

Robust and optimal output-feedback control for interval state-space Model: Application to a 2-DoF piezoelectric tube actuator

Mounir Hammouche, Philippe Lutz, and Micky Rakotondrabe*

Automatic Control and MicroMechatronic Systems department

FEMTO-ST Institute

Université Bourgogne Franche-Comté

24, rue Alain Savary

Besançon France

*Email: mrakoton@femto-st.fr

The problem of robust and optimal output feedback design for interval state-space systems is addressed in this paper. Indeed, an algorithm based on Set Inversion Via Interval Analysis (SIVIA) combined with interval eigenvalues computation and eigenvalues clustering techniques is proposed to seek for a set of robust gains. This recursive SIVIA-based algorithm allows to approximate with subpaving the set solutions $[K]$ that satisfy the inclusion of the eigenvalues of the closed-loop system in a desired region in the complex plane. Moreover, the LQ tracker design is employed to find from the set solutions $[K]$ the optimal solution that minimizes the inputs/outputs energy and ensures the best behaviors of the closed-loop system. Finally, the effectiveness of the algorithm is illustrated by a real experimentation on a piezoelectric tube actuator.

sector shown in fig.1 since it directly impacts many dynamic characteristics of the system, such as the degree of stability and damping ratio.

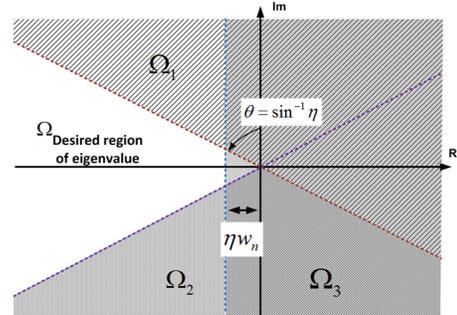


Fig. 1: Region Ω of desired eigenvalues for the closed-loop.

1 Introduction

During the last decades, the problem of designing robust control laws for interval systems has gained much attention [1–10]. The main advantage of intervals is that they permit to model parametric uncertainties easily by bounding them.

The robust control design for interval systems using interval transfer function representations has been widely studied in the literature after the well-known Kharitonov theorem [1–3, 8–11]. However, these techniques were not well adapted to multivariable systems. State-space based interval modeling and control design have therefore been studied.

The robust state-feedback controllers synthesis for interval state-space models has been considered in several works [4, 5, 7, 12]. The above works are focused on placing all the coefficients of the system's closed-loop characteristic polynomial within a desired closed-loop interval characteristic polynomial. However, only the degree of stability of the closed-loop system with state-feedback was addressed and no performances measure was discussed. In fact, the more practically important region for the closed-loop poles is the

Actually, the controllers provided by the robust feedback synthesis are usually not the optimal one and the combination between the robustness and the optimality is always a difficult task and always it necessitates to use some advanced mathematical techniques such as Linear Matrix Inequality (LMI) [13, 14]. Indeed, the process of modeling the errors by intervals gives the opportunity to use the traditional optimal synthesis such that the Linear Quadratic (LQ) approaches. However, the optimal synthesis yields interval Lyapunov and Riccati equations containing interval matrices which are by themselves an actual problem for the mathematics research communities and there are several works under this topics [15–18]. Notwithstanding, in the last decade there are some approaches reformulating the interval Lyapunov and Riccati equations as a convex optimization problem with LMI constrains under some assumptions. Nevertheless these methods contain a lot of parameters to set which

make them not practical.

The paper provides a simple algorithm to find the range of the robust and optimal feedback gains to control any interval system described by interval state-space model. As it is not systematic that all states of the system can be obtained, we restrict the analysis to robust output-feedback design which is not addressed in the previous works that deal with interval systems. However, the algorithm can also be used for robust state-feedback. The proposed algorithm is based on using the regional pole constraints to impact directly the dynamics of the interval system. Moreover, to reduce the computational complexity of the controller design, the problem of regional pole assignment is converted to eigenvalues inclusion problem and it is solved by a set-inversion algorithm called Set Inversion Via Interval Analysis (SIVIA) which is based on interval analysis [19]. Finally, the robust output-feedback design is combined with the Linear Quadratic (LQ) tracker design and a Particle Swarm Optimization (PSO) technique to obtain the feedback gains which robustly ensure the desired performances and, in addition to that, which minimize the linear quadratic cost function.

The paper is organized as follows. Section 2 is dedicated to brief preliminaries on intervals analysis and interval matrices theory including eigenvalues computation. Section 3 presents an overview of the structure of the output-feedback scheme for interval systems as well as the eigenvalues region of interest. Section 4 summaries briefly the general framework to find the robust and the optimal feedback gains. Section 5 gives a description of the proposed approach to synthesize the robust output-feedback controller itself. Section 6 describes the LQ tracker for output-feedback design to find the optimal gain. An application of the proposed method to control a piezoelectric tube actuator is discussed in Section 6. The experimental results and verification are presented in the same section. Finally, conclusions are given in Section 7.

2 Interval analysis and matrix theory preliminaries

During the remaining parts of the paper the standardized notations in interval analysis will be use [20, 21]. The bold font will symbolize interval values whereas usual font will symbolize point values. The interval values may also be denoted by Lie brackets. The lower and upper bounds of an interval will be denoted by underline and overline letters respectively. An interval value $[x, \bar{x}]$ can be defined by the set of $x \subset R$ such that $x \in \mathbf{x}$ and $\underline{x} \leq x \leq \bar{x}$. The set of real interval \mathbf{x} is denoted by IR .

Given two intervals $[x] = [\underline{x}, \bar{x}]$ and $[y] = [\underline{y}, \bar{y}]$. The result of an operation $\diamond \in \{+, -, \cdot, /\}$ between the two intervals is an interval that contains all possible solutions:

$$[x] \diamond [y] = \{x \diamond y \mid x \in [x], y \in [y]\} \quad (1)$$

2.1 Definition of interval matrix

A real interval matrix (interval matrix) is a matrix in which all the elements are interval numbers [21]. Further-

more, an interval matrix is also defined as a family of matrices:

$$\mathbf{A} := [\underline{A}, \bar{A}] = \{A \in R^{n \times n}; \underline{A} \leq A \leq \bar{A}\} \quad (2)$$

where $\underline{A}, \bar{A} \in R^{n \times n}$, $\underline{A} \leq \bar{A}$ being given matrices and the inequality being considered element-wise. The midpoint A_c and the radius A_Δ of \mathbf{A} are denoted respectively by:

$$A_c := \frac{1}{2} (\underline{A} + \bar{A}), \quad A_\Delta := \frac{1}{2} (\underline{A} - \bar{A}) \quad (3)$$

2.2 Eigenvalue computation

The eigenvalue set $\Lambda(\mathbf{A})$ corresponding to \mathbf{A} is defined as the set of all eigenvalues overall $A \in \mathbf{A}$, that is [21],

$$\Lambda(\mathbf{A}) = \{\lambda + i\mu \mid \exists A \in \mathbf{A}, \exists x \neq 0 : Ax = (\lambda + i\mu)x\} \quad (4)$$

Some interval matrices have symmetric interval matrices as subclass. In such a case, the symmetric interval matrix \mathbf{A}^S corresponding to interval matrix \mathbf{A} is defined as the family of all symmetric matrices denoted A^S in \mathbf{A} , that is,

$$\mathbf{A}^S = \{A^S \in \mathbf{A}\} \quad (5)$$

It is worthy to note that a real symmetric interval matrix $\mathbf{A}^S \in IR^{n \times n}$ has n interval eigenvalues which are real. The i^{th} interval eigenvalue of a symmetric interval matrix \mathbf{A}^S is:

$$\lambda_i(\mathbf{A}^S) = [\underline{\lambda}_i(\mathbf{A}^S), \bar{\lambda}_i(\mathbf{A}^S)] := \{\lambda_i(A) \mid A \in \mathbf{A}^S\} \quad i = 1, \dots, n \quad (6)$$

In the sequel, for simplicity reason, sometimes the terms eigenvalues, matrices and vectors will be used instead of interval eigenvalues, interval matrices and interval vectors. Furthermore, the $Re()$ and $Im()$ notations are used to denote the real and imaginary part of a complex number, respectively. Also the $\rho()$ symbol is used to denote the spectral radius of a matrix.

The recent advances on interval analysis computation have provided a new opportunity to estimate the outer bound of the eigenvalues of interval matrices. The estimation of the outer bounds of an interval eigenvalue means finding a unique interval in which all eigenvalues of the interval matrix are bounded. For example, Deif [22] and Kolev [23] proposed some methods to estimate the exact bounds under some hard assumptions which are however not easy to verify as mentioned in [24]. Mayer [25] proposed an enclosure method of the interval eigenvalue for a real and a complex interval matrices based on Taylor expansion whereas Ahn [26] proposed their estimation based on perturbation theory. Recently, Matcovschi [27] employs an optimization technique with Bilinear Matrix Inequality (BML) to find the outer bounds of an interval eigenvalue. A cheap formula to estimate the outer bound is proposed by Rohn [28] for symmetric interval matrices classes. This formula is presented in Theorem.1. The Rohn's result was extended to generalized interval matrices by Hladik [24] as reminded in Theorem.2.

Theorem 1 (Rohn, 2005 [28]). *Let $\mathbf{A}^S \in IR^{n \times n}$. For each $i = \{1, \dots, n\}$ it holds that*

$$\lambda_i(\mathbf{A}^S) \subseteq [\lambda_i(A_c) - \rho(A_\Delta), \lambda_i(A_c) + \rho(A_\Delta)] \quad (7)$$

Theorem 2 (Hladik, 2013 [24]). Let $\mathbf{A} \in \mathbb{R}^{n \times n}$. Then for each eigenvalue $\nu + \mu j \in \Lambda(\mathbf{A})$ we have

$$\underline{\lambda}_n \left(\begin{array}{cc} \frac{1}{2}(\mathbf{A}^T + \mathbf{A}) & 0 \\ 0 & \frac{1}{2}(\mathbf{A}^T + \mathbf{A}) \end{array} \right)^S \leq \nu \leq \overline{\lambda}_1 \left(\begin{array}{cc} \frac{1}{2}(\mathbf{A}^T + \mathbf{A}) & 0 \\ 0 & \frac{1}{2}(\mathbf{A}^T + \mathbf{A}) \end{array} \right)^S \quad (8)$$

$$\underline{\lambda}_n \left(\begin{array}{cc} 0 & \frac{1}{2}(\mathbf{A}^T - \mathbf{A}) \\ \frac{1}{2}(\mathbf{A}^T - \mathbf{A}) & 0 \end{array} \right)^S \leq \mu \leq \overline{\lambda}_1 \left(\begin{array}{cc} 0 & \frac{1}{2}(\mathbf{A}^T - \mathbf{A}) \\ \frac{1}{2}(\mathbf{A}^T - \mathbf{A}) & 0 \end{array} \right)^S \quad (9)$$

where $\underline{\lambda}_1$ and $\overline{\lambda}_n$ are the low and the high extreme eigenvalues limits respectively.

