From Impractical LPV Controllers to Practical and "Implementable" LPV Controllers: Verification with Research Airplane [★]

Masayuki Sato [∗]

[∗] *Japan Aerospace Exploration Agency, Mitaka, Tokyo 181-0015, Japan. (e-mail: sato.masayuki@jaxa.jp)*

Abstract: We first give a brief review of classical Gain-Scheduled (GS) controllers with our design example of Stability/Control Augmentation System (S/CAS) for Quad-Tilt-Wing UAV (QTWUAV), and we clarify the issues to be overcome from the S/CAS design example. Then, a brief review of Linear Parameter-Varying (LPV) GS controller design, which has been proposed to overcome the issues in classical GS controller design, is given from the viewpoint of the practicality. Finally, several methods, which have been proposed by the author, are briefly reviewed with some verification results using JAXA's research airplane MuPAL-α. This paper ends with concluding remarks and future research topics related to LPV GS controllers.

Keywords: Gain-Scheduled (GS) controllers, Linear Parameter-Varying (LPV) systems, implementability, flight test.

1. INTRODUCTION

Gain-Scheduled (GS) control technique is well recognized as a control technique for tackling the drastic change of the controlled systems' dynamics. The dynamics change may be caused by the change of the operating conditions, environment, internal states' variations, aging effect, etc. One typical example of such systems is airplane motions. The velocity of airplanes varies from zero to near the sound speed even for commercial jet airplanes, and the maximum speeds for typical jet fighters are far over the sound speed. Although the maximum speed is very low compared to jet airplanes, the dynamics change holds true for JAXA's research airplane MuPAL-α shown in Fig. 1 (Masui and Tsukano, 2000; Sato and Satoh, 2008, 2011). The authorized airspeed for flight tests with research Fly-By-Wire (FBW) system is about [50, 100] [m/s], which is a relatively small deviation compared to jet airplanes; however, the dynamics representing the motions of MuPAL-α change accordingly to the variation of aerodynamic pressure which governs the control device effect to airplane motions. Due to this property, even if very good Linear Time-Invariant (LTI) controllers, which have satisfactory control performance for the supposed flight conditions, are designed, the LTI controllers would be insufficient to control the airplanes for a wide variation of flight condition.

In such a case, other schemes should be invented. One solution is to adopt nonlinear control, and in particular, Nonlinear Dynamic Inversion (NDI) technique (e.g., (Lane and Stengel, 1988)) is a promising method and is well studied in flight control community by considering that the nonlinear systems controlled with NDI can be handled as "fictitious linear systems". The treatment of nonlinear

Fig. 1. MuPAL-α (Masui and Tsukano, 2000)

systems as "fictitious linear systems" is very attractive due to the simple structure of linear systems, and linear control techniques would be then applied combined with other techniques (such as, gain optimization under stochastic uncertainties for aircraft motion coefficients (Kawaguchi et al., 2011)). However, nonlinear control is not yet matured for civil aircraft due to the complexity of control algorithm by considering the difficulty of the authorization process for airworthiness.

Then, it can be seen that GS control is as a good compromise between simple structure but insufficient control performance of LTI controllers and possibly good control performance but complicated structure of nonlinear control. Furthermore, nowadays, linear control theory-based GS control, i.e., Linear Parameter-Varying (LPV) GS control designed using Linear Matrix Inequalities (LMIs), is well studied, and rigorous guarantee for control performance and closed-loop stability is obtained. For these reasons, design methods of LPV GS controllers have been extensively studied in academia since the late 1990's, and several survey papers were published (Leith and Leithead, 2000;
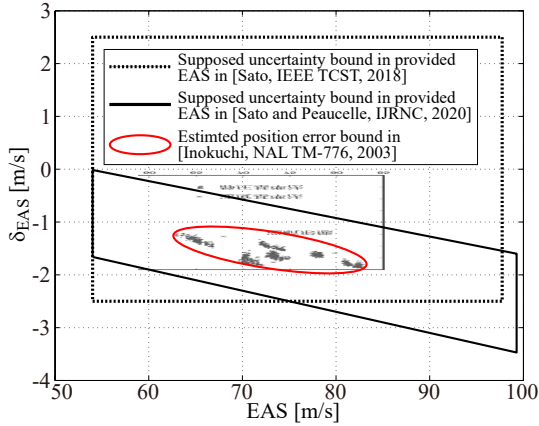
Fig. 2. EAS vs. supposed uncertainty δ_{EAS} in (Sato, 2018; Sato and Peaucelle, 2020) and estimated PE bound from (Inokuchi, 2003) in provided EAS for MuPAL-α

Rugh and Shamma, 2000) about two decades ago, and several books and papers gathering recent developments have been also published (Mohammadpour and Scherer, 2012; Sename et al., 2013; Hoffmann and Werner, 2015; Briat, 2015) in the last decade.

