Extended TP Model Transformation for Polytopic Representation of Impedance Model with Feedback Delay

PÉTER GALAMBOS [1,2], PÉTER BARANYI [1,3] and PÉTER KORONDI[1,4]
[1] Computer and Automation Research Institute, Hungarian Academy of Sciences, Cognitive Informatics Group
[2] Budapest University of Technology and Economics, Dept. of Manufacturing Science and Technology
[3] Budapest University of Technology and Economics, Dept. of Telecommunications and Media Informatics
[4] Budapest University of Technology and Economics, Dept. of Mechatronics, Optics and Engineering Informatics
Kenede u. 13-17, H-1111 Budapest, HUNGARY
{galambos, baranyi, korondi}@sztaki.hu

Abstract: In force controlled industrial robots, haptic and telemanipulation devices stability and control performance are contradicting requirements. In distributed systems where the sensors, actuators and control logic are separated in space, due to the network delays the control process may become unstable. In our research, we consider the coupled impedance type control algorithms as common used method in telemanipulation and interaction control of robots. In this paper, an extended TP model transformation is proposed to convert the delayed system into a polytopic tensor product (TP) model considering the value of the feedback delay as a parameter. Using such model, controller design become tractable as it does not contain feedback delay. A numerical example for a single degree of freedom impedance model with feedback delay is discussed. The results are confirmed by simulation.

Key–Words: Delayed systems, Feedback systems, Interaction control, Telemanipulation, Haptics, TP Model Transformation, qLPV model, Polytopic linear models

1 Introduction

Time delays are inherent attributes of feedback systems that usually have unfavorable effects on the performance of the controlled process. Distributed robot control and internet based teleoperation are typical examples where communication delay play crucial role [1, 2], but similar problem arises in haptic devices as well [3]. In this paper, telemanipulation is emphasized as a possible field of application but the results stand for impedance control generally. The key problem is that the transparency and stability of teleoperation can not be guaranteed at the same time under time delay. Transparency and stability of bilateral teleoperation was studied by Lawrence [4]. In the past decades several approaches were published addressing the stability problem of closed loop force reflecting telemanipulation. A comprehensive survey can be found in [5]. Common problem of the published approaches that however the stability is guaranteed even under unknown varying delay, the operator loses the realistic force sensation.

The class of compliance model based force reflecting algorithms [6, 7, 8, 9] is regarded in this paper. Compliance method is also common in robot interaction control [10]. Figure 1 illustrates the operation of such algorithms in teleoperation scenario. The goal of this research is to develop a model based

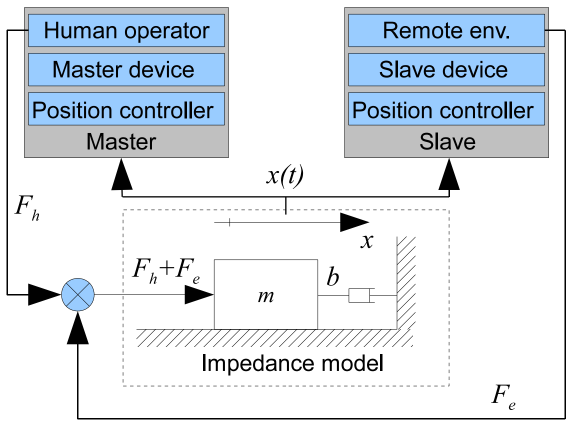

Figure 1: Scheme of the coupled impedance force reflecting algorithm

control scheme wherein the feedback gains are designed according to a non-delayed model, but the control signal acts in the original delayed system. As first step, the delayed system must be substituted with a non-delayed model as powerful control design tools are available for systems without delay. This paper introduces the extended tensor product (TP) model transformation to convert a system with feedback delay into a non-delayed quasi-linear parameter varying (qLPV) model where the actual value of the delay $\tau(t)$

becomes a parameter of the model. Proposed method results a qLPV state space model in finite element TP type polytopic form.

The paper is structured as follows: Section 2 defines the terms and notations. Section 3 gives the definitions of the conceptual basics are involved throughout this paper. The process of TP model transformation is discussed step by step in the section 4. Section 5 introduces an extension of TP model transformation for system models where the nonlinear terms are not known in analytical form. Section 6 goes trough the application of extended TP model transformation via a simple mass-damper compliance model with feedback delay: Reveal the effect of feedback delay, shows the process of extended TP model transformation on a numerical example, apply complexity-accuracy tradeoff on the resulted TP model and finally, validate the results by numerical simulation. Section 7 concludes the paper.

2 Nomenclature

F_h	Interaction force/torque with the operator $[N]$
F_e	interaction force/torque with the remote environment $[N]$
x	position of the impedance model $[m]$
m	mass $[kg]$
k	stiffness $[N/m]$
b	viscous damping $[Ns/m]$
τ	delay $[s]$
M	discretization grid
$a, b, \ldots$	scalar values
$\mathbf{a}, \mathbf{b}, \ldots$	vectors
$A, B, \ldots$	matrices
$\mathbf{A}, \mathbf{B}, \ldots$	tensors
$A \times_n \mathbf{U}$	n-mode product of a tensor by a matrix
$A \underset{n=1}{\overset{N}{\boxtimes}} \mathbf{U}_n$	multiple n-mode product as $A \times_1 \mathbf{U}_1 \times_2 \mathbf{U}_2 \cdots \times_N \mathbf{U}_N$

3 Basic concepts

The following definition are used in this paper:

