Matlab/RTW software. The model of controlled system was identified using output error (OE) method with sample time $T_s = 0.1$ s. The control scheme was designed in Matlab program and next the control algorithm was implemented using structured text (ST) on the RX3i programmable automation controller (PAC) and tested on the laboratory object. The control system (Fig. 4) developed and tested on the laboratory stand was based on the linear controllers used in the feedbacks of crane position and speed, and sensorless feedback of payload deviation estimated by the pendulum model assumed in the form of discrete-time transfer function (19). The P1-TS fuzzy system was used to approximate the parameters of controllers and pendulum mod-
P1-TS fuzzy scheduling control system design using local pole placement and interval analysis

Fig. 5. Planar model of a crane, where \( m, l, u \) and \( \phi \) are, respectively, mass of a payload, rope length, controlling signal corresponding to control force acting on a crane, and sway angle of a payload.

The closed-loop control scheme (Fig. 6) is assumed as a set of linear controllers for crane position, speed and first-order discrete-time controller of payload sway angle with parameters denoted \( k_1, k_2, q_0, q_1 \) and \( s_0 \). Hence, the transfer function of a closed-loop control system can be presented in the following form:

\[
\frac{\phi(z, l, m)}{X_r(z)} = \frac{k_1(l, m)k_2(l, m)d_0(l, m)\left(\frac{b_1(l, m)z^2}{z^5 + zS(l, m)r(l, m)}\right)}{+ b_0(l, m)s_0 + b_0(l, m)s_0}.
\]

where

\[
S = \begin{bmatrix}
    \begin{pmatrix}
        a_1 + c_0 - 1 \\
        c_0(a_1 + 1) \\
        d_0(a_1 + 1) \\
        0 \\
        0 \\
        0 \\
        1 \\
    \end{pmatrix}
    & \begin{pmatrix}
        a_0 - a_1 \\
        c_0(a_0 - a_1) \\
        d_0(a_0 - a_1) \\
        -a_0 \\
        -d_0a_0 \\
        -d_0b_1 \\
        -a_0c_0 \\
    \end{pmatrix}
    & \begin{pmatrix}
        -a_0c_0 \\
        -d_0a_0 \\
        -d_0b_1 \\
        -a_0b_0 \\
        -d_0b_1 \\
        -d_0b_0 \\
        -a_0b_0 \\
    \end{pmatrix}
    & \begin{pmatrix}
        0 \\
        0 \\
        0 \\
        0 \\
        0 \\
        0 \\
        0 \\
    \end{pmatrix}
\end{bmatrix}^T
\]

\[
z = [z^4, z^3, z^2, z^1, z^0],
\]

\[
r = [1, k_2, q_0, q_1, s_0, k_1k_2, k_2s_0, k_1k_2s_0]^T.
\]

Fig. 6. Discrete-time closed-loop control system
The P1-TS fuzzy system used to approximate the closed-loop system’s parameters based on the two scheduling variables \( w_1 = l \) and \( w_2 = m \) was designed through applying the iterative algorithm described in section 3. The objective of closed-loop system synthesis was to select the number and width of subintervals \([\alpha_{i,j}, \alpha_{i,j+1}]\) \((i = 1, 2)\) within the scheduling variables ranges \( l \in [1.0, 2.2] \) m and \( m \in [10, 90] \) kg, which satisfy the condition (15) for desired regions of closed-loop poles (21) determined in experiments carried out for \( \xi_y = 0.25 \), the condition (15) was satisfied at each considered sample point of intervals \([w_1]\) and \([w_2]\), that resulted in designing the P1-TS fuzzy interpolator with one interval specified for each input variable \([\alpha_{1,1}, \alpha_{1,2}] = [1.0, 2.2] \) m and \([\alpha_{2,1}, \alpha_{2,2}] = [10, 90] \) kg, respectively (Fig. 7).