Unfortunately, the proposed Hladik's and Rohn formulas are not always conservatives for large uncertainties and are feasible only when the interval components are thin due to the warping effect. In order to handle this problem, another interesting method for interval eigenvalues computation can be used in the case of large uncertainties is the vertex approach [29,30]. The later is based on the calculation of all exposed edges of interval matrix and the convex hull of all roots of possible characteristic equations. The main disadvantages of this approach is that it takes much time compared to others, but it is a more worthy method.

3 Overview of interval output-feedback design

The output-feedback control problem is among the most important open questions in control engineering [31]. The objective of output-feedback is to find the feedback matrix gain K such that the closed-loop system satisfies some desired transient part performances. In other words, the eigenvalues of the closed-loop system are placed in desired locations. The designed controllers are robust in the sense that all the eigenvalues of the closed-loop which contains the interval system are clustered inside a desired region [32].

Consider a linear Multi Input Multi Output (MIMO) interval uncertain system described by the state-space equation:

$$\begin{cases} \dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t) \\ y(t) = \mathbf{C}x(t) + \mathbf{D}u(t) \end{cases} \quad (10)$$

where $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $y \in \mathbb{R}^p$, $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$, $\mathbf{C} \in \mathbb{R}^{p \times n}$, and $\mathbf{D} \in \mathbb{R}^{p \times m}$. The matrices \mathbf{A} , \mathbf{B} , \mathbf{C} , \mathbf{D} are unknown but bounded by elements lying in known upper and lower bound; i.e. $\mathbf{A} = [\underline{\mathbf{A}}, \overline{\mathbf{A}}]$, $\mathbf{B} = [\underline{\mathbf{B}}, \overline{\mathbf{B}}]$, $\mathbf{C} = [\underline{\mathbf{C}}, \overline{\mathbf{C}}]$, and $\mathbf{D} = [\underline{\mathbf{D}}, \overline{\mathbf{D}}]$.

Remark 1. Notice that even if the input command u is known as non-interval signal, the model above normally yields interval state and output signals x and y respectively due to the interval parameters. However, the real process is non-interval but assumed to have a behavior inside the above model. therefore the signals x and y (and u) are maintained as non-intervals.

Theorem 3 (Smagina Y,2002 [4]). The pair (\mathbf{A}, \mathbf{B}) is controllable for any $A \in \mathbf{A}$ and $B \in \mathbf{B}$ if a square interval matrix

$$\mathbf{Y} = [\mathbf{B}, \mathbf{A}\mathbf{B}, \dots, \mathbf{A}^{n-1}\mathbf{B}] \quad (11)$$

satisfies the condition

$$0 \notin \text{Det}[\mathbf{Y}] \quad (12)$$

Since the theorem of Smagina is simply based on the verification of the rank of the interval controllability matrix using the determinant approach by checking that the zero value does not belong to the interval determinant. This theorem can be extended to verify the controllability and the observability of any interval state-space system with square or non-square controllability and observability interval matrices. However for the case of non-square controllability and observability interval matrices, we need to generate several square matrices from the non-square one and check the rank of each one separately: if the inclusion condition (12) is satisfied for one square matrix then we conclude for the controllability or the observability of the interval system. Furthermore, the theorem of Smagina can also be extended to verify the output controllability of the interval system using the following output controllability matrix [33].

$$\mathbf{Y} = [\mathbf{C}\mathbf{B}, \mathbf{C}\mathbf{A}\mathbf{B}, \dots, \mathbf{C}\mathbf{A}^{n-1}\mathbf{B}, \mathbf{D}] \quad (13)$$

Let us assume that the interval system $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ is output controllable. In classical output-feedback design, the linear output-feedback control law is presented by:

$$u(t) = Ky(t) + Nr(t) \quad (14)$$

where $K \in \mathbb{R}^{m \times p}$ is the output-feedback gains, $N \in \mathbb{R}^{m \times p}$ is the feedforward control gains, and $r(t) \in \mathbb{R}^p$ is the reference input.

For simplicity reason, the modified output-feedback will be used where the output-feedback control law is described by [34]:

$$u(t) = K(y(t) - \mathbf{D}u(t-1)) + Nr(t) \quad (15)$$

3.1 Modified output-feedback with integral compensator

In order to eliminate any steady-state offset that may occur, a compensator must be added to the closed-loop system [35]. The static feedforward gain (DC-gains) N is not the ideal solution in the case of interval system because it always creates a non-null steady-state error [5]. For this purpose the integral compensator is used. The proposed control law is therefore pictured in fig.2 and given by:

$$u(t) = k_y(y - \mathbf{D}u(t-1)) + K_i \xi(t) \quad (16)$$

where $\xi(t)$ is the integral of tracking error (i.e. $\dot{\xi} = r(t) - y(t)$).

It is well known that adding an integral action in the closed-loop system allows being robust with respect to slow system parameters variations. However, model parameters could be exhibit both fast and slow variations - for example in piezoelectric actuators cases, hysteresis could cause

slow and fast variations of parameters while creep nonlinearity and ambient temperature variation could cause very slow variation. In this case, the search for robust gains for the output feedback controller must be considered to ensure the stability and the desired performance of the closed-loop system.

The output-feedback control law can be presented by $(n + p)$ dimensional augmented state vector formed by the state vector $x(t)$ and the integrator variable $\xi(t)$. The state equations for both $\dot{x}(t)$ and $\dot{\xi}$ are consequently:

$$\begin{cases} \dot{x}(t) = (\mathbf{A} + \mathbf{BK}_y\mathbf{C})x(t) + \mathbf{BK}_i\xi(t) \\ y(t) = (\mathbf{C} + \mathbf{DK}_y\mathbf{C})x(t) + \mathbf{DK}_i\xi(t) \\ \dot{\xi}(t) = r(t) - y(t) = r(t) - (\mathbf{C} + \mathbf{DK}_y\mathbf{C})x(t) - \mathbf{DK}_i\xi(t) \end{cases} \quad (17)$$

From (16), the augmented state space model is yielded:

$$\begin{aligned} \begin{pmatrix} \dot{x}(t) \\ \dot{\xi}(t) \end{pmatrix} &= \underbrace{\begin{pmatrix} (\mathbf{A} + \mathbf{BK}_y\mathbf{C}) & \mathbf{BK}_i \\ -(\mathbf{C} + \mathbf{DK}_y\mathbf{C}) & -\mathbf{DK}_i \end{pmatrix}}_{[\mathbf{A}_{aug-cl}]} \begin{pmatrix} x(t) \\ \xi(t) \end{pmatrix} + \underbrace{\begin{pmatrix} 0 \\ \mathbf{I} \end{pmatrix}}_{[\mathbf{B}_{aug-cl}]} r(t) \\ y(t) &= \underbrace{\begin{pmatrix} (\mathbf{C} + \mathbf{DK}_y\mathbf{C}) & \mathbf{DK}_i \end{pmatrix}}_{[\mathbf{C}_{aug-cl}]} \begin{pmatrix} x(t) \\ \xi(t) \end{pmatrix} \end{aligned} \quad (18)$$

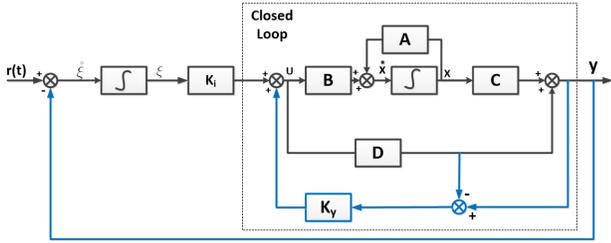


Fig. 2: Output-feedback with integral compensator.

3.2 Performances specifications and region of interest

The output-feedback controller composed of K_y and K_i will be computed by eigenvalues assignment technique. For that let us define the region of interest Ω which represents a convex polygon in the complex plan as shown in fig.1. The specifications of the desired region are related to the desired percent-overshoot $P.O$ and the specified settling time T_S for which the response remains within $n\%$ of the final value given by [34]:

$$P.O = 100.e^{-\frac{\pi\eta}{\sqrt{1-\eta^2}}}; \text{ and } T_S = \frac{1}{\eta\omega_n} \cdot \ln\left(\frac{100}{n}\right) \quad (19)$$

where η and ω_n are the damping ratio and natural pulsation respectively.

3.3 Eigenvalue clustering in subregions for interval systems

This subsection is based on the works of Horng [36] and Gutman [37]. They proposed various conditions which guar-

antee that all eigenvalues of non-interval and interval matrices lie inside algebraic regions of the complex plane. Gutman [37] gave necessary and sufficient conditions to verify that the eigenvalues of a non-interval matrix lie or not in a specified region. These conditions are known as the general theory for matrix root-clustering. As shown in Theorem.4, these conditions are very similar to Lyapunov equations. The proof of the following theorem are detailed in [37].

Theorem 4 (Gutman and Jury 1981 [37]). For the eigenvalue clustering of $A_c \in \mathbb{R}^{n \times n}$ in a region Ω , described by:

$$\begin{aligned} \Omega &= \{(x, y) \mid \gamma_{00} + \gamma_{10}x + \gamma_{01}y < 0\} \\ &= \left\{ \lambda \in \mathbb{C} \mid c_{00} + c_{10}\bar{\lambda} + c_{01}\lambda < 0 \right\} \end{aligned} \quad (20)$$

it is necessary and sufficient that, given any $Q \in H_+^n$ (The set of positive definite hermitian $n \times n$ matrices), there exists a unique $P \in H_+^n$, satisfying the following equation:

$$c_{00}P + c_{10}A_c^T P + c_{01}P A_c^T = -Q \quad (21)$$

where

$$\begin{cases} c_{00} = \gamma_{00} \\ c_{10} = \frac{1}{2}(\gamma_{10} + i\gamma_{01}) \\ c_{01} = \frac{1}{2}(\gamma_{10} - i\gamma_{01}) \end{cases} \quad (22)$$

On the other hand, Horng [36] extended the problem of eigenvalue clustering of non-interval matrices to interval matrices. He gave sufficient conditions permitting to ensure that the eigenvalues of interval matrices lie in various specified subregions of the complex plane such as: half plane, circular disk, parabola,.... These conditions could be very interesting to analyze the performances or to design controllers with robust performances for interval systems. One of these conditions deals with a half-plane subregion (positive, negative, vertical, horizontal, or inclined) of the complex plane and is given by Theorem.5.