Definition 1 *(qLPV model): Consider the Linear Parameter Varying State Space model:*

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}(\mathbf{p}(t))\mathbf{x}(t) + \mathbf{B}(\mathbf{p}(t))\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}(\mathbf{p}(t))\mathbf{x}(t) + \mathbf{D}(\mathbf{p}(t))\mathbf{u}(t),\end{aligned} \tag{1}$$

with input $\mathbf{u}(t)$, output $\mathbf{y}(t)$ and state vector $\mathbf{x}(t)$. The system matrix

$$\mathbf{S}(\mathbf{p}(t)) = \begin{pmatrix} \mathbf{A}(\mathbf{p}(t)) & \mathbf{B}(\mathbf{p}(t)) \\ \mathbf{C}(\mathbf{p}(t)) & \mathbf{D}(\mathbf{p}(t)) \end{pmatrix} \tag{2}$$

is a parameter-varying object, where $\mathbf{p}(t) \in \Omega$ is time varying $N-$dimensional parameter vector, where $\Omega = [a_1, b_1] \times [a_2, b_2] \times .. \times [a_N, b_N] \subset \mathbb{R}^N$ is a closed hypercube. $\mathbf{p}(t)$ can also include some elements of $\mathbf{x}(t)$, in this case (2) is termed as quasi LPV (qLPV) model. Therefore this type of model is considered to belong to the class of non-linear models. Assume that the size of the system matrix $\mathbf{S}(\mathbf{p}(t))$ is O times I.

Definition 2 *(Finite element polytopic model):*

$$\mathbf{S}(\mathbf{p}(t)) = \sum_{r=1}^{R} w_r(\mathbf{p}(t))\mathbf{S}_r \tag{3}$$

where $\mathbf{p}(t) \in \Omega$. $\mathbf{S}(\mathbf{p}(t))$ is given for any parameter vector $\mathbf{p}(t)$ as the parameter varying combinations of LTI system matrices $\mathbf{S}_r \in \mathbb{R}^{(m+k)\times(m+k)}$ called LTI vertex systems. The combination is defined by the weighting functions $w_r(\mathbf{p}(t)) \in [0, 1]$. By finite we mean that R is bounded.

Definition 3 *(Finite element TP type polytopic model): We say TP model for the sake of brevity. $\mathbf{S}(\mathbf{p}(t))$ in (3) is given for any parameter as the parameter-varying combination of LTI system matrices $\mathbf{S}_r \in \mathbb{R}^{(m+k)\times(m+k)}$.*

$$\mathbf{S}(\mathbf{p}(t)) = \sum_{i_1=1}^{I_1}\sum_{i_2=1}^{I_2} .. \sum_{i_N=1}^{I_N} w_{n,i_n}(p_n(t))\mathbf{S}_{i_1,i_2,..,i_N}, \tag{4}$$

applying the compact notation based on tensor algebra (Lathauwer's work [11] we have:

$$\mathbf{S}(\mathbf{p}(t)) = S \underset{n=1}{\overset{N}{\boxtimes}} \mathbf{w}\left(p_n\left(t\right)\right) \tag{5}$$

where the (N+2) dimensional coefficient tensor $\mathbf{S} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n \times (m+k) \times (m+k)}$ is constructed from the LTI vertex systems $\mathbf{S}_{i_1,i_2,..,i_N}$ (5) and the row vector $w_n\left(p_n\left(t\right)\right) \in [0, 1]$ contains one variable and continuous weighting functions $w_{n,i_n}(p_n(t))$,$(i_n = 1 \ldots I_N)$.

Remark 4 *: TP model (5) is a special class of polytopic models (2), where the weighting functions are decomposed to the Tensor Product of one variable functions.*

Definition 5 *The TP model is convex if the weighting functions satisfy the following criteria:*

$$\forall n, i, p_n(t) : w_{n,i}(p_n(t)) \in [0, 1]; \tag{6}$$

$$\forall n, p_n(t) : \sum_{i=1}^{I_n} w_{n,i}(p_n(t)) = 1. \tag{7}$$

We can define various types of convex TP models. These types can readily be determined via constraints defined for the weighting functions. Let us define the type of TP model which we use in this paper; the other possible types of TP models are discussed in [12]

Definition 6 *(NO/CNO, NOrmal type TP model): The convex TP model is a NO (normal) type model, if its w(p) weighting functions are Normal, that is, if it satisfies (6), (7) , and the largest value of all weighting functions is 1. Also, it is CNO (close to normal), if it is satisfies (6), (7) and the largest value of all weighting functions is 1 or close to 1.*

Remark 7 *: NO/CNO type convex TP model defines tight convex hull of a system [13].*

Definition 8 *(Exact / Non-exact TP model): A TP model is termed* Exact TP model*, if for all* $\mathbf{p}(t) \in \Omega$

$$\mathbf{S}(\mathbf{p}) = S \mathop{\boxtimes}_{n=1}^{N} \mathbf{w}_n(p_n) \tag{8}$$

A TP model is termed Non-Exact TP model *if:*

$$\hat{\mathbf{S}}(\mathbf{p}) = S \mathop{\boxtimes}_{n=1}^{N} \mathbf{w}_n(p_n) \tag{9}$$

holds, where $\hat{\mathbf{S}}(\mathbf{p})$ *is only an approximation of* $\mathbf{S}(\mathbf{p})$*, where the error* γ *is defined as:*

$$\max \|\mathbf{S}(\mathbf{p}) - \hat{\mathbf{S}}(\mathbf{p})\|_{\mathcal{L}_2} = \gamma \tag{10}$$

4 Steps of TP model transformation

In this section, the key phases of TP model transformation is introduced.