However, from the perspective of the author, most of the papers tackling LPV GS controllers do not consider their controllers' "implementability" so much. That is, when implementing LPV GS controllers to actual systems, several issues arise, such as, online discretization (which can be escaped for Linear Fractional Transformation (LFT)-based LPV GS controllers (Apkarian, 1997)) and the related online numerical burden for matrix inversion calculation (i.e., trapezoidal approximation (Apkarian, 1997)), non-causality of LPV GS controllers depending on the derivatives of scheduling parameters (Scherer, 1996; Wu et al., 1996; Apkarian and Adams, 1998) in Continuous-Time (CT) case and depending on one-step ahead scheduling parameters (Amato et al., 2005) in Discrete-Time (DT) case, and, last but not least, the inexactness of available scheduling parameters. One of the examples of the last issue is the so-called "Position Error" (PE) for Pitot tube in measuring airspeed of airplanes. Airspeed of airplanes is measured using Pitot tube with Bernoulli's principle. However, the pressure measurement is disturbed by the hull of fuselage, which is called as PE. Fig. 2 shows the estimated PE in the provided Equivalent Air Speed (EAS) from FBW system of MuPAL-α, and our supposed uncertainty bounds in LPV GS controller designs in (Sato, 2018; Sato and Peaucelle, 2020). The effect might be small just after airplanes' rollout; however, to estimate the uncertainties in precise way is hard due to possible delay of pressure propagation, and the effect might change accordingly to the period of use of the onboard instrument. Such effect should be considered in LPV GS controller design. Otherwise, actual performance in real environment would be far from the theoretically guaranteed performance in controller design.

Apart from the issues above, we still have several other issues to be tackled further, e.g., the reduction of large complexity due to the large number of parameters required for precise modeling (Hashemi et al., 2012), the implementation of unstable GS controllers to real systems (Balini

et al., 2012), standard toolbox for modeling and controller design (Hjartarson et al., 2015), etc.

Such implementability issues (particularly focusing on causality and inexact scheduling parameters) will be discussed in detail in the following sections.

1.1 Categorization of GS Controllers

Before getting into the detailed discussion of the implementability issues, we would like to categorize GS controllers for easiness of discussion.

We have several types of GS controllers. For example, I heard from one of my previous supervisors in JAXA that *mechanical* GS control technique was used for zero fighters during WWII. The algorithm is described below.

As the airspeed of airplanes increases the aerodynamic pressure accordingly increases. Thus, if the control devices using aerodynamic pressure effect (i.e., aileron, elevator, and rudder) move in the same way regardless of the aerodynamic pressure change, the resultant force driven by such control devices become too large in high aerodynamic pressure condition; that is, pilots feel too active and this might consequently introduce Pilot-Induced Oscillation (PIO) (Pratt, 2000; Stengel, 2004) or Pilot-In-the-loop Oscillation (PIO). One solution to this problem is to use flexible wires to drive the control devices, because the flexibility of the wires adjust the deflection angles of the control devices accordingly to the aerodynamic pressure. That is, in the condition of high aerodynamic pressure, the wires extend due to the increased force coming from high aerodynamic pressure and thus the resultant deflection angles of the control devices reduce; in contrast, in the condition of low aerodynamic pressure, the wires drive the control devices as supposed. Using this mechanical GS control technique, pilots can control the airplanes accordingly to their intentions.

The author was amazed at such a primitive but very effective solution to the change of aerodynamic pressure. Though, such GS control techniques are out of the scope of this paper. Only software-based GS control techniques are reviewed and discussed in this paper, and they are categorized as follows.

Classical GS controllers The plant systems are modeled as nonlinear systems, in which the coefficients of the dynamics are modeled as input-output maps depending on the environment, internal states, etc., and GS controllers are interpolated LTI controllers, which are mutually independently designed for the plant models at *a priori* selected design points. The structure of the LTI controllers, which is usually very simple, e.g., PID controllers, are determined before designing the controller gains. Therefore, the overall control performance heavily depends on the selection of the structure of LTI controllers, the selection of the design points, the interpolation method, the selected scheduling parameters, and so on. A lot of numerical simulations are thus necessary to evaluate the overall control performance using Monte-Carlo method, etc.

Advanced classical GS controllers Classical GS controllers do not consider the so-called "hidden coupling terms" (Rugh and Shamma, 2000). These terms are

caused by the use of endogenous signals as scheduling parameters. "Advanced classical GS controllers" address this issue. One of the effective methods to tackle this issue is to use velocity algorithm (Kaminer et al., 1995; Leith and Leithead, 2000); however, this algorithm needs additional signals, and thus not so practical. Another approach has been recently proposed using structured H_∞ controller design (Lhachemi et al., 2015, 2016) with non-smooth optimization technique (Apkarian and Noll, 2006; Apkarian, 2011). The structure of this type of GS controllers is *a priori* determined as relatively simple structure, such as, PID controllers, then the gains in the structured controllers are optimized directly under the consideration of the effect of hidden coupling terms. The optimization problem is not convex; however, recently developed powerful non-smooth optimization technique gives practical GS controllers.

LPV GS controllers The plant systems are modeled as LPV systems, and the controllers are also designed as LPV systems. Thus, LPV GS controllers designed by using LMIs or parametrically dependent LMIs. However, conservative design caused by the use of Parametrically inDependent Lyapunov Functions (PiDLFs) (Beckar and Packard, 1994; Apkarian et al., 1995) should be admitted, or only conservatism reduction is addressed with use of Parametrically Dependent Lyapunov Functions (PDLFs) (Scherer, 1996; Wu et al., 1996; Apkarian and Adams, 1998; Masubuchi et al., 2004). In other words, the implementability issues are not fully addressed in controller design process.