Theorem 5 (Horng 1993 [36]). If all the eigenvalues of A_c lie in the half plane $H_{pl}(\gamma_{00}, \gamma_{10}, \gamma_{01})$, then all the eigenvalues of $A \in [\mathbf{A}]$ are inside the same half plane if

$$\rho_{H_{pl}} \equiv \lambda_{Max}([\gamma_{10} - i\gamma_{01}]P[A_{\Delta}]_s) < \lambda_{min}(Q) \quad (23)$$

where $Q \in H_+^n$ and $P \in H_+^n$ satisfies (21).

Particularly, in this paper a combination of three half planes have been used (Ω_1, Ω_2 and Ω_3) to describe the desired region as shown in fig.1. The combination of several subregions is easily done by a propagation method. For more details on the combination of several subregions, refer to [36].

3.4 MIMO cyclic output feedback design

Cyclic design is the process of reducing the problem of Multi Input Multi Output (MIMO) design into a problem of Single Input Multi Output (SIMO) design based on the idea that if a state equation of the MIMO system can be controlled by many inputs, therefore it can be controlled by only one input which is a linear combination of the different initial inputs under the condition that the matrix A is cyclic, i.e. the characteristic polynomial of A equals its minimal polynomial [12]. For this purpose we remind the following three theorems introduced by [38].

Theorem 6 ([38]). *A realization $\{A, B, C\}$ in (10) is minimal if and only if it is controllable and observable.*

Theorem 7 ([38]). *If $\{A, B, C\}$ is a minimal realization, then almost any (constant) output feedback K will make $A+BKC$ cyclic.*

Theorem 8 ([38]). *If $\{A, B\}$ is controllable and A is cyclic, there exists at least one vector q such that $\{A, Bq\}$ is controllable.*

If the cyclic condition is violated for some $A \in \mathbf{A}$ and $B \in \mathbf{B}$ then according to [4] and [38] for the controllable pair (\mathbf{A}, \mathbf{B}) , a feedback matrix K can be found to make the pair $(\mathbf{A} + \mathbf{B}K\mathbf{C}, \mathbf{B})$ cyclic. Adopting the interval extension of this design principle, the robust output feedback gain for interval MIMO systems assumes the form $K = qk$ for almost all $q \in R^m$, where m is the number of system inputs and k can be obtained by letting $[b] = [B]q$ and then by applying the design procedure discussed in the next sections of this paper.

4 Framework for Robust and Optimal Controller Design

In this section the general framework used in this paper is provided to find the robust and the optimal feedback gains in an offline manner. Foremost, to find the range of interval robust gains the proposed SIVIA-based algorithm is used which is a combination of interval eigenvalue computation, eigenvalues clustering, and interval analysis techniques. Moreover, in order to find the optimal gains, the LQ tracker design is used which is based on the resolution of interval Lyapunov equation and the use of Particle Swarm Optimization (PSO) algorithm to minimize a linear quadratic cost function. The flowchart in fig.3 summaries the general framework, whereas, each framework component will be detailed in the next sections(4 and 5).

5 Problem formulation using Set-Inversion

The problem of robust output-feedback control can be summarized by searching for the matrix gain $[K]$ (with $[K] = [[K_y] [K_i]]$) that assigns the interval system eigenvalues to a desired region in the complex plan, taking into account the uncertainty of the interval system. Thus, the problem consists in finding the range of the robust gains $[K]$ of the closed-loop system such that the inclusion (24) is satisfied.

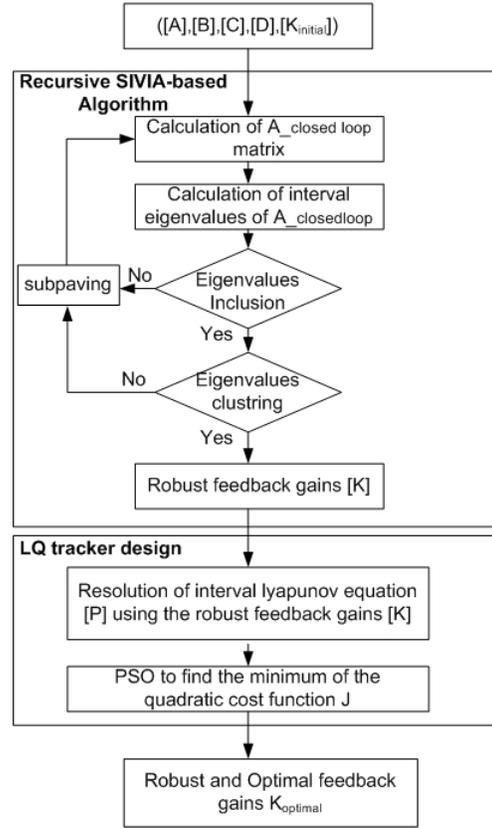


Fig. 3: General framework for the design of optimal and robust controllers.

$$\text{eig}[A_{aug-cl}([A],[B],[C],[D],[K])] \subseteq \Omega_{Desired\ region} \quad (24)$$

where A_{aug-cl} is the augmented closed-loop matrix of the system (18) and $\Omega_{Desired\ region}$ is the desired subregion of eigenvalues.

This problem of interval eigenvalues assignment can be transformed into a set-inversion problem that can be solved using inversion algorithms. The set inversion operation consists in searching the reciprocal image X of a compact set Y by a function f , i.e. $X = f^{-1}(Y) = \{X \in R^n \mid f(x) \in Y\}$ [1, 19]. The Set Inversion Via Interval analysis (SIVIA) algorithm, introduced by Jaulin in 1993 [19], is one of the most powerful approach existing in the literature. This algorithm is based on the subpaving technique which uses a set of non-overlapping boxes to approximate the solution set of the inversion problems. In this work, the SIVIA algorithm is adapted with the aim to approximate with subpaving the solution set $[K]$ that satisfies the inclusion (24). In the sequel, this modified SIVIA algorithm is called by the recursive SIVIA-based algorithm.

The recursive SIVIA-based algorithm to compute the output-feedback parameters is outlined in Table.1. It requires initial box $[K_0]$ that may contain the solution, the interval state-space matrices, the desired region of eigenvalues, and the required accuracy for the paving ϵ . Since the closed-loop matrix is non-symmetric, the eigenvalues inter-

val approximation are calculated by mean of Hladik formula (Theorem.2) whereas in the case of symmetric matrix the Rohn's formula (Theorem.1) can be used. Furthermore the sufficient conditions of eigenvalues clustering (Theorem.5) are used as second check to the solution boxes $[K_{in}]$ that satisfy the eigenvalues inclusion (24) to be sure that the closed-loop eigenvalues are certainly inside the desired regions. The proposed algorithm provides complete information about the ranges of feedback gains including: inner (solution), checked (verified solution), outer (undefined), and infeasible (no solution) subpavings where all the sets subpavings are initially empty.

Remark 2. *In this paper, the Horn clustering conditions is used (Theorem.5) as a second check in our algorithm to be sure that the eigenvalues of the closed-loop system are strictly inside the desired region as explained previously. However, both of the interval eigenvalue computation approach and the eigenvalue clustering technique are guaranteed and it is enough to use only one of them with the SIVIA algorithm to sought for the robust gains.*

Remark 3. *The proposed recursive SIVIA-based algorithm provides a complete information about the ranges of feedback gains that ensure the stability and the desired performance. Contrariwise, it is time demanding especially in the case of system with large number of unknown parameters. This problem is well known to the interval arithmetic community and several works were undertaken to reduce the time of computation. In our case, the synthesis of the robust gains are done offline thus the time consumption is not critical. However, it can always be handled by:- Reducing the complexity of the system by splitting the system to several sub-systems using cyclic design or decentralized feedback control (as explained in the sequel of this paper). - Using the contractor-separator approach proposed recently by Jaulin in [39]. - Furthermore, the parallel structure of the recursive SIVIA-based algorithm allows the use of parallelism and distributed computation and other intelligent methods.*

Remark 4. *For the output-feedback, when $mp \geq n$ (m, n and p are defined in (10)) the recursive SIVIA-based algorithm can find the solution easily whereas in the case of SISO (Single Input Single Output) or when $mp < n$ it is not guaranteed that the algorithm finds a solution. In fact there are several necessary and sufficient conditions to verify the existence of output-feedback gains as outlined in [31]. However in our algorithm if the solution doesn't exist the algorithm will stop at the first iteration in step 3 as long as $eig([A_{aug-cl}]) \cap Y = \emptyset$ is maintained.*

The proposed recursive SIVIA-based algorithm can be used for robust output-feedback as well as for robust state-feedback. In addition it can also be used for non-interval as well as for interval systems. In all these cases it only suffices to replace the closed-loop matrix $A_{closedloop}$ by the corresponding one.

Table 1: The proposed recursive SIVIA-based algorithm for solving a set-inversion problem.

	SIVIA (in: $[A], [B], [C], [D], [K] = [initialbox], [k_{in}] = \emptyset, [K_{out}] = \emptyset, [K_{Unfeasible}] = \emptyset, [K_{checked}] = \emptyset, \epsilon, Y = \Omega_{DesiredregionofEigenvalue}$)
Step 1	Iteration i - Calculate $A_{aug-cl}([A], [B], [C], [D], [K])$ - Calculate $eig([A_{aug-cl}])$ using Theorem.1 or 2
Step 2	-If $eig([A_{aug-cl}]) \subseteq Y$ Then $[k_{in}] = [k_{in}] \cup [K]$ -If the eigenvalues clustering in subregions algorithm is checked (Theorem.5) Then $[k_{checked}] = [k_{checked}] \cup [K_{in}]$ Go to step 6
Step 3	-If $eig([A_{aug-cl}) \cap Y = \emptyset$ Then $[k_{Unf}] = [k_{Unf}] \cup [K]$ Go to step 6
Step 4	-If $width[K] < \epsilon$ Then $[k_{out}] = [k_{out}] \cup [K]$ Go to step 6
Step 5	- Else bisect $[K]$ and stack the two resulting boxes.
Step 6	-If the stack is not empty, then unstack into $[K](i+1)$, increment i and go to Step 1. -Else End.

6 LQ tracker design to choose the optimal gains

In fact, the obtained solution by the proposed recursive SIVIA-Based algorithm will guarantee that any choice inside the solution boxes $[K]$ will ensure the specified performances. However, there is no method to choose among these solutions the optimal gains that minimize the inputs/outputs energy and ensure the best behaviors of the closed-loop system. For this reason the LQ design is adopted to handle this problem. Indeed, in control design we are often interested not in regulating the state near zero but in following a nonzero reference command signal. This reference-input tracking or servodesign problem is called LQ tracker problem. In this section the LQ tracker design of an interval state-space model is covered.