4.1 Discretization

The goal is to represent the a given parameter dependent system by a tensor that is ready to find the tensor product structure according to the parameter dependent system. First, we define the transformation space Ω in which we expect the TP model be relevant.

The transformation space Ω is a bounded hyper rectangular space where the parameter vector of the system matrix varies: $\mathbf{p}(t) \in \Omega : [a_1, b_1] \times [a_2, b_2] \times .. \times [a_N, b_N] \subset \mathbb{R}^N$. Ω can be arbitrarily defined, however the resulting TP model is interpretable only in Ω. In practice, Ω should be defined according to the working space os $\mathbf{p}$.

Ω is discretized according to a hyper rectangular discretization grid M. In general, de grid points can be arbitrarily located in the intervals, however equidistant grid is suggested.

S^D discretized system matrix represents the discretized qLPV model over the hyper rectangular grid M in Ω. The entries of S^D are

$$S^D_{m_1,m_2,...,m_N} = \mathbf{S}(\mathbf{p}) \tag{11}$$

where $\mathbf{p}$ denotes the parameter vector in Ω according to the grid points of M.

4.2 Extracting the TP structure

The goal of this step to reveal the TP structure of the given qLPV model and find the minimal number of LTI components and perform accuracy trade-off in case of searching a non-exact TP model. Higher Order Singular Value Decomposition (HOSVD) is used to find the TP structure of the model. HOSVD is a generalization of matrix SVD for higher order tensors. Introduction of HOSVD can be found in [11]. Applying HOSVD on S^D and discarding the zero singular values we get the form:

$$S^D = S \mathop{\boxtimes}_{n} \mathbf{U}_n \tag{12}$$

Where the size of S is $I_1 \times I_2 \times ... \times I_n \times (m+k) \times (m+l)$. $I_n = rank_n(S^D) \leq M_n \forall n = 1..N$. Here, $rank_n$ denotes the n-mode rank of a tensor [11]. Tensor S contains the LTI systems $\mathbf{S}_{i_1,i_2,...,i_N}$, $i_n = 1..I_n$.

If we would like further decrease the number of the LTI systems we can execute RHOSVD (Reduced Higher Order SVD). In RHOSVD, we discard non-zero singular values (not only the zero ones) and the corresponding singular vectors, then the decomposition results in an approximation of S^D. Discarding non-zero singular values, we can perform trade-off between the error and the complexity in terms of the number of LTI systems.

Having the HOSVD based canonical TP structure, the matrix transformation $\mathbf{T}$ can be defined to transform singular matrices $\mathbf{U}_n$ to $\mathbf{U}'_n$:

$$\mathbf{U}'_n = \mathbf{U}_n(T)_n \tag{13}$$

Then the LTI systems can be calculated as:

$$S' = S \mathop{\boxtimes}_{n} \mathbf{T}_n \tag{14}$$

If the transformation $\mathbf{T}_n$ is defined in a special way we obtain S' in such a way that the LTI systems defines different types of convex hull of the system. (e.g. CNO type convex hull is defined in Definition 6)

4.3 Determination of the weighting functions

The parameter dependent system matrix can be obtained by (5) if the weighting functions are known. The matrices $\mathbf{U}_n$ or $\mathbf{U}'_n$ defines the discretized weighting functions: The i_nth column vector $\mathbf{u}_{n,i_n}$ of matrix $\mathbf{U}_n \in \mathbb{R}^{M_n \times I_n}$ determines $\mathbf{w}_n^D$ that is the discretization of $w_{n,i_n}(p_n)(n = 1..N)$ over M. Continuous weighting functions can be calculated by simple linear interpolation between the discretized values.

Further details on TP model transformation can be found in [14, 15] and the TP Tool website [16]. In the past few years several studies have been published on the TP model based control design, some of them are referred here: Precup et al. investigated the stability of nonlinear Fuzzy control systems [17] and the possibilities of the combination of TP and fuzzy models for control system development [18]. Kolonic, Poljugan and Petrovic applied the TP model transformation for gantry crane control system [19].

5 Extended TP model transformation

The key idea is to apply the TP model transformation on a set of LTI state space models representing the delayed system on a discretized space of possible delay values. In the investigated problem, the larger feedback delay cause increased settling time while the system goes to instability. The nonlinear terms of system parameters characterizing the delay dependent behavior are not known in analytical form. The TP model transformation is extended with a preliminary step, in which the sampled set of parameter-dependent (delay dependent) system matrices are determined by black box identification.

5.1 Reformulation and discretization

The non-delayed system is represented by its LTI state space model:

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t)\end{aligned} \tag{15}$$

A system with internal delay can be described by the following system of delay-differential-algebraic equation. For the details see [20].

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{A}\mathbf{x}(t) + \mathbf{B_1}\mathbf{u}(t) + \mathbf{B}_2\mathbf{w}(t) \\ \mathbf{z}(t) &= \mathbf{C}_2\mathbf{x}(t) + \mathbf{D}_{21}\mathbf{u}(t) + \mathbf{D}_{22}\mathbf{w}(t) \\ \mathbf{w}(t) &= [z_1(t-\tau), ..., z_N(t-\tau)]^T \\ \mathbf{y}(t) &= \mathbf{C}_1\mathbf{x}(t) + \mathbf{D}_{11}\mathbf{u}(t) + \mathbf{D}_{12}\mathbf{w}(t)\end{aligned} \tag{16}$$

The delayed model can be approximated by a system without delay. The approximated system is similar to the original (delayed) system in term of input-response characteristics. The quasi equivalent system model is determined by black box identification. In this paper the output error model (17) is used, but many other methods are applicable. The coefficients of the polynomials are estimated using a prediction error/maximum likelihood method. For further details on the model and the algorithm see [21].

$$y(t) = \frac{B(q)}{F(q)}u(t-nk) + e(t) \tag{17}$$

The identified non-delayed model can be converted into continuous state space form. Executing the identification at each discrete τ value on an interval, the tensor S^D become ready in the same form as it is written in section 4.1.