Practical LPV GS controllers Similarly to LPV GS controllers, the plant systems are modeled as LPV systems and the controllers are also designed as LPV systems. However, the implementability and the practicality are addressed in the controller design phase in exchange for slightly increased conservatism and numerical complexity in controller design; that is, causal LPV GS controller design using PDLFs (Apkarian and Adams, 1998; Köroğlu, 2010; Masubuchi and Kurata, 2011; Sato, 2011; Sato and Peaucelle, 2011; Masubuchi and Yabuki, 2020; Sato and Peaucelle, 2021; Sato, 2021), robustness of LPV GS controllers against the uncertainties in provided scheduling parameters (Ohara et al., 2001; Daafouz et al., 2008; Sato and Peaucelle, 2013; Sadeghzadeh, 2018; Sato and Peaucelle, 2020), and their combinations (Sato and Peaucelle, 2012; Sadeghzadeh, 2018)

Other than the above types, several other types exist, e.g., Youla parametrization-based GS controllers (Matsumura et al., 1996), LFT-based GS controllers (Packard, 1994; Apkarian and Gahinet, 1995; Barker and Balas, 2000; Fialho et al., 2000; Scherer, 2001), the combined method of LFT and LPV GS controller design (Wu and Dong, 2006), switching or blending GS controllers (Shin et al., 2002; Zhao and Nagamune, 2017), LPV GS controllers incorporating parameter estimator, etc. Due to the limited space, they are not discussed in this paper.

The remainder of this paper is structured as follows: Section 2 shows our fundamental idea to design practical and implementable LPV GS controllers; Section 3 gives the review of classical GS controller design process with a flight controller design example for a configuration con-

vertible Unmanned Aerial Vehicle (UAV), which consequently raises the drawbacks and the issues to be tackled; Section 4 gives a brief review of LPV GS controllers, and then give our propositions to design practical LPV GS controllers with a flight controller design example for MuPAL-α. Finally, we give concluding remarks.

All proofs of lemmas and theorems are omitted as they are given in conference or journal papers in the literature.

1.2 Notation

We summarize the notation used in this paper. $\mathbf{0}$ and $\mathbf{I}_n$ respectively denote a zero matrix of compatible dimensions and an $n \times n$ identity matrix; $\mathbb{R}^{n \times m}$ and $\mathbb{S}_+^n$ respectively denote the sets of $n \times m$ real matrices and $n \times n$ positive-definite real matrices; sym in a matrix represents an abbreviated off-diagonal part; $\mathsf{He}\{X\}$ is a shorthand notation of $X + X^T$ for a square matrix X; and $\mathsf{diag}\{X_1, \cdots, X_k\}$ denotes a block diagonal matrix composed of $X_1, \cdots, K_k$.

2. FUNDAMENTAL IDEA IN OUR METHOD FOR IMPLEMENTABILITY

Our methods shown in Section 4 replace a Hermitian term, which causes the implementability issues with respect to (w.r.t.) scheduling parameters, by another term with use of Elimination lemma (Gahinet and Apkarian, 1994; Iwasaki and Skelton, 1994). In other words, a problematic Hermitian term is over-bounded by another term using Elimination lemma. Therefore, we first show our approach for the replacement to circumvent the implementability issues, then give some remarks on conservatism, and finally give the advantage of our approach.

2.1 Elimination Lemma Approach for Over-bounding

The fundamental problem, i.e. the feasibility verification of the following inequality, is now considered:

$$Q_0 - \mathsf{He}\{Q_1 Q_2\} \succ 0, \tag{1}$$

where $Q_0 \in \mathbb{S}_+^n$, $Q_1 \in \mathbb{R}^{n \times l}$, and $Q_2 \in \mathbb{R}^{l \times n}$ are given.

It is obviously possible to verify the feasibility directly; however, we pursue another approach.

Lemma 1. (Sato and Peaucelle, 2020) The followings are both equivalent to (1).

$$\exists R \in \mathbb{R}^{l \times l} \text{ s.t. } \begin{bmatrix} Q_0 & Q_1 \\ \mathsf{sym} & \mathbf{0} \end{bmatrix} + \mathsf{He}\left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_l \end{bmatrix} R [\, Q_2 \ \mathbf{I}_l \,] \right\} \succ 0 \tag{2}$$

$$\exists \tilde{R} \in \mathbb{R}^{l \times l} \text{ s.t. } \begin{bmatrix} Q_0 & \mathsf{sym} \\ Q_2 & \mathbf{0} \end{bmatrix} + \mathsf{He}\left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{I}_l \end{bmatrix} \tilde{R} [\, Q_1^T \ \mathbf{I}_l \,] \right\} \succ 0 \tag{3}$$

Lemma 1 means that verifying (1) is equal to searching R such that $\left(Q_1 + Q_2^T R^T\right) \left(\mathsf{He}\{R\}\right)^{-1} \left(Q_1 + Q_2^T R^T\right)^T \succeq \mathsf{He}\{Q_1 Q_2\}$ holds, i.e., searching an over-bounding term.

2.2 On Conservatism of Over-bounding Methods

We show several similar ideas for over-bounding the term $\mathsf{He}\{Q_1 Q_2\}$ in (1).

One of the most well-established and widely-used methods for the over-bounding is to use square completion for $\mathsf{He}\{Q_1Q_2\}$; that is, $\mathsf{He}\{Q_1Q_2\}$ is over-bounded by $Q_1\mathcal{E}^{-1}Q_1^T + Q_2^T\mathcal{E}Q_2$ with $\mathcal{E} \in \mathbb{S}_+^l$.

Lemma 2. (Xie and de Souza, 1992) If (4) holds then inequality (1) holds.

$$\exists \mathcal{E} \in \mathbb{S}_+^l \text{ s.t. } \begin{bmatrix} Q_0 & Q_1 & Q_2^T\mathcal{E} \\ Q_1^T & \mathcal{E} & \mathbf{0} \\ \mathcal{E}Q_2 & \mathbf{0} & \mathcal{E} \end{bmatrix} \succ 0 \qquad (4)$$

This method is an extension of the method in (Petersen, 1987) from a scalar $\varepsilon \in \mathbb{R}_+$ to a symmetric matrix $\mathcal{E} \in \mathbb{S}_+^l$.