First of all, to streamline the notation let us start by redefining the input and the output matrices as described by equations (25).

$$\begin{aligned}
 \begin{pmatrix} \dot{x}(t) \\ \dot{\xi}(t) \end{pmatrix} &= \underbrace{(A^* + B^* K^* C^*)}_{C_c} \begin{pmatrix} x(t) \\ \xi(t) \end{pmatrix} + \underbrace{\begin{pmatrix} 0_{n \times m} \\ I_{m \times m} \end{pmatrix}}_{C_c} r(t) \\
 \dot{X}(t) &= A_c X(t) + B_c r(t) \\
 y(t) &= \underbrace{(C^* + D^* K^* C^*)}_{C_c} \begin{pmatrix} x(t) \\ \xi(t) \end{pmatrix} \quad (25)
 \end{aligned}$$

such that

$$A^* = \begin{pmatrix} A & 0_{n \times p} \\ -C & 0_{p \times p} \end{pmatrix}; B^* = \begin{pmatrix} B \\ -D \end{pmatrix}; C^* = \begin{pmatrix} C & 0_{p \times m} \\ 0_{m \times n} & I_{m \times m} \end{pmatrix};$$

$$K^* = (k_y, k_i); D^* = \begin{pmatrix} \mathbf{D} \\ 0_{p \times m} \end{pmatrix}$$

The LQ tacker design is based on finding the feedback coefficient matrix K that minimizes the following quadratic cost function [40],

$$J = \int_0^{\infty} (\tilde{X}^T Q \tilde{X} + u^T R u) + \frac{1}{2} e^T V e \quad (26)$$

with $R > 0$, $Q > 0$, and $V \geq 0$ are the weighting parameters. The e and $\tilde{X}(t)$ are the tracking and the state errors respectively and they are given by:

$$e = r(t) - y(t) \quad (27)$$

$$\tilde{X}(t) = X(t) - X_{ss} \quad (28)$$

where X_{ss} is the steady state response. Since the closed-loop system will be asymptotically stable for acceptable design and for any $A_c \in \mathbf{A}_c$ and $B_c \in \mathbf{B}_c$, thus the steady state response can be given by (29).

$$\dot{X} = 0 = A_c X_{ss} + B_c r_0 \Rightarrow X_{ss} = -A_c^{-1} B_c r_0 \quad (29)$$

where the matrices A_c and B_c are non-intervals and belong to the interval matrices \mathbf{A}_c and \mathbf{B}_c respectively.

Due to the integral compensator, the steady-state error e is automatically equal to zero. The quadratic cost function is

$$J = \int_0^{\infty} (\tilde{X}^T Q \tilde{X} + u^T R u) \quad (30)$$

The weighting matrices (Q and R) regulate the penalties on the transient part of the state variables $X(t)$ and the control signal $u(t)$.

The optimal cost is found to satisfy the equation (31), for more detail refer to [40].

$$J = \frac{1}{2} \tilde{X}^T(0) \mathbf{P} \tilde{X}(0) \quad (31)$$

Where \mathbf{P} is a positive definite interval matrix and is a solution of the following interval Lyapunov equation

$$0 = g = \mathbf{A}_c^T \mathbf{P} + \mathbf{P} \mathbf{A}_c + Q + \mathbf{C}^T K^T R K C \quad (32)$$

From (28) and from the initial condition (i.e. the system starts at rest $X(0) = 0$), we define $\tilde{X}(0) = -X_{ss}$ so that the optimal cost (31) becomes,

$$J = \frac{1}{2} \tilde{x}^T(0) \mathbf{P} \tilde{x}(0) = \frac{1}{2} \text{tr}(\mathbf{P} \mathbf{S}) \quad (33)$$

$$\text{with } S = X_{ss} X_{ss}^T = A_c^{-1} B_c r_0 r_0^T B_c^T A_c^{-T}.$$

It is now necessary to solve for the optimal feedback gain K that minimizes the quadratic cost function. In order to find the optimal gains there are gradient-based and non-gradient-based approaches [40]. Generally the gradient-based are faster than the non-gradient techniques. However in the case of interval state-space model the gradient approaches are not applicable because the Lyapunov equation

(32) is an interval equation as shown by (34), where the solution \mathbf{P} is non-convex. Therefore another method based on Particle Swarm Optimization (PSO) is proposed as explained in the following subsections.

6.1 Resolution of interval Lyapunov equation

Let the interval Lyapunov equation be defined by (34):

$$0 = [A_c^T][P] + [P][A_c] + [C]^T K^T R K [C] + Q \quad (34)$$

To solve an interval Lyapunov equation of the form (34), there are many iterative and direct methods existing in the literature [41,42]. In fact in the literature, only the case of interval Sylvester equation is studied. However, since the interval Lyapunov equation is a special form of the interval Sylvester equation, all the previous studies can be used for the resolution of (34). These methods are based on different techniques such as simulation approach, linear programming, iterative techniques as well as vertex approach [41–43]. However the simulation approach is very time consuming because it is simulated by mean of Monte Carlo method and in each step a Sylvester equation is solved via a Hessenberg-Schur which leads to huge computational time. In addition the linear programming approach [43] is based on solving $2^{m \times n}$ linear programming problems that makes this approach very troublesome even with small values of m and n (defined previously (10)) [41]. Furthermore the iterative techniques are proved to be less efficient and work only for narrow interval matrices as explained in [41], whereas the vertex approach proposed by Rohn [42] is shown to be a good method for solving the interval Lyapunov equation nevertheless its computational time increases with the number of interval matrices elements.

The vertex approach consists in computing a set solution of 2^n extreme or vertex matrices of the interval linear system (35) obtained from the interval Lyapunov equation (34) by means of the Kronecker product (36).

$$[G] \mathbf{p} = [F] \quad (35)$$

where,

$$[G] = ([A_c^T] \otimes I_n) + (I_n \otimes [A_c]) \quad (36)$$

and

$$\mathbf{p} = \text{vec}([P]) = ([P_{11}], [P_{12}], \dots, [P_{1n}], \dots, [P_{m1}], [P_{m2}], \dots, [P_{mn}]^T)$$

$$[F] = \text{vec}(-[C]^T K^T R K [C] - Q) = ([f_{11}], \dots, [f_{1n}], \dots, [f_{m1}], \dots, [f_{mn}])^T$$

Here, the Kronecker product $W \otimes Z$ of two matrices W and Z is the block matrix whose (i, j) block is $w_{ij} Z$.

Eventually the interval solution $[P]$ is obtained using convex-hull function on the 2^n solution set of P .

6.2 Using Particle Swarm Optimization (PSO) to find the optimal gains

In fact, by using the convex-hull function an overestimated interval solution of $[P]$ is obtained. In order to avoid the convex-hull function and since we are only interested in the minimization of the quadratic cost function (33), we propose to calculate the sum of the 2^n quadratic cost function as shown in (37) corresponding to 2^n solutions of Lyapunov equation.

$$J(k_1, \dots, k_n) = \frac{1}{2} \left(\sum_{i=1}^{2^n \text{vertex}} P_i(k_1, \dots, k_n) S \right) \quad (37)$$

In other hand, we can notice that the quadratic cost function (37) can be obtained if and only if all the 2^n Lyapunov equations have solutions. This condition can be verified by mean of interval eigenvalues as shown in Theorem.9.

Theorem 9 (Seif, 1994 [41]). *A necessary and sufficient condition for the interval Lyapunov equation is that $0 \notin \lambda_i^l$, $\forall i$ where $\lambda_i^l = \{\lambda_i(A_c) : A_c \in \mathbf{A}_c\}$ is the interval eigenvalue.*

Actually, there is a complementary coordination between the previous section of regional poles assignment and LQ tracker design. Indeed, the LQ tracker design provides an optimal gains if and only if all the interval eigenvalues of the closed-loop system doesn't equal zero (Theorem.9) and the system is asymptotically stable. However, the regional poles assignment technique (Table.1) provides the intervals of gains which ensure that all the interval eigenvalues are strictly negative. Therefore, it is sufficient to apply an iterative search strategy only on the solution boxes provided by the recursive SIVIA-based algorithm (Table.1) to find the gains that minimize the quadratic cost function (37). This search can be done using intelligent optimization techniques such as Particle Swarm Optimization (PSO) and Genetic algorithm (GA).

PSO and GA have an equal effectiveness to find the minimum quadratic cost function but in view of computational efficiency, we propose to use the PSO because it consumes less time than the GA to converge to the minimum as explained in [44]. An overview of Particle Swarm Optimization (PSO) technique is provided in Appendix.A.

Before moving to the experimental validation of the proposed approach, the main work done in this paper can be summarized by the following points:

- If an interval system $(\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ is output controllable (i.e. satisfies (13)) for any $A \in \mathbf{A}$ and $B \in \mathbf{B}$, then A set of robust gains can be found using the proposed SIVIA-based algorithm (Table1) that ensure the stability and satisfy the desired performances that take into account the physical limitation of the system.

- If the interval system is a high order system, it is always preferable to reduce the complexity of the problem of searching for a robust gains by reducing the order of the system using cyclic design technique as outlined in subsection 3.4.

- After finding the set of robust gains using the SIVIA-based algorithm we can always search for an optimal gains

among the set of the robust gains by the resolution of the interval Lyapunov equation (34) and the use of the PSO techniques as summarized in section.6.

7 Application to piezoelectric tube actuator

Piezoelectric tube actuators are among the most used actuator in micro/nano-scales applications particularly in micro/nano manipulation, Scanning Probe Microscopy (SPM), and Atomic Force Microscopy (AFM) due to their high speed (large bandwidth up to 1kHz), high precision (sub-nanometric), high resolution, and multi-degrees of freedom [45–48]. Unfortunately, piezoelectric tube actuators are characterized by nonlinearities (hysteresis and creep) and by high sensitivity to the environment (vibrations and thermal variation) which will be discussed in the next subsection.

The experimental setup is pictured in fig.4. It is composed of a piezoelectric tube actuator (PT230.94), two optical displacement sensors (LC2420 from Keyence company), a voltage amplifier (up to $\pm 200V$). The actuator and the sensors are connected to the computer through a dSPACE-1103 acquisition board. The piezoelectric tube is made of lead-zirconate-titanate (PZT) material coated by one inner electrode (in silver) and four external electrodes (in copper-nickel alloy).

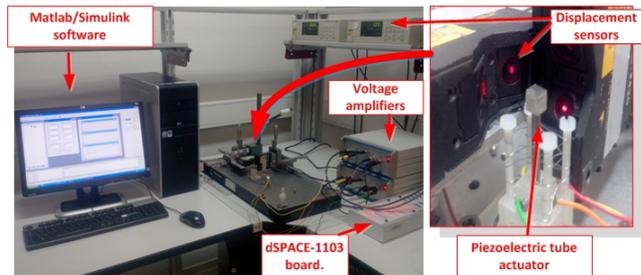


Fig. 4: Presentation of the experimental setup.