The control algorithm was implemented using structured text on the PAC system and tested on the laboratory object for operating points lying within the scheduling variables intervals \([\alpha_{i,j}, \alpha_{i,j+1}]\). The examples of experiments conducted for P1-TS fuzzy system interpolating the closed-loop system parameters within intervals \([\alpha_{1,1}, \alpha_{1,2}] = [1.0, 1.6] \) m, \([\alpha_{1,2}, \alpha_{1,3}] = [1.6, 2.2] \) m and \([\alpha_{2,1}, \alpha_{2,2}] = [10, 90] \) kg (Fig. 7a) are presented in the form of unit-step responses of a system (Figs. 8, 9), where the payload deflection measured using incremental encoder (solid line) is compared with the signal estimated by the model (19) and used as a feedback in the sensorless control system (Fig. 4). The similarity of the object and model responses in the transient states confirms assumptions which were applied for object modeling.

The iterative procedure (Algorithm 1) implemented in Matlab program was employed to design P1-TS fuzzy interpolator for desired regions of closed-loop poles (21) determined for the parameters \( \xi_\alpha = 0.1 \) and \( \xi_y = 0.25 \). In both experiments the scheduling variables \([w_1]\) and \([w_2]\) were divided into 12 and 8 subintervals, respectively. The first experiment, carried out for \( \xi_\alpha = 0.1 \), resulted in determining the new nominal point \( l = 1.6 \) m required to satisfy the condition (15) for the considered interval system. Thus, after identification of model’s parameters at operating points \( [1.6, 10] \) kg and \( [1.6, 90] \) kg, the P1-TS fuzzy interpolator was designed with intervals \([\alpha_{1,1}, \alpha_{1,2}] = [1.0, 1.6] \) m, \([\alpha_{1,2}, \alpha_{1,3}] = [1.6, 2.2] \) m and \([\alpha_{2,1}, \alpha_{2,2}] = [10, 90] \) kg specified for input variable \( w_1 = l \) and \( w_2 = m \) respectively. In the second experiment, conducted for \( \xi_y = 0.25 \), the condition (15) was satisfied at each considered sample point of intervals \([w_1]\) and \([w_2]\), that resulted in designing the P1-TS fuzzy interpolator with one interval specified for each input variable \([\alpha_{1,1}, \alpha_{1,2}] = [1.0, 2.2] \) m and \([\alpha_{2,1}, \alpha_{2,2}] = [10, 90] \) kg, respectively (Fig. 7).

The objective of the control was positioning a crane and reducing the payload deflection with tolerance \( \pm 0.02 \) m. Figure 8 presents the results of experiments carried out for \( l = [1.0, 1.6, 2.2] \) m and \( m = 50 \) kg, while Fig. 9 depicts results obtained for \( l = [1.3, 1.9] \) and \( m = 50 \) kg. For operating points lying within intervals \([1.0, 2.2] \) m and \([10, 90] \) kg, in the experiments realized with using P1-TS fuzzy interpolator designed for \( \xi_\alpha = 0.1 \) (Fig. 7a), the settle time, was in the interval \([5.5, 6.7] \) s, while for control system designed for \( \xi_y = 0.25 \) the settle time was \([5.5, 7.5] \) s, that is illustrated.
P1-TS fuzzy scheduling control system design using local pole placement and interval analysis

Fig. 8. Crane position and payload deflection – experiments at operating points \( l = \{1.0, 1.6, 2.2\} \) m and \( m = 50 \) kg

in Fig. 10 presenting the comparison of control system performances at operating point \( \{1.6 \text{ m}, 50 \text{ kg}\} \). Interpolation within intervals \([1.0, 1.6] \) m and \([1.6, 2.2] \) m (Fig. 7a) results in settle time about 6.1 s, while using the P1-TS system designed for \( \xi_y = 0.25 \) (Fig. 7b) it was 7.5 s, however the payload deviation is reduced to the expected range ±0.02 m about 2 seconds later.