Note that inequality in (4) is equivalent to the following: $Q_0 - Q_1\mathcal{E}^{-1}Q_1^T - Q_2^T\mathcal{E}Q_2 \succ 0$. This inequality confirms that $\mathsf{He}\{Q_1Q_2\}$ is over-bounded by $Q_1\mathcal{E}^{-1}Q_1^T + Q_2^T\mathcal{E}Q_2$.

Another over-bounding has been also proposed.

Lemma 3. C.f. (Sato and Peaucelle, 2012) [1] If one of either conditions (4) or (5) holds, then inequality (1) holds.

$$\exists \mathcal{E} \in \mathbb{S}_+^l \text{ s.t. } \begin{bmatrix} Q_0 & Q_1 + Q_2^T\mathcal{E} \\ Q_1^T + \mathcal{E}Q_2 & \mathcal{E} \end{bmatrix} \succ 0 \qquad (5)$$

As is obvious, Lemma 3 is no more conservative than Lemma 2, since condition (4) is included in Lemma 3. However, both of them are merely sufficient conditions for (1). That is, both conditions are conservative compared to Lemma 1. Then, the following is claimed.

Lemma 4. (Sato and Peaucelle, 2020) If either (4) or (5) holds, then (2) and (3) hold.

In summary, with a slight abuse of mathematical expressions, the following relation holds on the feasibility of (1): Lemma 2 $\subseteq$ Lemma 3 $\subseteq$ Lemma 1 $\Leftrightarrow$ (1).

2.3 Advantage of Elimination Lemma Approach

We would like to clarify the advantage of Lemma 1.

As is shown in (Sato and Peaucelle, 2020, 2021), the corresponding term of Q_0 in practical LPV GS controller design is affine w.r.t. decision matrices; however, both Q_1 and Q_2 contain decision matrices and thus the term Q_1Q_2 is bilinear w.r.t. decision matrices.

On this bilinear term, the following two approaches are widely used. In particular, the former one has been adopted from μ-synthesis (e.g., (Balas et al., 1998)) and constant-scaled H_∞ controller design.

Iterative algorithm: Set some decision matrices in Q_1 or Q_2 *completely fixed* with *a priori* defined ones to remove the multiplications of decision matrices and solve (1) with the remaining decision matrices, then set the other decision matrices comprising the multiplications of decision matrices *completely fixed* with the optimized ones in the previous step and solve (1) with the remaining decision matrices. Iterate the above two steps until the feasibility is confirmed or the supposed maximum iteration number is reached.

Line search algorithm: Set some decision matrices in Q_1 or Q_2 *partially fixed* using line search parameters

[1] A slightly different version with $Q_1^T = [\Upsilon_1 \ \mathbf{0}]$ and $Q_2 = [\mathbf{0} \ \Upsilon_2]$ is shown in (Sato and Peaucelle, 2012).

Fig. 3. JAXA's QTWUAV "AKITSU"

Table 1. Dimensions of AKITSU

Length	1750 [mm]	Span	2377 [mm]
Height	720 [mm]	Cruise speed	50 [m/s]
Gross weight	43.2 [kg]	Electric motors	4×7000 [W]

to remove the multiplications of decision matrices, and solve (1) with the remaining decision matrices for the prefixed variations of the line search parameters.

However, both algorithms impose severe constraints to some of decision matrices, which is a sharp contrast to Lemma 1; that is, such constraints on the original decision matrices in Q_1 and Q_2 are totally removed in Lemma 1. In other words, even if the new decision matrices R and $\tilde{R}$ are set as $r\mathbf{I}$ using a line search parameter r, the original decision matrices in Q_1 and Q_2 have no structural constraints imposed. While this remedy obviously imposes a structural constraint on the auxiliary matrices R and $\tilde{R}$, the original decision matrices can escape from structural constraints, which is the main advantage of Lemma 1.

Nevertheless, we would also like to emphasize that the equivalence between the feasibility of (1) and the condition in Lemma 1 holds if and only if the auxiliary matrices R and $\tilde{R}$ are set as *totally free full* matrices. In this sense, searching decision matrices in Q_1 and Q_2, and R or $\tilde{R}$ cannot escape from the bilinear property in (1).

3. REVIEW OF CLASSICAL GS CONTROLLER DESIGN WITH A DESIGN EXAMPLE

Classical GS controller design technique has already been presented in many papers and books. In particular, as mentioned above, flight control is one of the most beneficiaries of the advantages of classical GS controllers (Stevens and Lewis, 1992).

In this section, we show a design example of classical GS controllers for Quad Tilt Wing (QTW) UAV (Sato and Muraoka, 2015), whose configuration change is shown in Fig. 3 and dimensions are given in Table 1, in order to raise the drawbacks and the difficulties in controller design.

Our addressed problem is Stability/Control Augmentation System (S/CAS) design for the QTWUAV. Usually, S/CAS is designed for the linearized longitudinal and lateral-directional motions independently, which is the same as for our design. The objective of SAS is to stabilize the aircraft motion as much as possible (stabilization
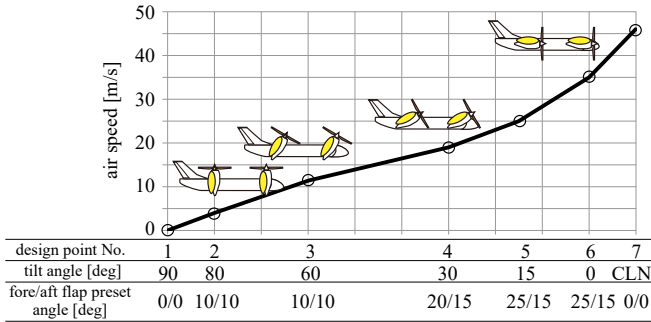
Fig. 4. Preset tilt angle schedule vs. trim airspeed at straight level flight

cannot be always realized), and that of CAS is to realize good tracking performance for attitude commands (pitch angle θ and roll angle ϕ) given by a remote pilot.