In order to control the tube along X-axis or Y-axis, a potential $+U$ is applied on one electrode and the opposite potential $-U$ is applied to the counterpart electrode as depicted in fig.5. Furthermore, if potentials with the same sign is applied on the four electrodes a relative displacement on the Z-axis will be produced. In the terminal of the piezoelectric tube, a small cube with perpendicular and flat sides have been placed to be able to measure the linear displacement of the tube deflections.

7.1 Modeling of piezoelectric tube actuator

Piezoelectric actuators such as piezoelectric tube and piezoelectric multimorph cantilever are sensitive to the environmental variations (e.g. temperature variation) and are characterized by nonlinearities (hysteresis and creep..) which make the control of these systems not a trivial task. In fact, there are several methods to control these actuators including real-time adaptive and nonlinear techniques that take

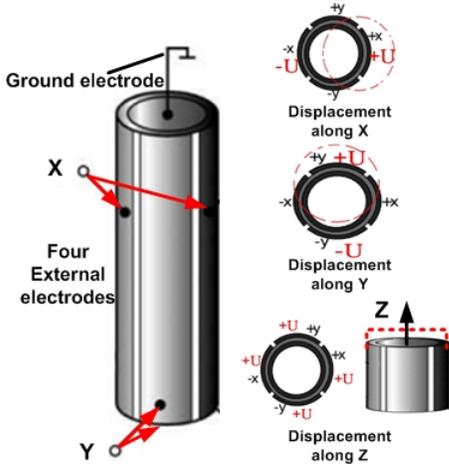


Fig. 5: Structure and operation of the piezoelectric tube actuator.

into account these nonlinearities [3, 45, 49]. There are also other robust methods to control the piezoelectric actuators on the basis of linear models with uncertainties that embrace the nonlinearities [50, 51]. Moreover, in [1, 2, 9], an interval technique has been used which is based on bounding the parametric uncertainties by intervals using transfer function representation. In this paper an interval state-space representation will be used to further derive a robust and optimal controller using the proposed algorithm.

During the experimental process we focus only on two axes (two degrees of freedom: 2-DoF). We will note U_x and U_y the related applied voltages. The 2-DoF model for the piezoelectric tube actuator can be expressed by the following linear equations which represent the relation between the deflections along the x and y axes and the applied input voltages, whereas, the actuator sensitivity and its nonlinear behaviors will be approximated by parametric uncertainties bounded by intervals [45].

$$\begin{aligned}\sigma_x &= G_{xx}(s)U_x + G_{xy}(s)U_y \\ \sigma_y &= G_{yx}(s)U_x + G_{yy}(s)U_y\end{aligned}\quad (38)$$

where σ_x and σ_y are the deflections along x- and y-axis. $G_{xx}(s)$, $G_{xy}(s)$, $G_{yx}(s)$, and $G_{yy}(s)$ are the transfer functions that represent the relations between $U_x \rightarrow \sigma_x$, $U_x \rightarrow \sigma_y$, $U_y \rightarrow \sigma_x$, and $U_y \rightarrow \sigma_y$ respectively.

To characterize the 2-DoF piezoelectric tube, we apply first a step voltage U_x of amplitude 200V and we set U_y to zero and capture the deflections σ_x and σ_y . Then we repeat the same procedures with U_y in which we apply a step voltage of amplitude 200V on U_y where U_x is a null input and we capture the deflection σ_y and σ_x . The transfer function $G(s)$ to be identified for 2-DoF piezoelectric tube is composed of four transfer functions $G_{xx}(s)$, $G_{xy}(s)$, $G_{yx}(s)$, and $G_{yy}(s)$ as shown in equation (39). After identification by mean of System Identification MatlabToolbox [52] using Box-Jenkins method and the step response data, the transfer functions (39) is defined. Particularly, a fourth order model for $G_{xx}(s)$ and $G_{yy}(s)$ and a second order model for cross-couplings trans-

fer functions $G_{yx}(s)$ and $G_{xy}(s)$ have been chosen in our case because they are largely sufficient to represent the dynamics of the piezoelectric tube.

$$G(s) = \begin{pmatrix} G_{xx}(s) & G_{xy}(s) \\ G_{yx}(s) & G_{yy}(s) \end{pmatrix}\quad (39)$$

where,

$$G_{xx}(s) = \frac{(0.144)s^4 + (128.4)s^3 + (5.151 \times 10^5)s^2 + (1.603 \times 10^7)s + (3.45 \times 10^7)}{s^4 + (574.5)s^3 + (4.735 \times 10^6)s^2 + (1.454 \times 10^8)s + (3.053 \times 10^8)}$$

$$G_{xy}(s) = \frac{(-0.0004783)s^2 + (4.155)s + (10.75)}{s^2 + (1055)s + (2926)}$$

$$G_{yx}(s) = \frac{(0.00367)s^2 + (1.076)s + (0.2271)}{s^2 + (213.1)s + (54.99)}$$

$$G_{yy}(s) = \frac{(0.1773)s^4 + (29.88)s^3 + (2.148 \times 10^6)s^2 + (4.518 \times 10^7)s + (6.658 \times 10^7)}{s^4 + (265.6)s^3 + (1.98 \times 10^7)s^2 + (4.061 \times 10^8)s + (5.816 \times 10^8)}$$

To obtain the interval model, we propose to consider each parameter of (39) as center and we add a radius of 10%. We therefore obtain:

$$G_{xx}(s) = \frac{[b_{10}]s^4 + [b_{11}]s^3 + [b_{12}]s^2 + [b_{13}]s + [b_{14}]}{s^4 + [a_{11}]s^3 + [a_{12}]s^2 + [a_{13}]s + [a_{14}]}$$

$$G_{xy}(s) = \frac{[b_{20}]s^2 + [b_{21}]s + [b_{22}]}{s^2 + [a_{21}]s + [a_{22}]}$$

$$G_{yx}(s) = \frac{[b_{30}]s^2 + [b_{31}]s + [b_{32}]}{s^2 + [a_{31}]s + [a_{32}]}$$

$$G_{yy}(s) = \frac{[b_{40}]s^4 + [b_{41}]s^3 + [b_{42}]s^2 + [b_{43}]s + [b_{44}]}{s^4 + [a_{41}]s^3 + [a_{42}]s^2 + [a_{43}]s + [a_{44}]}$$

where,

$$\begin{aligned}[b_{10}] &= [0.1295, 0.1584] & ; & [a_{11}] = [517.0544, 631.9554] ; \\ [b_{11}] &= [115.5967, 141.2849] & ; & [a_{12}] = [4.2614, 5.2085] \times 10^6; \\ [b_{12}] &= [4.6361, 5.6665] \times 10^5 & ; & [a_{13}] = [1.3083, 1.5992] \times 10^8; \\ [b_{13}] &= [1.4428, 1.7636] \times 10^7 & ; & [a_{14}] = [2.7480, 3.3588] \times 10^8; \\ [b_{14}] &= [3.1052, 3.7954] \times 10^7 & ; & [a_{21}] = [0.9492, 1.1603] \times 10^3; \\ [b_{20}] &= [-0.5262, -0.4304] \times 10^{-3} & ; & [a_{22}] = [2.6331, 3.2183] \times 10^3; \\ [b_{21}] &= [3.7399, 4.5711] & ; & [a_{41}] = [239.0092, 292.1225] ; \\ [b_{22}] &= [9.6729, 11.8225] & ; & [a_{42}] = [1.7819, 2.1780] \times 10^7; \\ [b_{40}] &= [0.1595, 0.1951] & ; & [a_{43}] = [3.6549, 4.4672] \times 10^8; \\ [b_{41}] &= [26.8926, 32.8689] & ; & [a_{44}] = [5.2346, 6.3979] \times 10^8; \\ [b_{42}] &= [1.9330, 2.3627] \times 10^6 & ; & [a_{31}] = [191.8224, 234.4497] ; \\ [b_{43}] &= [4.0660, 4.9697] \times 10^7 & ; & [a_{32}] = [49.4899, 60.4878] ; \\ [b_{44}] &= [5.9926, 7.3244] \times 10^7 & ; & [b_{31}] = [0.9688, 1.1842] ; \\ [b_{30}] &= [3.3, 4.1] \times 10^{-3} & ; & [b_{32}] = [0.2043, 0.2498] ;\end{aligned}$$

In fact, it is shown that creating the interval model by using 10% of radius calculated from the center is sufficient enough for a large number of cases and of applications in piezoelectric actuators [1, 2]. Also these 10% are a good compromise between the thinness of parameters uncertainties and the feasibility to find a feedback controller.

The above interval transfer function model of the piezoelectric tube actuator can be expressed by the following MIMO state-space model using canonical blocks [38, 53]:

$$\begin{cases} \dot{x}(t) = \mathbf{A}x(t) + \mathbf{B}u(t) \\ y(t) = \mathbf{C}x(t) + \mathbf{D}u(t) \end{cases}\quad (41)$$

where,

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ -[a_{14}] - [a_{13}] - [a_{12}] - [a_{11}] & 0 & 0 & -[a_{22}] - [a_{21}] & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & -[a_{32}] - [a_{31}] - [a_{44}] - [a_{43}] - [a_{42}] - [a_{41}] & 0 & 0 & 0 & 0 & 1 \end{bmatrix};$$

$$\mathbf{B} = \begin{bmatrix} 00010001 \\ 00010001 \end{bmatrix}^T; \mathbf{D} = \begin{bmatrix} [b_{10}] & [b_{20}] \\ [b_{30}] & [b_{30}] \end{bmatrix}; \mathbf{C} = \begin{bmatrix} C_{xx} & C_{xy} \\ C_{yx} & C_{yy} \end{bmatrix}$$

$$C_{xx} = [b_{14}] - [a_{14}][b_{10}][b_{13}] - [a_{13}][b_{10}][b_{12}] - [a_{12}][b_{10}][b_{11}] - [a_{11}][b_{10}];$$

$$C_{xy} = [00][b_{22}] - [a_{22}][b_{20}][b_{21}] - [a_{21}][b_{20}];$$

$$C_{yx} = [00][b_{32}] - [a_{32}][b_{30}][b_{31}] - [a_{31}][b_{30}];$$

$$C_{yy} = [b_{44}] - [a_{44}][b_{40}][b_{43}] - [a_{43}][b_{40}][b_{42}] - [a_{42}][b_{40}][b_{41}] - [a_{41}][b_{40}];$$

7.2 Controller calculation and experimental tests

The use of the interval model of the piezoelectric tube allows us to find a robust output-feedback controller which satisfies the desired performances. Indeed, in micro positioning applications, the rapidity is highly required and the overshoot is extremely undesirable because it may cause micro/nano objects damage. Therefore, the following desired performances are adopted: negligible overshoot (1%) and a settling time $T_s \leq 40ms$. By using (19) we have $\xi = 74.9$ and $\omega_n = 52rad/s$.