Fig. 9. Crane position and payload deflection – experiments at operating points \( l = \{1.3, 1.9\} \) m and \( m = 50 \) kg

The interval of acceptable deviation from a nominal operating point determines the width of intervals \([\alpha_i, \alpha_{i+1}]\) and interpolation errors, that obviously has influence on the closed-loop system performances deterioration especially at the midpoints of the intervals \([\alpha_i, \alpha_{i+1}]\). Figure 11 presents the relative errors between natural not dumped pulsation of the pendulum model (19), identified in the series of experiments carried out at sample points specified within the interval \([w_1] = [1.0, 2.2] \) m and for the midpoint of the interval \( \text{mid}(w_2) = 50 \) kg, and the natural pulsation of a model with parameters interpolated using the P1-TS fuzzy system. The maximum relative errors correspond to the midpoints of intervals specified for input variables of fuzzy interpolator. In the considered example, the midpoint of the interval \([1.0, 2.2] \) m simultaneously corresponds to the nominal point determined by the iterative algorithm for \( \xi_x = 0.1 \). Thus, identification of model’s parameters at operating points \( \{1.6 \text{ m}, 10 \text{ kg}\} \) and \( \{1.6 \text{ m}, 90 \text{ kg}\} \) results in decreasing the interpolation errors, and verify the condition (15) for the midpoints of intervals \([1.0, 1.6] \) and \([1.6, 2.2] \) m, at which the closed-loop
system polynomial coefficients are close to the bounds of a desired interval determined for $\xi_k = 0.1$.

Fig. 10. Crane position and payload deflection – comparison of experiments carried out at operating point $l = 1.6 \text{ m}$ and $m = 50 \text{ kg}$ with P1-TS system interpolating the system’s parameters within the intervals [1.0, 1.6] m and [1.6, 2.2] m, and within the interval [1.0, 2.2] m

![Crane position and payload deflection comparison](image)

Fig. 11. Interpolation errors for P1-TS systems used to interpolate the model’s parameters within intervals [1.0, 1.6] m and [1.6, 2.2] m (dotted line), and between the interval [1.0, 2.2] m (solid line)

![Interpolation errors comparison](image)

5. Conclusions

The effectiveness of fuzzy scheduling control schemes is confirmed in a wide range of applications described in the literature. However, the less attention is focused on the problem of selecting the appropriate number of intervals within the scheduling variable ranges, especially for the general problem of multi input fuzzy interpolator design. In this paper, this problem is addressed using the LPV discrete-time model based iterative procedure based on the local pole placement and interval analysis of closed-loop system polynomial coefficients. The reduction of computational complexity of a fuzzy scheduler design and improvement of hardware implementation feasibility can be obtained utilizing the P1-TS theory-based fundamental matrix method and a recursive procedure used for fuzzy rule-based interpretation. The usability of the proposed method was verified on the laboratory scaled overhead crane. The fuzzy interpolator with two input variables was designed for two different assumptions for a desired region of closed-loop stable poles. The control algorithm was implemented using ST on the PAC system and successfully tested on the laboratory stand. The future challenge is the implementation of this method for design the control system of a large scale material handling device.

Appendix

Considering, that the coefficients of a desired characteristic polynomial of the closed-loop system (Fig. 1) are the nominal points of an interval vector (13), the Diophantine equation

\[
(z^n + a_{n-1}z^{n-1} + \ldots + a_0)(z^m + c_{m-1}z^{m-1} + \ldots + c_0) + (b_{n-1}z^{n-1} + b_{n-2}z^{n-2} + \ldots + b_0)(d_{m-1}z^{m-1} + d_{m-2}z^{m-2} + \ldots + d_0) = z \cdot \text{mid}(p) \mid p
\]