Although the configuration of QTWUAV is neither a conventional airplane nor a conventional rotorcraft, the relation between control input and the resultant motion is the same; that is, control devices create rotational moment to control QTWUAV; that is, flaps mounted at the trailing edge of each wing are used as aileron to control roll-axis motion and as elevator to control pitch-axis motion in airplane mode, but also used as rudder to control yaw-axis motion in helicopter mode, and thrust given by propellers mounted at the leading edge of each wing is used as elevator to control pitch-axis motion and as aileron to control roll-axis motion via differential thrust in helicopter mode, but also used as rudder to assist the conventional rudder to control yaw axis motion in airplane mode.

Therefore, we follow the conventional design procedure for classical GS controller design; that is, we follow the following steps.

(1) Select the scheduling parameter(s), and select design points for individual controller design.
(2) Design LTI controllers at the selected design points.
(3) Obtain a GS controller with use of (linear) interpolation of the designed LTI controllers.
(4) Examine overall control performance for the whole parameter region with numerical simulations.
(5) Examine overall actual control performance under real environment.

The details at each step are given below.

At the first step, since QTWUAV dynamics change accordingly to the tilt angles, we select the tilt angle as the scheduling parameter. This choice is reasonable and does not introduce any confusion among engineers. We next select seven design points as in Fig. 4 by considering a variety of issues; that is, if we choose so many design points, then the remote pilot may be confused of the current tilt angles and may make a mistake for controlling the aircraft; however, if we choose small numbers of design points, then suitable GS controllers might not be designed because the dynamics changes between design points are too large. We also have to consider the hardware limitation, because the remote radio control console has limited numbers of servo inputs. Thus, in the author's understanding, the choice of design points might be a source of many "trial-and-errors".

At the second step, we set PI-D controller as S/CAS, which is a conventional and typical controller structure for controlling aircraft. To design PI-D controllers at the selected design points, we first design SAS with only attitude rate-feedback control (i.e., D controller design), then design CAS using the difference between attitude commands and current attitude angles (i.e., PI controller design). In this S/CAS design, multiple model approach (Ackermann, 1985) is used to obtain robust controllers; that is, the worst performance of selected performance indices (the maximum real part of the closed-loop poles in SAS design and tracking error for step attitude command in CAS design) among the selected nominal and perturbed models are minimized at each design point. This kind of "practical robustification" has been well demonstrated in flight control community (Miyazawa, 1992; Ohno et al., 1999) as well as control community (de Aguiar et al., 2018). Thus, in a sense, this step does not contain the possible source of many "trial-and-errors"; however, in general, we cannot escape from "many trial-and-errors" to set appropriate performance indices with appropriate design specifications using weighting functions, design constraints, etc.

At the third step, we implement the designed PI-D controllers with linear interpolation of the obtained gains in the previous step. It might be possible to use second-order or higher-order interpolations; however, this can be approximately achieved by increasing the number of design points. Therefore, this step does not contain the possible source of many "trial-and-errors".

At the fourth step, we can examine the overall control performance for all possible variations of scheduling parameters by numerical simulations, in which aircraft dynamics are represented by nonlinear equations with many maps representing coefficients related to aircraft motions, under pilot control input. In this numerical simulations, a lot of factors effecting maneuverability should be incorporated, e.g., wind gust, modeling errors (including aircraft dynamics as well as onboard actuators), delay due to implemented hardware, etc. If the overall control performance is satisfactory, we move on to the final step.

At last, we examine the overall control performance under real environment, i.e., flight tests. However, even if all the other steps are passed without iterative trial-and-errors, we still have a possibility which may make us design from the scratch. In our case, even if pilot says "the maneuverability is good and acceptable" in numerical simulations, he sometimes says "the motion is sluggish and maneuverability is not good" in flight tests. If we obtain such comments from the pilot, we examine his words and make a guess what he means, as he is not so familiar with control technique. Then, we usually go back to the second step, and conduct re-design of PI-D gains with slight revisions of control specifications, e.g., revisions of overshoot constraints, settling time, acceptable region of closed-loop poles, etc. After many trial-and-errors of controller design (over 25 times only for major revisions), several unexpected accidents, and one crash, we finally realize safe full conversion flight shown in Fig. 5.