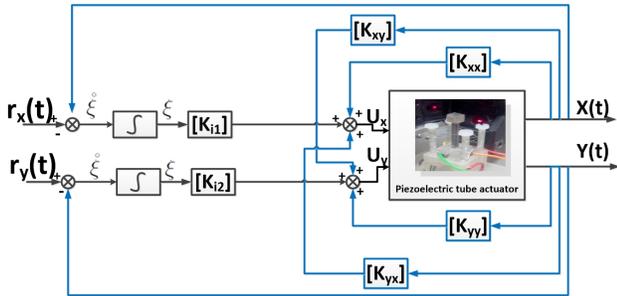


Fig. 6: The control design of a multi-input multi-output piezoelectric tube actuator using output-feedback with integral compensator.

Before using the recursive SIVIA-based algorithm (Table.1) to compute the real gain $p \times n$ matrix K for output-feedback control, the MIMO cyclic output-feedback design is used to reduce the complexity of calculation which is simply based on converting the MIMO to SIMO system under cyclic condition as explained previously. However to make the problem as simple as possible, before applying cyclic design, the MIMO system is considered as two subsystems with two independent outputs. Thus, two Multi Input Single Output (MISO) systems are obtained ($\mathbf{A}, \mathbf{B}, \mathbf{C}_1, \mathbf{D}_1$) and ($\mathbf{A}, \mathbf{B}, \mathbf{C}_2, \mathbf{D}_2$) where $\mathbf{C}_1 = [C_{xx} \ C_{xy}]$, $\mathbf{C}_2 = [C_{yx} \ C_{yy}]$, $\mathbf{D}_1 = [D_{xx} \ D_{xy}]$ and $\mathbf{D}_2 = [D_{yx} \ D_{yy}]$ then the cyclic design is applied on each

subsystem by replacing $\mathbf{b} := [B]q$ such that $q = [\alpha\beta]^T$. In order to retain controllability, we will need to choose α and β such that $\alpha + \beta \neq 0$. This condition ensures that \mathbf{b} is not a zero vector. For each subsystem we choose its corresponding vector q . For example we can choose $q_1 = [1 \ 0.1]^T$ and $q_2 = [0.1 \ 1]^T$. In fact the recursive SIVIA-based algorithm (Table.1) can be used directly to compute the real gain $p \times n$ matrix K for the output-feedback control but it takes much time, so it is always preferable to reduce the complexity of computation and the number of parameters as possible to make the convergence to the solution more rapidly. In the case of MIMO systems, this can be done by using cyclic design or decentralized feedback control [54].

To characterize the set solution $[K]$ of each subsystem (with $[K] = [[K_{y1} \ K_{i1}]$ or $[K] = [[K_{y2} \ K_{i2}]$) the proposed recursive SIVIA-based algorithm described in Table.1 is used. Foremost an initial box $[K_y] \times [K_i] = [-5 \times 10^1, 5 \times 10^1] \times [-1 \times 10^3, 1 \times 10^3]$ is chosen for each subsystem and an accuracy of paving $\epsilon = 0.1$. The obtained subpaving of each subsystem is depicted in fig.7. The red boxes correspond to the inner subpavings $[K_{in}]$, i.e. the set solutions $[K_y]$ and $[K_i]$ that satisfy the inclusion (24). The white boxes correspond to the subpavings $[K_{Infeasible}]$ where the inclusion condition is not satisfied. The yellow boxes refer to $[K_{out}]$ where no decision on the inclusion is taken. The boxes in green color correspond to the checked solution $[K_{checked}]$ in which the sufficient condition of eigenvalues clustering in the desired region (Theorem.5) is verified.

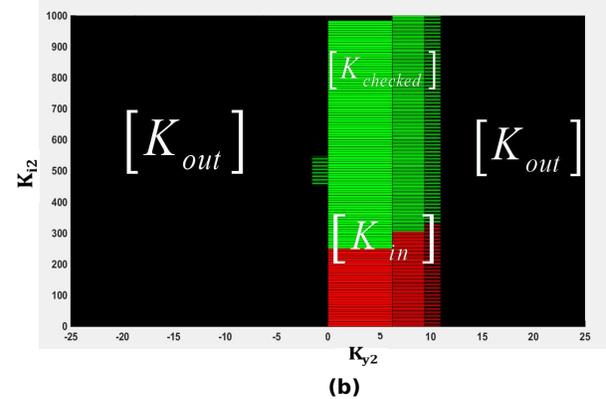
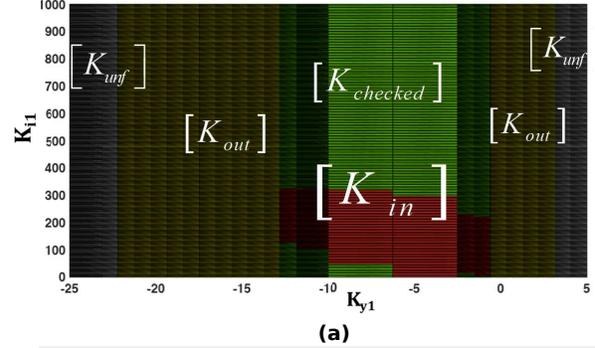


Fig. 7: Resulting subpaving $[K_y]$ and $[K_i]$.

Furthermore to test the obtained solutions an arbitrary

point from the checked solution fig.7-a is selected that corresponds to the controller parameters of the first subsystem $K_{y1} = -5$ and $K_{i1} = 1000$, and from fig.7-b with the same manner the parameters of the second subsystem is chosen as $K_{y2} = -1$ and $K_{i2} = 1000$. Thus the obtained gain matrices are:

$$k_y = [q_1 K_{y1}, q_2 K_{y1}] = \begin{bmatrix} -5 & 1 \\ -0.5 & 10 \end{bmatrix}, \text{ and } k_i = \begin{bmatrix} K_{i1} & 0 \\ 0 & K_{i2} \end{bmatrix} = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix}.$$

The simulation and experimental step responses for the closed-loop system are plotted in fig.8 and fig.10 respectively. In fig.8, three simulation results were performed using three different values of the system matrices (A, B, C, D) inside the interval system ($[A], [B], [C], [D], [K]$), where the $\text{sup}()$, $\text{inf}()$, and $\text{mid}()$ refer to the superior, inferior, and middle of the interval matrices. Indeed, it is clear that the closed-loop system matrices satisfies the desired performance with negligible overshoot and settling time less than 40ms.

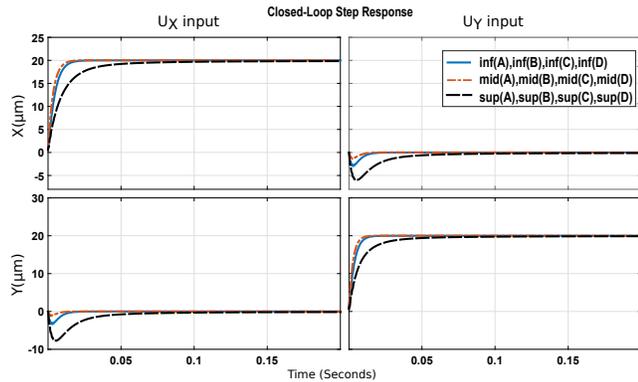


Fig. 8: Step response of piezoelectric tube for the closed-loop system (Simulation using Matlab).

In order to simulate the impact of the systems uncertainties on the locations of the closed-loop eigenvalues and to show the robustness of the proposed controller against these uncertainties, a simulation test using Monte-Carlo Technique is performed. During the simulation test, robust control gains were selected randomly from the solution boxes, depicted in fig.7, and values for the system matrices (A, B, C) inside the interval system ($[A], [B], [C]$) and in each time the obtained eigenvalues are drawn, as depicted in fig.9. The eigenvalues placement show clearly that the closed-loop system is always stable and satisfies the desired performance related to the damping ratio and the natural pulsation (defined previously), as well as the system matrices are inside the interval system ($[A], [B], [C]$) which means that the controller is robust against system uncertainties.

Fig.11 represents the bode diagrams of the identified transfer functions G_{xx} , G_{xy} , G_{yx} , and G_{yy} and the closed-loop transfer functions T_{xx} , T_{xy} , T_{yx} , and T_{yy} . This figure shows that, with the calculated controller, the closed-loop system

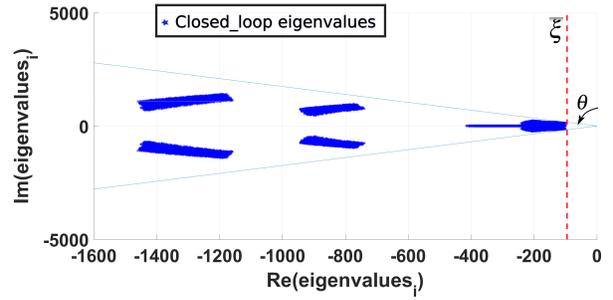


Fig. 9: The test of the robustness of the closed-loop system using Monte-Carlo techniques.

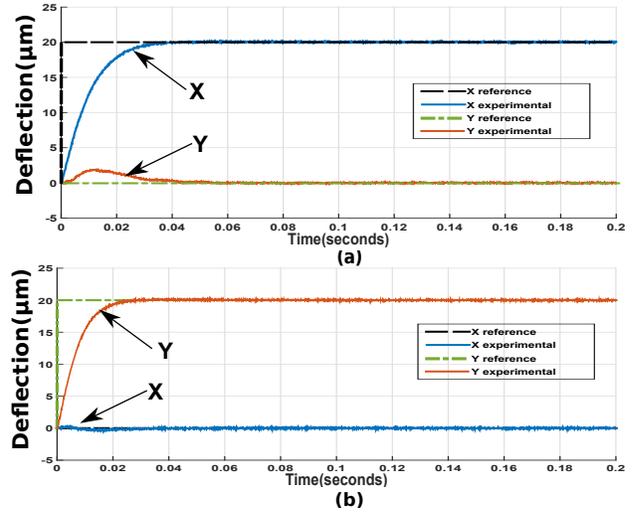


Fig. 10: Step response of piezoelectric tube for the closed-loop system (experimental test).

achieves a convenient bandwidth relative to the bandwidth of the open-loop system for G_{xx} and G_{yy} . Moreover, from the bode diagram of the cross-couplings transfer functions G_{xy} and G_{yx} (fig.11-b and -c) it can be seen that the closed-loop system ensures the rejection of the perturbation in low and high frequencies, whereas, for the low frequencies the rejection has a very small amplitude compared with high frequencies.