5.2 The rest of the algorithm

Steps of TP model transformation can be a applied without modification, as it is presented in the subsections 4.2 and 4.3. The transformation results a qLPV state space model in finite element TP type polytopic form.

6 Application of extended TP model transformation for delayed impedance model

In this section, the extended TP model transformation will be applied on single degree of freedom impedance model with feedback delay. The content of this chapter is broken into three parts. Firstly, the feedback delay and its unfavorable effect is discussed in a typical impedance model that is commonly used in robotics. In the second subsection the extended TP model transformation is demonstrated on a numerical example and a compromise is made between the accuracy and the complexity of the model. In the third part, the resulted TP model is validated under constant and varying delays via numerical simulation.

6.1 Problem formulation

Consider the single degree of freedom mechanical system depicted by Figure 2 as a simplified model of impedance control. Mass m and viscous damping b are virtual properties defining the desired dynamics of the interaction. Stiffness k means the stiffness of the environment. In real case, the environment usually more complicated, but this simplified model is suitable for investigating the effect of time delay.

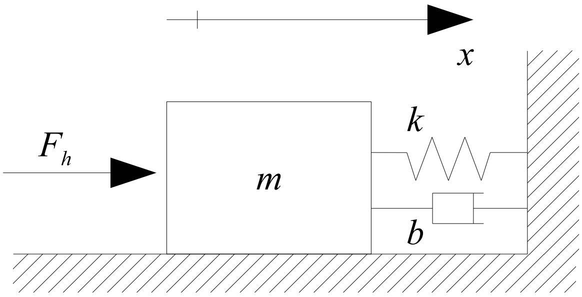

Figure 2: Mass-Spring-Damper system

In the scheme of coupled impedance based force reflecting algorithm (also referred as force-sum-driven internal virtual model) [6, 7] (Figure 1), the mass m and the viscous damping b are virtual parameters characterizing the dynamic behavior of the telemanipulator devices, while the stiffness k represents the remote environment. Virtual parameters are chosen according to the transparency $\leftrightarrow$ stability trade-off: The lower mass and damping values result the more transparent behavior and the less robustness against feedback delay. The equation of motion of this system is as follows:

$$\ddot{x}(t) = \frac{F_h(t)}{m} - \frac{b}{m}\dot{x}(t) - F_e(t) \tag{18}$$

Introducing the time delay τ in the remote interaction:

$$\ddot{x}(t) = \frac{F_h(t)}{m} - \frac{b}{m}\dot{x}(t) - F_e(t-\tau(t)) \tag{19}$$

The remote environment is modeled as a single ideal spring (linear stiffness). Substituting the k stiffness into the formula the equation of motion is:

$$\ddot{x}(t) = \frac{F_h(t)}{m} - \frac{b}{m}\dot{x}(t) - \frac{k}{m}x(t-\tau(t)). \tag{20}$$

One can see that the resulted equation represents a mass-spring-damper system where the effect of the spring is delayed by $\tau(t)$. Figure 3 illustrates the resulted delayed model. The network delay and the remote environment block contains the overall communication delay, the actuator dynamics , and the environment dynamics. In the discussed example, the dynamics of the actuator is neglected and the remote environment is considered as a linear stiffness (k).

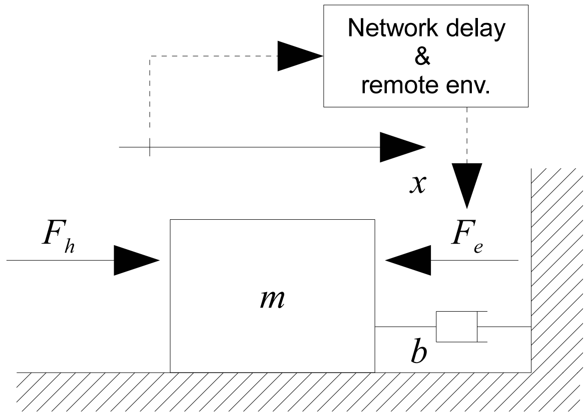

Figure 3: Impedance model with feedback delay

Figure 4 shows the effect of τ feedback delay: By the increasing delay, the system getting more susceptible for oscillation. In other words the so-called damping coefficient become smaller and the system become unstable. Over a certain value of feedback delay the system become unstable.