In Fig. 5, the transition intervals between two adjacent design points are hatched with gray color. Most of the transitions do not introduce oscillatory motions; however,
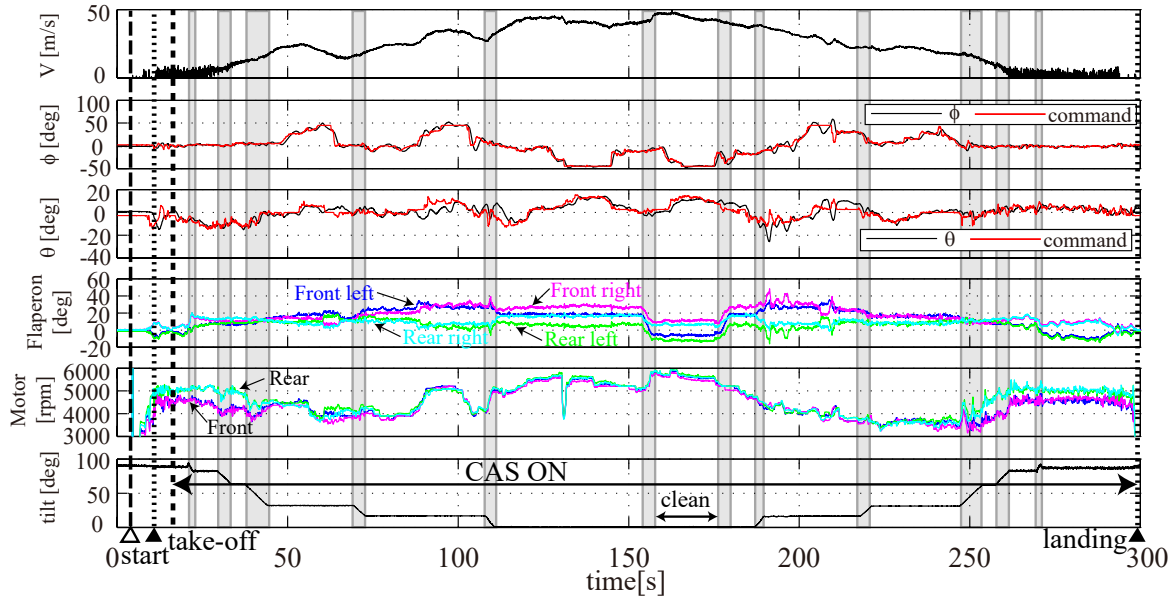
Fig. 5. Flight test result (gray intervals represent transition intervals moving between adjacent design points)

some transitions does, e.g., from tilt angle 15 [deg] to 0 [deg] around 110 [s] in pitch angle (θ), from clean configuration to tilt angle 0 [deg] around 175 [s] in roll angle (ϕ), and from tilt angle 15 [deg] to 30 [deg] around 220 [s] in pitch angle (θ). We also have a large oscillation in pitch angle (θ) at tilt angle 15 [deg] around 180 [s] We conduct many revisions for PI-D gains (as mentioned above, over 25 times only for major revisions) in accordance with pilot comments; however, the maximum achievable control performance is not perfectly satisfactory for the pilot.

We learn that classical GS controllers have a good potential to control well-established dynamical systems for several reasons. First of all, the structure of GS controllers can be determined reasonably by considering the plant dynamics; secondly, the determined structure is, in general, relatively simple compared (e.g., PID, PI-D controllers) to full-order H_∞ controllers; thirdly, such simple structure is easily implemented to hardware systems, because discretization can be conducted beforehand or even online discretization is possible thanks to scalar divisions instead of matrix inversions; fourthly, this simple structure neither introduces severe numerical burden to computers nor severe confusion to engineers in controller re-design even if on-site re-design should be conducted; and finally the well-established GS controller technique has a large potential of applicability even for newly developed systems as long as the governing dynamics of the systems are similar.

However, as mentioned above, we cannot escape from many "trial-and-errors", which are mainly from the fact that we cannot obtain the overall control performance at the individual controller design in step (2). Furthermore, if the overall performance at steps (4) and (5) is not satisfactory, which step should we go back to? The answer to this question is not obvious. One solution to these issues is to design GS controllers which guarantee the overall control performance for all possible parameter variations in controller design phase. If the plant systems are modeled as LPV systems, then, LPV GS controllers designed using parametrically affine LMIs guarantee the overall control performance, which is obtained in controller design phase

(e.g., induced L_2/l_2 norm, etc), for all possible parameter variations due to the convexity of parametrically affine LMIs w.r.t. the related parameters (Apkarian et al., 1995). Thus, many theoretical researchers tackle LPV GS controller design using LMIs.

4. REVIEW OF LPV GS CONTROLLER AND PRACTICAL LPV GS CONTROLLER DESIGNS

4.1 Review of LPV GS Controller Design

After the paper which addresses LPV GS controller design using PiDLFs (Apkarian et al., 1995) is published, many researchers have started to tackle LPV GS controller design in LMI framework. At first, PiDLFs are used for simplicity; however, design methods of LPV GS controllers using PDLFs have been invented (Scherer, 1996; Wu et al., 1996; Apkarian and Adams, 1998; Amato et al., 2005) to reduce the conservatism, because PiDLFs cannot consider the bounded parameter variation rate. In this aspect, the research has advanced; however, in another aspect, i.e., from the viewpoint of implementability, the progress is questionable at least to the author. This is because LPV GS controllers designed using PDLFs become, in general, impractical for their implementation; that is, non-causal LPV GS controllers are obtained, *viz.*, LPV GS controllers require the derivatives of scheduling parameters in CT case and the one-step ahead scheduling parameters in DT case.

On this issue, several methods have already been proposed to escape from the non-causality of LPV GS controllers, e.g., LPV GS controller design using partially PDLFs (Apkarian and Adams, 1998), additional filters for scheduling parameters (Masubuchi and Kurata, 2011), and over-bounding approach (shown in Section 2) for problematic terms (Köroğlu, 2010; Sato, 2011; Sato and Peaucelle, 2011, 2021; Sato, 2021), and the use of scheduling parameters in the previous steps (Masubuchi and Yabuki, 2020) and constant auxiliary matrices to avoid future scheduling parameters (de Caigny et al., 2012). The last two methods are only for the DT case.

4.2 Review of Practical LPV GS Controller Design Using Over-Bounding Approach

We now focus on our proposed methods to design practical LPV GS controllers using over-bounding approach shown in Section 2. As mentioned in Section 2, even if over-bounding approach is adopted, the derived conditions are still Bilinear Matrix Inequalities (BMIs). Thus, to solve them, the combined method of line search and iterative algorithm is adopted in (Sato and Peaucelle, 2020, 2021).