(22)

can be presented as follows

\[
\begin{align*}
\begin{bmatrix}
1 & 0 & \cdots & 0 & 0 & 0 \\
0 & a_{n-1} & 1 & \cdots & 0 & 0 \\
a_n & a_{n-2} & c_{m-1} & \cdots & \cdots & \cdots \\
0 & 0 & \cdots & a_0 & a_1 & a_2 \\
0 & 0 & \cdots & 0 & a_0 & a_1 \\
0 & 0 & \cdots & 0 & 0 & a_0 \\
\end{bmatrix}
&= z \cdot \text{mid}(p) \\
\begin{bmatrix}
b_{n-1} & 0 & \cdots & 0 & 0 & 0 \\
0 & \cdots & a_{n-1} & b_{n-1} & 0 & 0 \\
a_{n-2} & a_{n-1} & b_{n-1} & \cdots & 0 & 0 \\
0 & \cdots & 0 & b_0 & b_1 & b_2 \\
0 & 0 & \cdots & 0 & b_0 & b_1 \\
0 & 0 & \cdots & 0 & 0 & b_0 \\
\end{bmatrix}
&= z \cdot \text{mid}(p)
\end{align*}
\]

(23)
where \( z = [z^{n+m}, z^{n+m-1}, ..., 1] \). Hence, the equations system (14) is given
\[
\begin{bmatrix}
1 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
a_{n-1} & 1 & \cdots & 0 & 0 & 0 & \cdots & 0 & 0 \\
a_{n-2} & a_{n-1} & \cdots & 0 & 0 & b_{n-1} & 0 & \cdots & 0 & 0 \\
a_{n-3} & a_{n-2} & \cdots & 0 & 0 & b_{n-2} & b_{n-1} & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & a_1 & a_2 & 0 & 0 & \cdots & b_1 & b_2 \\
0 & 0 & \cdots & a_0 & a_1 & 0 & 0 & \cdots & b_0 & b_1 \\
0 & 0 & \cdots & 0 & a_0 & 0 & 0 & \cdots & 0 & b_0
\end{bmatrix}
\]
Removing the first equation (1 = 1), the system of Eqs. (24) can be rewritten as
\[
\begin{bmatrix}
1 & 0 & \cdots & 0 & 0 & 0 & 0 & \cdots & 0 & 0 \\
a_{n-1} & 0 & \cdots & 0 & 0 & b_{n-1} & 0 & \cdots & 0 & 0 \\
a_{n-2} & 0 & \cdots & 0 & 0 & b_{n-2} & b_{n-1} & \cdots & 0 & 0 \\
\vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\
0 & \cdots & a_1 & a_2 & 0 & 0 & \cdots & b_1 & b_2 \\
0 & \cdots & a_0 & a_1 & 0 & 0 & \cdots & b_0 & b_1 \\
0 & 0 & \cdots & 0 & a_0 & 0 & 0 & \cdots & 0 & b_0
\end{bmatrix}
\]
\[
\begin{bmatrix}
c_{m-1} \\
\vdots \\
c_1 \\
c_0 \\
d_{m-1} \\
d_{m-2} \\
d_1 \\
d_0
\end{bmatrix}
= \text{mid}
\begin{bmatrix}
1 \\
[p_n+m-1] \\
[p_n+m-2] \\
[p_n+m-3] \\
[p_n+m-4] \\
[p_2] \\
[p_1] \\
[p_0]
\end{bmatrix}
\]
(24)

\[
\begin{bmatrix}
c_{m-1} \\
\vdots \\
c_1 \\
c_0 \\
d_{m-1} \\
d_{m-2} \\
d_1 \\
d_0
\end{bmatrix}
= \text{mid}
\begin{bmatrix}
[p_n+m-1] - a_{n-1} \\
[p_n+m-2] - a_{n-2} \\
[p_n+m-3] - a_{n-3} \\
[p_n+m-4] \\
[p_2] \\
[p_1] \\
[p_0]
\end{bmatrix}
\]
(25)

REFERENCES

techniques”, *Int. Conf. on Computational Intelligence for Modeling, Control and Automation* 1, CD-ROM (2006).