In order to test the tracking performances of the closed-loop system, an experimental test is carry out with various trajectories including helix and series of steps shape trajectories. The result for the series of steps trajectory is depicted in fig.12. It is clearly shown that the piezoelectric tube actuator tracks successfully the reference with the desired performances (negligible overshoot (1%) and settling time less than 40ms).

To test the ability of the closed-loop to track complex trajectories, the piezoelectric tube has been forced to track an helix shaped trajectory, by applying simultaneously a sine wave of amplitude 20µm on X and Y axes with frequencies 0.2Hz and 0.5Hz, respectively. Fig.13 shows that the helix trajectory tracking is successfully achieved, with the tracking error less than 0.5µm for X and Y axis.

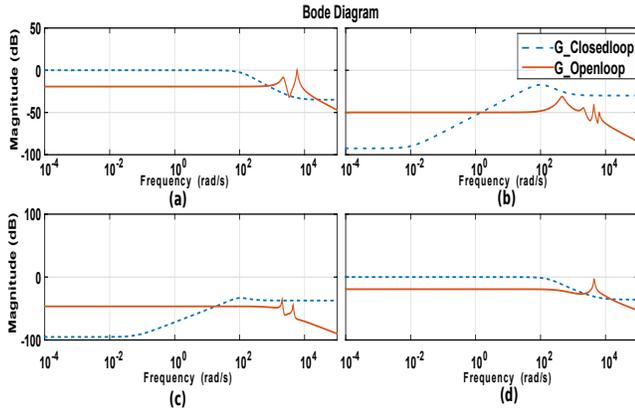


Fig. 11: Open- and closed-loop frequency responses. (a), (b), (c) and (d) for G_{xx} , G_{xy} , G_{yx} , and G_{yy} respectively.

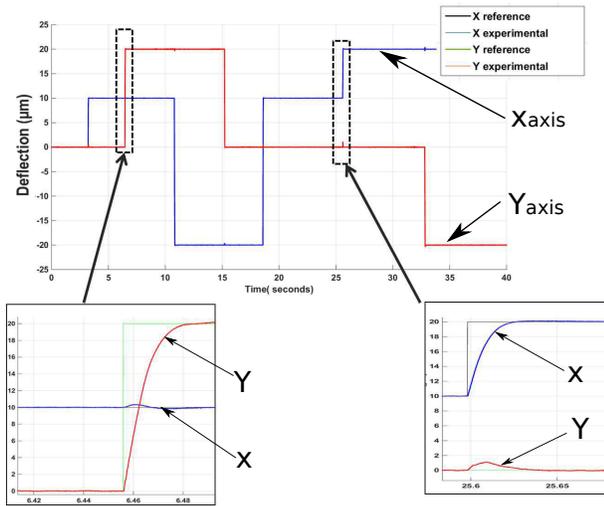


Fig. 12: Deflection response of piezoelectric tube.

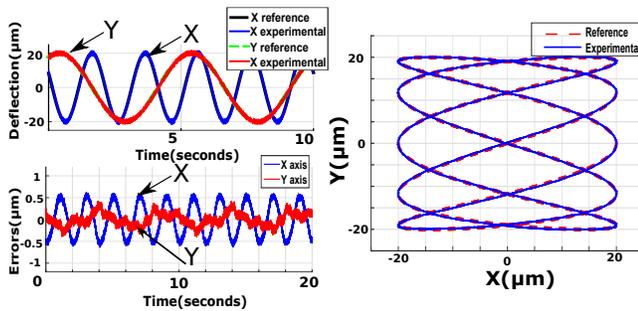


Fig. 13: Helix trajectory tracking

7.3 Controller calculation using LQ tracker design

To demonstrate the advantage of using the LQ tracker design, an experimental comparison between choosing the feedback gains from the solution boxes (fig.7) randomly and by means of LQ tracker design has been considered. As explained in section 5, in order to find the optimal gains that minimize the quadratic cost function (37) we take the solution boxes $[K_{in}]$ from fig.7-a and -b provided by our recursive

SIVIA-based algorithm and we apply the Particle Swarm Optimization (PSO) technique. Indeed the number of vertex matrices used to calculate the quadratic cost function (37) is $2^{12} = 4096$ vertex which makes the use of intelligent optimization technique compulsory. Notwithstanding, the use PSO highly reduce the time of finding the optimal gains. After several iteration of PSO algorithm using the weight factors of 0.5, 1.5 and 1.5 for w , c_1 , and c_2 respectively the optimal gains were obtained at $[-11.155 \ 3.8795 \ 756.7 \ 717.85]$.

Fig.14 shows a comparison between choosing the feedback gains from the solution boxes (fig.7) randomly and by means of LQ tracker design. Actually it can see clearly that all of the step responses of the closed-loop system using different gains satisfy the desire performance with negligible overshoot (1%) and with a settling time $T_s \leq 40ms$. However the optimal gains provide relatively better response. Hence, we can conclude that the obtained robust and optimal output-feedback gains ensure that the closed-loop system is always stable under system uncertainties and simultaneously minimize the quadratic cost function.

Finally, the advantage of the proposed approach is that it provides a global ideas about the variation of the gains and their impact on the system stability which gives the possibility to choose the robust gains easily. Furthermore, with the proposed optimal synthesis it is very easy to find the optimal gains among the robust ones with better understanding of the system functionality.

8 Conclusions

In this paper an algorithm to determine the bounds of the gains of an output-feedback control scheme, with integral compensator, that clusters the closed-loop eigenvalues of interval systems inside a desired region was presented. The algorithm, called recursive SIVIA-based algorithm, is based on the combination of the Set Inversion Via Interval Analysis (SIVIA) approach, the Hladik and Rohn methods of calculation of intervals eigenvalues and the eigenvalues clustering techniques. The proposed recursive SIVIA-based algorithm is used here for robust output-feedback, but it can be used for robust state-feedback for both non-interval and interval systems. Furthermore, an LQ tracker design with Particle Swarm Optimization (PSO) technique is proposed to find the optimal gains that ensure the best behaviors of the closed-loop system and minimize the inputs/outputs energy. Experimental applications were carried out on a piezoelectric tube actuator working in MIMO (Multi Input Multi output) case demonstrated the efficiency of the proposed approach.

Acknowledgements

This work was partially supported by the Labex-ACTION project (ANR-11-LABX-0001-01).

References

- [1] Khadraoui, S., Rakotondrabe, M., and Lutz, P., 2012. "Interval modeling and robust control of piezoelectric

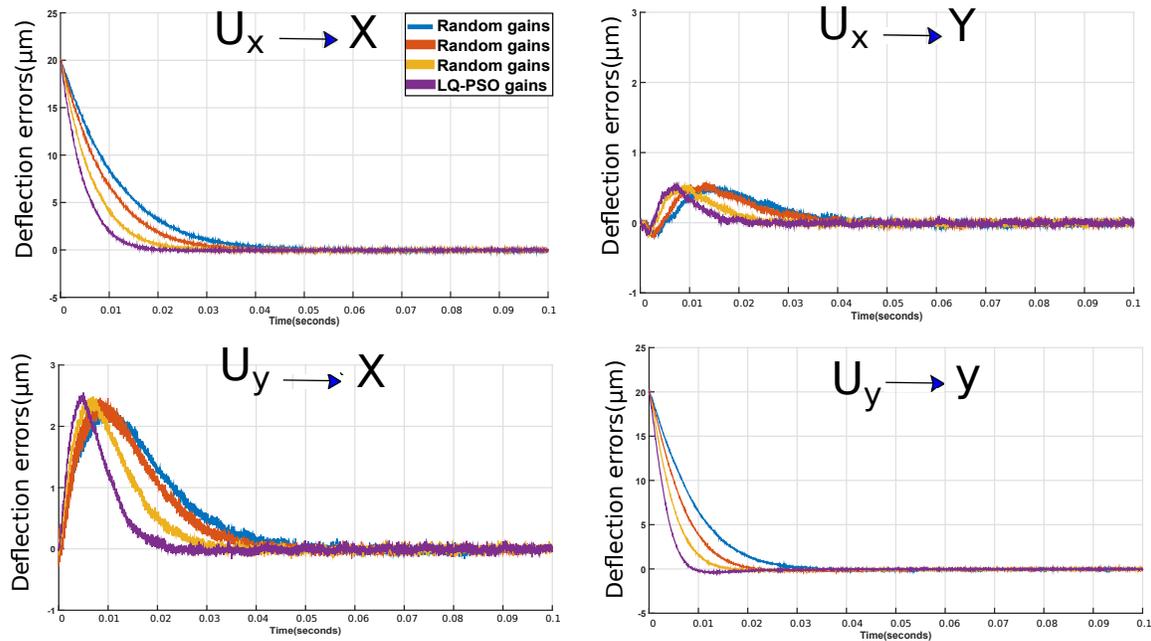


Fig. 14: Step response of piezoelectric tube actuator with different gains values.