Causal LPV GS Controller Design Using PDLFs The cause of deriving non-causal LPV GS controllers is that, in the process of change-of-variables, decision matrices contain the derivatives of a part of PDLFs. Thus, to derive causal LPV GS controllers, we only have to conduct change-of-variables without including the derivatives of PDLFs. To this end, the following re-formulation, in which $\mathcal{Z}(\theta)$ represents a symmetric sub-block of parametrically dependent Lyapunov matrix, is first adopted, and then change-of-variables is conducted (Sato and Peaucelle, 2021).

$$-\begin{bmatrix} \mathbf{0} & \text{sym} \\ \dot{\mathcal{Z}}(\theta)\mathcal{Z}(\theta)^{-1} & \mathbf{0} \end{bmatrix} = -\mathsf{He}\left\{ \begin{bmatrix} \mathbf{0} \\ \dot{\mathcal{Z}}(\theta) \end{bmatrix} \begin{bmatrix} \mathcal{Z}(\theta)^{-1} & \mathbf{0} \end{bmatrix} \right\} \tag{6}$$

$$\begin{bmatrix} \mathbf{0} & \mathcal{Z}(\theta^+)^{-1}\mathcal{Z}(\theta) \\ \text{sym} & \mathcal{Z}(\theta)\mathcal{Z}(\theta^+)^{-1}\mathcal{Z}(\theta) \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{I} & \mathcal{Z}(\theta) \end{bmatrix}$$
$$-\mathsf{He}\left\{ \begin{bmatrix} \mathbf{0} \\ \mathcal{Z}(\theta^+) - \mathcal{Z}(\theta) \end{bmatrix} \begin{bmatrix} \mathcal{Z}(\theta^+)^{-1} & \frac{1}{2}\mathcal{Z}(\theta^+)^{-1}\mathcal{Z}(\theta) \end{bmatrix} \right\} \tag{7}$$

Equation (6) is for the CT case, and setting $Q_1 = \begin{bmatrix} \mathbf{0} & \dot{\mathcal{Z}}(\theta) \end{bmatrix}^T$ and $Q_2 = \begin{bmatrix} \mathcal{Z}(\theta)^{-1} & \mathbf{0} \end{bmatrix}$ directly leads to the application of the over-bounding method in Section 2.

In contrast, equation (7), which is for the DT case, is slightly tricky. However, this treatment with setting $Q_1 = \begin{bmatrix} \mathbf{0} & \mathcal{Z}(\theta^+) - \mathcal{Z}(\theta) \end{bmatrix}^T$ and $Q_2 = \begin{bmatrix} \mathcal{Z}(\theta^+)^{-1} & \frac{1}{2}\mathcal{Z}(\theta^+)^{-1}\mathcal{Z}(\theta) \end{bmatrix}$ directly leads to the application of the over-bounding method in Section 2, and it also recovers the design method using PiDLFs similarly to the CT case.

We also would like to emphasize that when using the dual formulation of (Sato and Peaucelle, 2021), the counterpart formulation in the CT case is similarly derived; however, the counterpart formulation in the DT case cannot be obtained, and LPV GS controllers depending on the one-step ahead scheduling parameters are derived (Sato, 2021).

LPV GS Controllers Depending on Inexact Scheduling Parameters In this case, PDLFs are used; however, by following the method in (de Caigny et al., 2012), constant auxiliary matrix is used to design parametrically multi-affine LPV GS controllers. This setup generally introduces conservatism. However, it also increases the implementability of the designed LPV GS controllers, since there is no matrix inversion calculation online. This property is important from industry's perspective.

In the CT and DT cases, the difficulty of designing LPV GS controllers depending on inexact scheduling parameters is caused again by change-of-variables. That is, decision matrices after change-of-variables should depend only on the provided scheduling parameters; however, the LPV plant model is governed by exact scheduling parameters. In other words, we need to simultaneously design LPV plant models, which is governed by exact scheduling parameters but should be amended to provide inexact scheduling parameters to LPV GS controllers, and LPV GS controllers depending only on inexact scheduling parameters.

In both the CT and DT cases, the problematic term is treated as follows:

$$\begin{bmatrix} \begin{bmatrix} \mathbf{0} \\ \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathcal{Y}\left(A(\theta) - A(\hat{\theta})\right)\mathcal{X} & \mathbf{0} \end{bmatrix} \end{bmatrix} & \begin{matrix} \text{sym} \\ \mathbf{0} \end{matrix} \end{bmatrix}$$
$$= \mathsf{He}\left\{ \begin{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathcal{Y}\left(A(\theta) - A(\hat{\theta})\right) \end{bmatrix} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} \mathcal{X} & \mathbf{0} \end{bmatrix} & \mathbf{0} \end{bmatrix} \right\}$$
$$= \mathsf{He}\left\{ \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathcal{Y} \end{bmatrix} \begin{bmatrix} \begin{bmatrix} \left(A(\theta) - A(\hat{\theta})\right)\mathcal{X} & \mathbf{0} \end{bmatrix} & \mathbf{0} \end{bmatrix} \right\} \tag{8}$$

Here, $\mathcal{X}$ and $\mathcal{Y}$ are constant auxiliary decision matrices, $A(\theta)$ represents the state transition matrix of LPV plant model with actual parameter vector θ, and $\hat{\theta}$ represents the provided inexact scheduling parameter vector. Using the formulation in (8) removes the term containing both the actual exact and the provided inexact scheduling parameters in change-of-variables process.