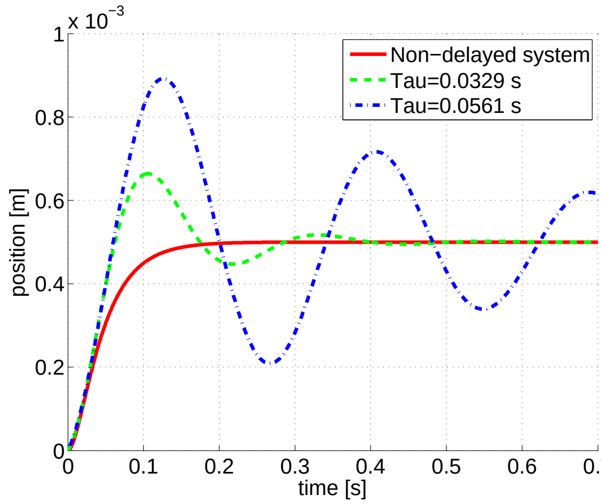

Figure 4: Step response of the model with various delay values

6.2 Transformation and Complexity Trade-off

The model characterized by equation (20) can be converted into an LTI state space model with internal delay as described in the section 5.1. To reach this form, the stiffness term must be treated separately and connected to the mass-damper part in a negative feedback term with input delay.

Going through the steps of the extended TP model transformation, the polytopic form of the delay dependent model results as:

$$\mathbf{S}(\tau) = \sum_{r=1}^{R} w_r(\tau)\mathbf{S}_r \tag{21}$$

where R equals the rank of the S^D coming from the HOSVD.

A numerical example is presented using the following parameter set: $m = 1kg$, $k = 2000N/m$, $b = 100Ns/m$. The non-delayed systems poles are $p_1 = -27.64$, $p_2 = -72.36$ so the system is overdamped. The one dimensional discretization grid M is defined over the interval $\tau = [0..0.7]$ contains 137 equidistant points.

The non-delayed system can be represented by its linear state space model:

$$\begin{aligned} \dot{\mathbf{x}}(t) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{aligned} \tag{22}$$

Where the elements in this concrete example are the follows according to the eq.(20) if $\tau = 0$:

$$\mathbf{x}(t) = \begin{bmatrix} \dot{x} \\ x \end{bmatrix} \qquad \mathbf{u}(t) = F_h(t)$$

$$\mathbf{A} = \begin{bmatrix} -100 & -2000 \\ 1 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix} \qquad \mathbf{D} = 0$$

The state space system changes if delay $\tau > 0$ introduced in the stiffness term of eq.(20). Due to the black box identification, the state variables lose their original physical meaning. As example, the identified state space system at $\tau = 0.252$:

$$\mathbf{A} = \begin{bmatrix} -983.9 & 1016 \\ -984.5 & -1015 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0.001805 \\ -0.001249 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad \mathbf{D} = 0$$

If $\tau = 0.51$:

$$\mathbf{A} = \begin{bmatrix} 996.6 & 1003 \\ -996.9 & -1003 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 0.00356 \\ -0.003279 \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \qquad \mathbf{D} = 0$$

In the 6.1 it is shown, how the system behavior is changing due to the feedback delay. After the identification, the time constants of the resulted systems can be analyzed. Figure 5 illustrates the excursion of the poles in the function of τ delay. The pole trajectories are parameterized by τ that is increasing in the direction of the arrows. Initially, the poles getting closer to each other on the real axis then become complex conjugates with decreasing real parts until reach the imaginary axis and become unstable. This behavior is consistent with the step responses shown in Figure 4.

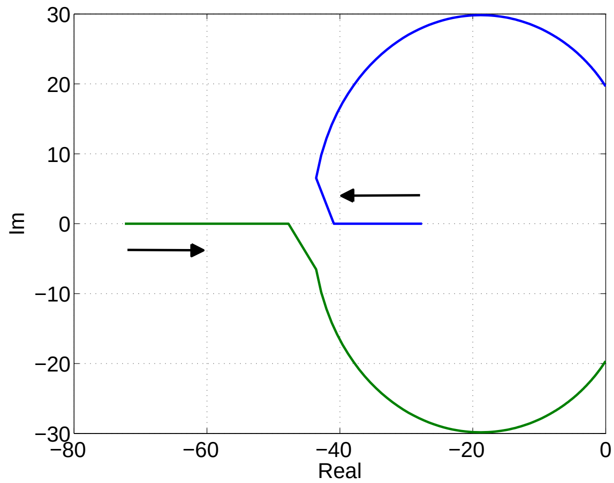

Figure 5: Displacement of the poles with increasing τ

By the identification the system is sampled over the one dimensional discretization grid M in 137 equidistant points. The resulted systems are arranged in the following matrix form:

$$\mathbf{S}_i^D = [\mathbf{A}_i \mid \mathbf{B}_i] \in \mathbb{R}^{2\times 3} \tag{23}$$

Where $i \in [1..137]$. Matrices $\mathbf{C}$ and $\mathbf{D}$ are omitted as they are constants.

Executing the HOSVD on the tensor of discretized LTI systems, the number of resulted singular values (rank) was 6. Our aim is to apply LMI based control design method on the polytopic model. NO/CNO type TP model is generated as it defines tight convex hull of the qLPV system that is suitable for LMIs [22]. For this reason, CNO type models are discussed.

If all nonzero singular values are used, the polytopic model is exact. The singular values are 2.3414×10^4, 349.52, 0.4567, 0.0025, 6.2041×10^{-4} and 7.7044×10^{-7} in decreasing order. Figure 6 shows the CNO weighting functions of the exact model.