- microactuators”. *Control Systems Technology, IEEE Transactions on*, **20**(2), pp. 486–494.
- [2] Khadraoui, S., Rakotondrabe, M., and Lutz, P., 2014. “Interval force/position modeling and control of a microgripper composed of two collaborative piezoelectric actuators and its automation”. *International Journal of Control, Automation and Systems*, **12**(2), pp. 358–371.
- [3] Khadraoui, S., Rakotondrabe, M., and Lutz, P., 2013. “Design of a fixed-order rst controller for interval systems: Application to the control of piezoelectric actuators”. *Asian Journal of Control*, **15**(1), pp. 142–154.
- [4] Smagina, Y., and Brewer, I., 2002. “Using interval arithmetic for robust state feedback design”. *Systems & Control Letters*, **46**(3), pp. 187 – 194.
- [5] Smagina, Y., and Brewer, I., 2000. “Robust modal p and pi regulator synthesis for a plant with interval parameters in the state space”. In American Control Conference, 2000. Proceedings of the 2000, Vol. 2, pp. 1317–1321 vol.2.
- [6] Dugarova, 1989. “Application of interval analysis for the design of the control systems with uncertain parameters.”. PhD thesis, Tomsk State University, Tomsk (In Russian).
- [7] Patre, B. M., and Deore, P. J., 2010. “Robust state feedback for interval systems: An interval analysis approach”. *Reliable Computing*, **14**(1), pp. 46–60.
- [8] Rakotondrabe, M., 2011. “Performances inclusion for stable interval systems”. In American Control Conference (ACC), 2011, IEEE, pp. 4367–4372.
- [9] Khadraoui, S., Rakotondrabe, M., and Lutz, P., 2012. “Combining h-inf approach and interval tools to design a low order and robust controller for systems with parametric uncertainties: application to piezoelectric actuators”. *International Journal of control*, **85**(3), pp. 251–259.
- [10] Khadraoui, S., Rakotondrabe, M., and Lutz, P., 2010. “Robust control for a class of interval model: application to the force control of piezoelectric cantilevers”. In Decision and Control (CDC), 2010 49th IEEE Conference on, IEEE, pp. 4257–4262.
- [11] Kharitonov, V. L., 1978. “Asymptotic stability of an equilibrium position of a family of systems of linear differential equations”.
- [12] Prado, M. L., Lordelo, A. D., and Ferreira, P. A., 2005. “Robust pole assignment by state feedback control using interval analysis”. In World Congress, Vol. 16, pp. 951–951.
- [13] Ordaz, P., and Poznyak, A., 2015. “KI-gain adaptation for attractive ellipsoid method”. *IMA Journal of Mathematical Control and Information*, **32**(3), pp. 447–469.
- [14] Ordaz, P., 2017. “Nonlinear robust output stabilization for mechanical systems based on luenberger-like controller/observer”. *Journal of Dynamic Systems, Measurement, and Control*, **139**(8), p. 084501.
- [15] Haqiri, T., and Poloni, F., 2015. “Methods for verified solutions to continuous-time algebraic riccati equations”. *arXiv preprint arXiv:1509.02015*.
- [16] Miyajima, S., 2015. “Fast verified computation for solutions of continuous-time algebraic riccati equations”. *Japan Journal of Industrial and Applied Mathematics*, **32**(2), pp. 529–544.
- [17] Hashemi, B., and Dehghan, M., 2012. “The interval lyapunov matrix equation: analytical results and an efficient numerical technique for outer estimation of the united solution set”. *Mathematical and Computer Modelling*, **55**(3), pp. 622–633.

- [18] Frommer, A., and Hashemi, B., 2012. “Verified error bounds for solutions of sylvester matrix equations”. *Linear Algebra and its Applications*, **436**(2), pp. 405–420.
- [19] Jaulin, L., and Walter, E., 1993. “Set inversion via interval analysis for nonlinear bounded-error estimation”. *Automatica*, **29**, pp. 1053–1064.
- [20] Jaulin, L., 2001. *Applied interval analysis: with examples in parameter and state estimation, robust control and robotics*, Vol. 1. Springer Science & Business Media.
- [21] Moore, R. E., Kearfott, R. B., and Cloud, M. J., 2009. *Introduction to interval analysis*. SIAM.
- [22] Deif, A., 1991. “The interval eigenvalue problem”. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, **71**(1), pp. 61–64.
- [23] Kolev, L., and Petrakieva, S., 2005. “Assessing the stability of linear time-invariant continuous interval dynamic systems”. *Automatic Control, IEEE Transactions on*, **50**(3), pp. 393–397.
- [24] Hladík, M., 2013. “Bounds on eigenvalues of real and complex interval matrices”. *Applied Mathematics and Computation*, **219**(10), pp. 5584–5591.
- [25] Mayer, G., 1994. “A unified approach to enclosure methods for eigenpairs”. *ZAMM-Journal of Applied Mathematics and Mechanics/Zeitschrift für Angewandte Mathematik und Mechanik*, **74**, pp. 115–128.
- [26] Ahn, H.-S., Moore, K. L., and Chen, Y., 2006. “Monotonic convergent iterative learning controller design based on interval model conversion”. *Automatic Control, IEEE Transactions on*, **51**, pp. 366–371.
- [27] Leng, H., and He, Z., 2017. “Eigenvalue bounds for symmetric matrices with entries in one interval”. *Applied Mathematics and Computation*, **299**, pp. 58–65.
- [28] Rohn, J., 2005. “A handbook of results on interval linear problems”. *Internet text available at <http://www.cs.cas.cz/rohn/handbook>*.
- [29] Bhattacharyya, S., Chapellat, H., and Keel, L., 1995. “Robust control: the parametric approach”. *Upper Saddle River*.
- [30] Hussein, M. T., 2011. “Assessing 3-d uncertain system stability by using matlab convex hull functions”. *International Journal of Advanced Computer Science and Applications, (IJACSA)*, **2**(6).
- [31] Syrmos, V. L., Abdallah, C. T., Dorato, P., and Grigoriadis, K., 1997. “Static output feedback a survey”. *Automatica*, **33**, pp. 125–137.
- [32] Lee, T.-T., and Lee, T.-T., S.-H., 1987. “Root clustering in subregions of the complex plane”. *International journal of systems science*, **18**(1), pp. 117–129.
- [33] Chen, C.-T., 1970. *Introduction to linear system theory*. Holt, Rinehart and Winston.
- [34] Dorf, R. C., and Bishop, R. H., 1998. “Modern control systems”.
- [35] Tymerski, R., and Rytkonen, F., 2012. Control system design.
- [36] Horng, I.-R., Horng, H.-Y., and Chou, J.-H., 1993. “Eigenvalue clustering in subregions of the complex plane for interval dynamic systems”. *International journal of systems science*, **24**, pp. 901–914.
- [37] Gutman, S., and Jury, E. I., 1981. “A general theory for matrix root-clustering in subregions of the complex plane”. *Automatic Control, IEEE Transactions on*, **26**(4), pp. 853–863.
- [38] Kailath, T., 1980. *Linear systems*. Prentice Hall information and system sciences series. Prentice-Hall, Englewood Cliffs.
- [39] Jaulin, L., and Desrochers, B., 2014. “Introduction to the algebra of separators with application to path planning”. *Engineering Applications of Artificial Intelligence*, **33**, pp. 141–147.
- [40] Lewis, F. L., Vrabie, D., and Syrmos, V. L., 2012. *Optimal Control*. John Wiley & Sons.
- [41] Seif, N., Hussein, S., and Deif, A., 1994. “The interval sylvester equation”. *Computing*, **52**(3), pp. 233–244.
- [42] Rohn, J., 1989. “Systems of linear interval equations”. *Linear algebra and its applications*, **126**, pp. 39–78.
- [43] Oettli, W., 1965. “On the solution set of a linear system with inaccurate coefficients”. *Journal of the Society for Industrial and Applied Mathematics, Series B: Numerical Analysis*, **2**(1), pp. 115–118.
- [44] Hassan, R., Cohanin, B., De Weck, O., and Venter, G., 2005. “A comparison of particle swarm optimization and the genetic algorithm”. In Proceedings of the 1st AIAA multidisciplinary design optimization specialist conference, pp. 18–21.
- [45] Rakotondrabe, M., 2013. *Smart Materials Based Actuators at the Micro/Nano-Scale. Characterization, Control and Applications*. SPRINGER - VERLAG, Jan.
- [46] Xie, H., Rakotondrabe, M., and Régnier, S., 2009. “Characterizing piezoscanner hysteresis and creep using optical levers and a reference nanopositioning stage”. *Review of Scientific Instruments*, **80**(4), p. 046102.
- [47] Devasia, S., Eleftheriou, E., and Moheimani, S. O. R., 2007. “A survey of control issues in nanopositioning”. *IEEE Transactions on Control Systems Technology*, **15**(5), Sept, pp. 802–823.
- [48] Agnus, J., Chaillet, N., Clévy, C., Dembélé, S., Gauthier, M., Haddab, Y., Laurent, G., Lutz, P., Piat, N., Rabenorosoa, K., et al., 2013. “Robotic microassembly and micromanipulation at femto-st”. *Journal of Micro-Bio Robotics*, **8**(2), pp. 91–106.
- [49] Juan Antonio Escareno, M. R., and Habineza, D., 2015. “Backstepping-based robust-adaptive control of a nonlinear 2-dof piezoactuator”. *IFAC - Control Engineering Practice*, **41**, pp. 57–71.
- [50] Rakotondrabe, M., Haddab, Y., and Lutz, P., 2009. “Quadrilateral modelling and robust control of a nonlinear piezoelectric cantilever”. *IEEE Transactions on Control Systems Technology*, **17**(3), pp. 528–539.
- [51] Rakotondrabe, M., 2014. “Piezoelectric systems for precise and high dynamic positioning: design, modeling, estimation and control”. *HDR halititation thesis*.
- [52] Ljung, L., 1988. *System identification toolbox: User’s*

guide. Citeseer.

- [53] Premaratne, K., Jury, E., and Mansour, M., 1990. "Multivariable canonical forms for model reduction of 2-d discrete time systems". *IEEE transactions on circuits and systems*, **37**(4), pp. 488–501.
- [54] Skogestad, S., 2007. *Multivariable feedback control: analysis and design*, Vol. 2.

Appendix A: Particle Swarm Optimization(PSO)

Particle Swarm Optimization (PSO) is a heuristic search method that is inspired from the collaborative behavior and swarming in biological populations [44]. It consists of three steps: 1) generation of positions and of velocities of particles, 2) updating velocity, 3) and position update. In our case a particle refers to a point, which represent a feedback gains in the set solution given by the recursive SIVIA-based algorithm, and that changes its position from one move (iteration) to another based on velocity updates.

In the following we denote consequently x_k^i and v_k^i the position and the velocity respectively of the i^{th} particle at time k . First, the positions x_k^i and velocities v_k^i of the initial swarm of particles are randomly generated using upper and lower bounds on the design variables values x_{min} and x_{max} as expressed in (42) and (43). The uniformly distributed random variable (rand) is used on (42) and (43) to allow the swarm particles to be randomly distributed.

$$x_0^i = x_{min} + rand(x_{max} - x_{min}) \quad (42)$$

$$v_0^i = \frac{x_{min} + rand(x_{max} - x_{min})}{\Delta t} \quad (43)$$

The second step is to update the velocities of all particles at time $k + 1$ using the particles fitness or objective values which represent in our case the quadratic cost function (37).

$$v_{k+1}^i = wv_k^i + c_1 rand \frac{(J^i - x_k^i)}{\Delta t} + c_2 rand \frac{(J_k^g - x_k^i)}{\Delta t} \quad (44)$$

where c_1 and c_2 are self and swarm confidence factors consequently, w is inertia factor, J^i is the best position of each particle over time, and J_k^g is the best global value in the current swarm.

Lastly, The Position of each particle is updated in each iteration using its velocity vector as shown in (45).

$$x_{k+1}^i = x_k^i + v_{k+1}^i \Delta t \quad (45)$$

The three steps are repeated until a desired convergence criterion is achieved, in which the maximum change in best fitness J_k^g should be smaller than specified tolerance for a specified number of iterations.