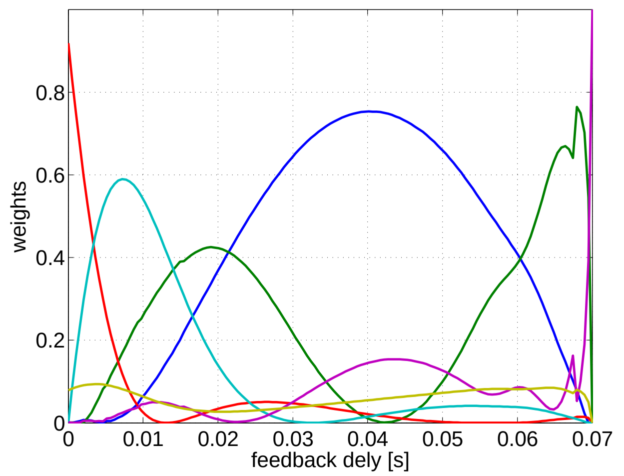

Figure 6: CNO weighting functions of the exact TP model

The relatively large number of vertex systems (LTI system) means gratuitous complexity and so large computational cost in the course of the control design. It is reasonable to make trade-off between the accuracy and the complexity of the model. Of course, after the controller design the closed-loop system must be verified carefully, because it is not trivial to judge if the simplification was permissible. A non-exact TP model has been created by discarding the 3 smallest nonzero singular values. Figure 7 displays the weighting functions of the non-exact model.

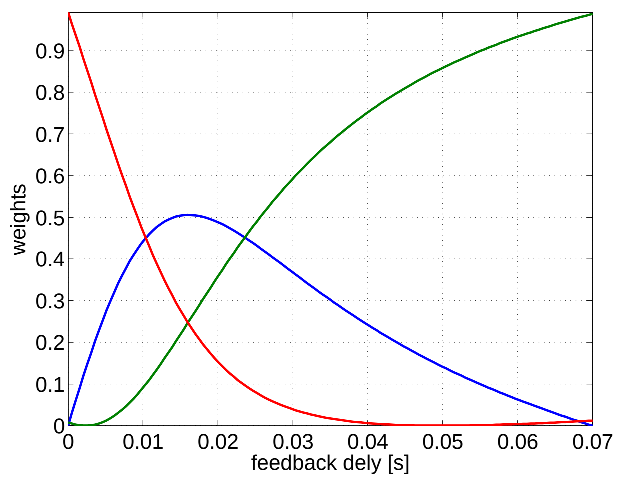

Figure 7: CNO weighting functions of the non-exact model keeping the three largest singular values

The resulted vertex LTI system are the followings:

$$\mathbf{S_1} = \begin{bmatrix} 971.3454 & 1028.5 & -4\times 10^{-4} \\ -972.2555 & -1027.5 & 1.3\times 10^{-3} \end{bmatrix}$$

$$\mathbf{S_2} = \begin{bmatrix} 1000.5 & 999.47 & 4.1\times 10^{-3} \\ -1000.6 & -999.28 & -3.9\times 10^{-3} \end{bmatrix}$$

$$\mathbf{S_3} = \begin{bmatrix} 949.7911 & 1051.6 & 0.08\times 10^{-3} \\ -950.7800 & -1050.6 & 0.87\times 10^{-3} \end{bmatrix}$$

The exact and non-exact models are compared in term of the error expression defined in 10. The numbers should be treated with circumspection because the effect of differences of γ is not traceable in the time domain behavior of the TP model. Table 1 contains γ for the exact (6sv) and the non-exact(6sv) models. The difference is smaller than one order of magnitude, so the non-exact model can be considered as a good approximation.

Table 1: Accuracy comparison of the exact and non-exact TP models

	γ (error)
Exact model (6 sv)	0.0262
Non-exact model (3 sv)	0.0947

6.3 Validation

In the simulation, the previously introduced non-exact TP model were used. The comparison of the TP model and the original delayed system was done by a MATLAB/Simulink model (Figure 8). The upper part of the block diagram represents the original system with delay in the feedback loop. The value of the transport delay can be set as constant τ value. The lower part of the figure contains the TP model, where $\dot{\mathbf{x}}$ is computed by the "TP_dx" block and a StateSpace block is used for the integration. For the computation of $\dot{\mathbf{x}}$, the measured (estimated) delay value $\tau(t)$ is required in real implementations. In the constant delay simulation τ is known and wired directly to the "TP_dx" block.

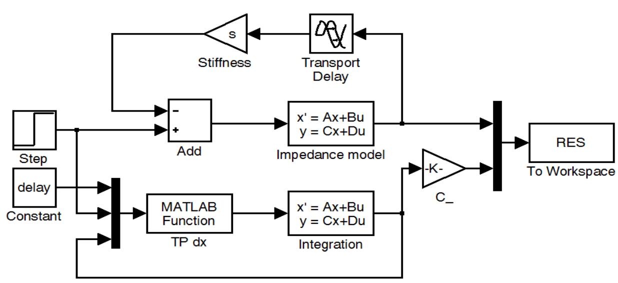

Figure 8: Simulink diagram of the validation

Figure 9 shows the step response of the compared systems. As input signal, a $1N$ force step was used at $0.1s$ in the simulation. As can be seen from the time plot, at an arbitrarily chosen delay value ($\tau = 0.04752$), the identified system is in very close fitting with the original one. For the quantitative comparison the $\mathcal{L}_2$ norm of the position error and the maximum error is computed at four arbitrarily chosen τ values. (neither of them are among the discretized τs.) The result are displayed in Table 2.

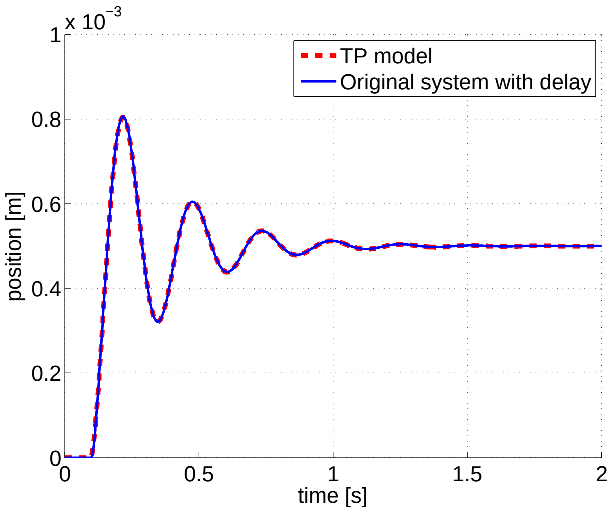

Figure 9: Example for the validation $\tau = 0.04752s$

Table 2: Results of the comparison

	$\mathcal{L}_2$ error	Max error
$\tau = 0.01375s$	2.6279×10^{-5}	9.8521×10^{-7}
$\tau = 0.02941s$	4.0380×10^{-5}	5.9765×10^{-6}
$\tau = 0.04752s$	4.3281×10^{-5}	1.0500×10^{-5}
$\tau = 0.06393s$	1.0851×10^{-4}	1.3048×10^{-5}

The models have been compared under varying delay as well. The value of $\tau(t)$ was varied as a sine function of time $\tau(t) = 0.03 + \sin(t\pi)0.025$. The input signal was a square wave with the frequency of $2Hz$ and amplitude of $1N$. Figure 10 shows the result of the simulation.

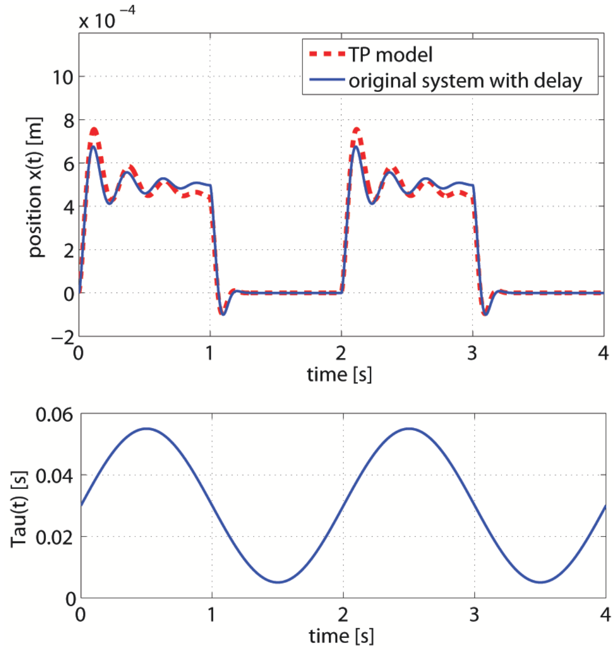

Figure 10: Comparison under varying delay

As worst-case scenario, the models was compared under random delay. τ was generated as uniform random number between $0.001s$ and $0.07s$ in each simulation step. Result of the simulation under random delay is shown in Figure 11.

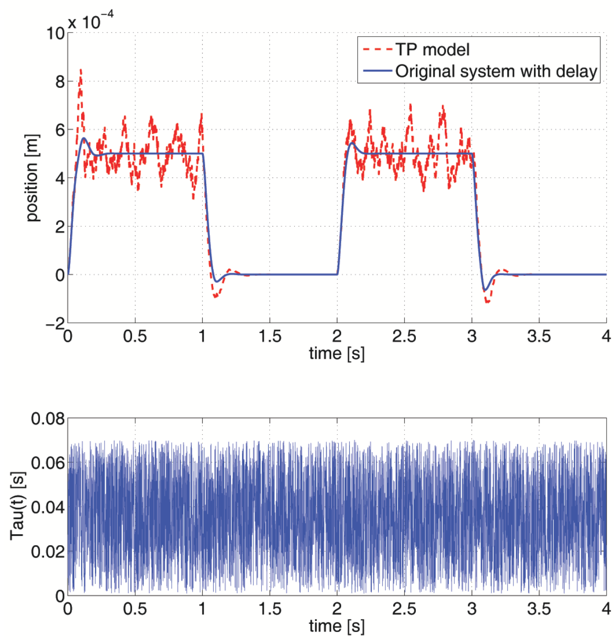

Figure 11: Comparison under random delay

Results with constant delay show very close fitting. Under varying time delay the TP model still follows the original delayed system with reasonable accuracy. The varying delay case is more realistic regarding the application in telemanipulation systems.

7 Conclusion

In this paper, an extended version of TP model transformation were applied to represent a system with feedback delay in finite element TP type polytopic form. As consequence of the transformation the value of the feedback delay become the parameter of the model. As a possible application example, the type of impedance model based control scheme was considered as it is often used in distributed control systems such as bilateral telemanipulation, haptic devices and compliance control of robots. In these systems, feedback delays occurs due to the computer network between the sensor-, actuator- and processing devices.

The proposed transformation method was introduced generally and via a concrete numerical example of a single degree of freedom compliance model. The effect of feedback delay in the discussed system was demonstrated. Simulated results shows, that in case of constant time delay the polytopic model gives practically coincident step responses with the original delayed system. Under varying time delay, the TP model follows the simulated delayed system fairly but with less accuracy. We assume, that the model conformity under varying delay, highly depends on the characteristics of the delay curve: Smooth $\tau(t)$ function with bounded gradients results better conformity. The investigation of this relationship will be the subject of further research.

The resulted polytopic model provides a good basis for LMI based control design. The crux of the method is that the actual value of the feedback delay should be estimated in real-time during the control process. This approach requires that the computing elements be programmed so that the delay value is traceable.

Acknowledgements: The research was supported by the National Research and Technology Agency, (ERC_09) (OMFB-01677/2009) (ERC-HU-09-1-2009-0004 MTASZTAK).

References:

[1] S. Munir and W. J. Book, "Internet-based teleoperation using wave variables with prediction," *Mechatronics, IEEE/ASME Transactions on*, vol. 7, no. 2, pp. 124 –133, Jun. 2002.

[2] I. Polushin, P. Liu, and C.-H. Lung, "A force-reflection algorithm for improved transparency in bilateral teleoperation with communication delay," *Mechatronics, IEEE/ASME Transactions on*, vol. 12, no. 3, pp. 361 –374, june 2007.

[3] C. Cho, J.-B. Song, and M. Kim, "Stable haptic display of slowly updated virtual environment with multirate wave transform," *Mechatronics, IEEE/ASME Transactions on*, vol. 13, no. 5, pp. 566 –575, oct. 2008.

[4] D. Lawrence, "Stability and transparency in bilateral teleoperation," *Robotics and Automation, IEEE Transactions on*, vol. 9, no. 5, pp. 624 – 637, oct 1993.

[5] P. F. Hokayem and M. W. Spong, "Bilateral teleoperation: An historical survey," *Automatica*, vol. 42, no. 12, p. 20352057, Dec. 2006.

[6] P. Galambos, M. Boleraczki, A. Juhasz, , J. Tóth, András Puspoki, and G. Arz, "Development of the prototype of a master-slave servoelectrical gripper with forcefeedback for telemanipulation tasks," in *Proceedings of the Fifth Conference on Mechanical Engineering*, Budapest, Hungary, May 2006.

[7] M. Otsuka, N. Matsumoto, T. Idogaki, K. Kosuge, and T. Itoh, "Bilateral telemanipulator system with communication time delay based on force-sum-driven virtual internal models," in *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, Nagoya, Japan, 1995, pp. 344–350.

[8] W. Kim, B. Hannaford, and A. Bejczy, "Force-reflection and shared compliant control in operating telemanipulators with time delay," *Robotics and Automation, IEEE Transactions on*, vol. 8, no. 2, pp. 176 –185, apr 1992.

[9] P. Korondi, P. Szemes, and H. Hashimoto, *The industrial information technology handbook*. CRC Press, 2005, ch. Internet based Telemanipulation, pp. 601–625.

[10] S. Chiaverini, B. Siciliano, and L. Villani, "A survey of robot interaction control schemes with experimental comparison," *Mechatronics, IEEE/ASME Transactions on*, vol. 4, no. 3, pp. 273 –285, sep 1999.

[11] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "A multi linear singular value decomposition," SIAM *Journal on Matrix Analysis and Applications*, vol. 21, no. 4, pp. 1253–1278, 2000.

[12] P. Baranyi, "Tensor product model based control of 2-d aeroelastic system," *Journal of Guidance, Control, and Dynamics*, vol. 29, no. 2, pp. 391–400, March-April 2006.

[13] P. Baranyi, Z. Petres, P. Vrkonyi, P. Korondi, and Y. Yam, "Determination of different polytopic models of the prototypical aeroelastic wing section by tp model transformation," *Journal of Advanced Computational Intelligence*, vol. 10, no. 4, pp. 486–493, 2006.

[14] P. Baranyi, "TP model transformation as a way to LMI based controller design," *IEEE Transaction on Industrial Electronics*, vol. 51, no. 2, pp. 387–400, April 2004.

[15] P. Baranyi, D. Tikk, Y. Yam, and R. J. Patton, "From differential equations to PDC controller design via numerical transformation," *Computers in Industry, Elsevier Science*, vol. 51, pp. 281–297, 2003.

[16] P. Baranyi, Z. Petres, and S. Nagy, "Tptool – tensor product matlab toolbox," *Website*, 2007. [Online]. Available: http://tptool.sztaki.hu/

[17] M. Tomescu, S. Preitl, and R. Precup, "Stability analysis method for fuzzy control systems dedicated controlling nonlinear processes," *Acta Polytechnica Hungarica*, vol. 4, no. 3, pp. 127–141, 2007.

[18] R. Precup, S. Preitl, B. Ursache, P. A. Clep, P. Baranyi, and J. K. Tar, "On the combination of tensor product and fuzzy models," in *Proceedings of the 2008 IEEE International Conference on Automation, Quality and Testing, Robotics - Volume 02*. IEEE Computer Society, 2008, pp. 48–53.

[19] F. Kolonic, A. Poljugan, and I. Petrovic, "Tensor product model transformation-based controller design for gantry crane control system an application approach," *Acta Polytechnica Hungarica*, vol. 3, no. 4, pp. 95–112, 2006.

[20] L. F. Shampine and P. Gahinet, "Delay-differential-algebraic equations in control theory," *Appl. Numer. Math.*, vol. 56, no. 3, pp. 574–588, 2006.

[21] L. Ljung, *System Identification: Theory for the User*, 2nd ed. Prentice Hall, 1999.

[22] Z. Szabo, P. Gaspar, S. Nagy, and P. Baranyi, "TP model transformation for control-oriented qLPV modeling," *Australian Journal of Intelligent Information Processing Systems*, vol. 10, no. 2, pp. 36–53, 2008.