In this case, we have two choices for Q_1 and Q_2 in equation (1). That is, one choice is $Q_1 = \begin{bmatrix} \mathbf{0} & \left(A(\theta) - A(\hat{\theta})\right)^T \mathcal{Y}^T \end{bmatrix}^T$ and $Q_2 = \begin{bmatrix} \mathcal{X} & \mathbf{0} \end{bmatrix}$, and the other is $Q_1 = \begin{bmatrix} \mathbf{0} & \mathcal{Y}^T \end{bmatrix}^T$ and $Q_2 = \begin{bmatrix} \left(A(\theta) - A(\hat{\theta})\right)\mathcal{X} & \mathbf{0} \end{bmatrix}$. Unfortunately, we have not yet clarified which is more conservative than the other. We thus need to solve the design problem with two formulations for better LPV GS controllers.

However, using the formulation in (8), we can design LPV GS controllers depending on inexactly provided scheduling parameters by applying Lemma 1 in Section 2.

4.3 Flight Controller Design Example

Using the design method in (Sato, 2018; Sato and Peaucelle, 2020) for LPV GS controllers depending on inexact scheduling parameters, we also design flight controllers. In those papers, DT LPV plant model, which is derived from CT LPV plant model, is set for controller design to escape from the numerical complexity of online discretization and the related online matrix inversion issue. The block diagram for the design is shown in Fig. 6. We give a brief explanation of blocks and signals therein.

Blocks: $G_{L/D}(V_{true})$, $G_{act}(V_{true})$, G_{dex} and $K(V_{prov})$ respectively denote the lateral-directional motions of MuPAL-α scheduled by true EAS which is denoted by V_{true}, the actuator dynamics scheduled by V_{true}, one step delay model and the to-be-designed LPV GS controller scheduled by provided EAS which is denoted by $V_{prov}(= V_{true} + \delta_{EAS})$ with the uncertainty in the provided EAS data δ_{EAS} in Fig. 2; W, W_g and W_M respectively denote the weighting functions for uncertainties related to the on-board actuators, gust suppression performance and model-matching performance; $\Delta = \mathsf{diag}\{\Delta_{per}, \Delta_{gust}, \Delta_a, \Delta_r\}$
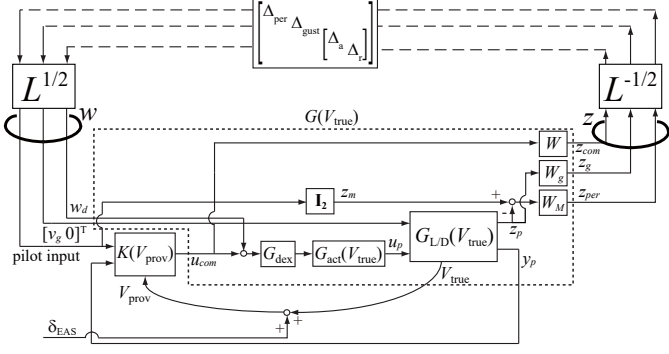
Fig. 6. Block diagram for simultaneous realization of model-matching and gust suppression in (Sato, 2018; Sato and Peaucelle, 2020)

denotes the structured uncertainty block composed of 2×2-dimensional model-matching performance block (Δ_{per}), 2×2-dimensional gust suppression performance block (Δ_{gust}), and two scalar uncertainty blocks to represent the uncertainties related to the onboard aileron and rudder actuators (Δ_a and Δ_r); and $L^{1/2}$ and its inverse denote the constant scaling matrix to reduce conservatism due to the structured uncertainty block Δ. It is supposed that $\|\Delta\| \leq 1$ holds.

Signals: u_{com}, u_p, z_p, y_p and δ_{EAS} respectively denote the control command produced by $K(V_{prov})$, i.e., $u_{com} = [\delta_{a_c} \; \delta_{r_c}]^T$ (δ_{a_c} and δ_{r_c} respectively denote aileron and rudder deflection commands), actual control input to MuPAL-α, i.e., $u_p = [\delta_a \; \delta_r]^T$ (δ_a and δ_r respectively denote aileron and rudder deflections), plant output to be controlled, i.e., $z_p = [v_a \; \phi]^T$ (v_a and ϕ respectively denote lateral airspeed and roll angle), measurement output, i.e., $y_p = [v_a \; p \; \phi \; r]^T$ (p and r respectively denote roll and yaw rates) and uncertainties in the provided EAS; the pair of w_d and z_{com} denotes the signals to evaluate the uncertainties related to the onboard actuators, *viz.*, they respectively denote fictitious external input and weighted u_{com} to compensate the uncertainties of the onboard actuators; the pair of $[v_g \; 0]^T$ and z_g denotes the signals to evaluate gust suppression performance, *viz.*, v_g and z_g respectively denote sideway gust and performance output z_p multiplied by the weighting function W_g; and the pair of z_m, which is the output of $\mathbf{I}_2$ driven by "pilot input", and z_{per} denotes the signals to evaluate model-matching performance, *viz.*, they respectively denote the model output which is to be reproduced by MuPAL-α and performance output z_p multiplied by the weighting function W_M. Note that v_g is augmented as $[v_g \; 0]^T$ to comply with the size of z_g.

The designed LPV GS controller is implemented to the FBW system of MuPAL-α, and control performance comparison between an LPV GS controller, which is designed using Lemma 3 to robustify LPV GS controller against the uncertainties in the provided scheduling parameters, and an LPV GS controller, which is designed using Lemma 1 to robustify LPV GS controller against the uncertainties in the provided scheduling parameters, are conducted. The control performance improvement of the latter LPV GS controller compared to the former LPV GS controller is confirmed but not so large, since both are designed to be robust against the uncertainties in the provided EAS.
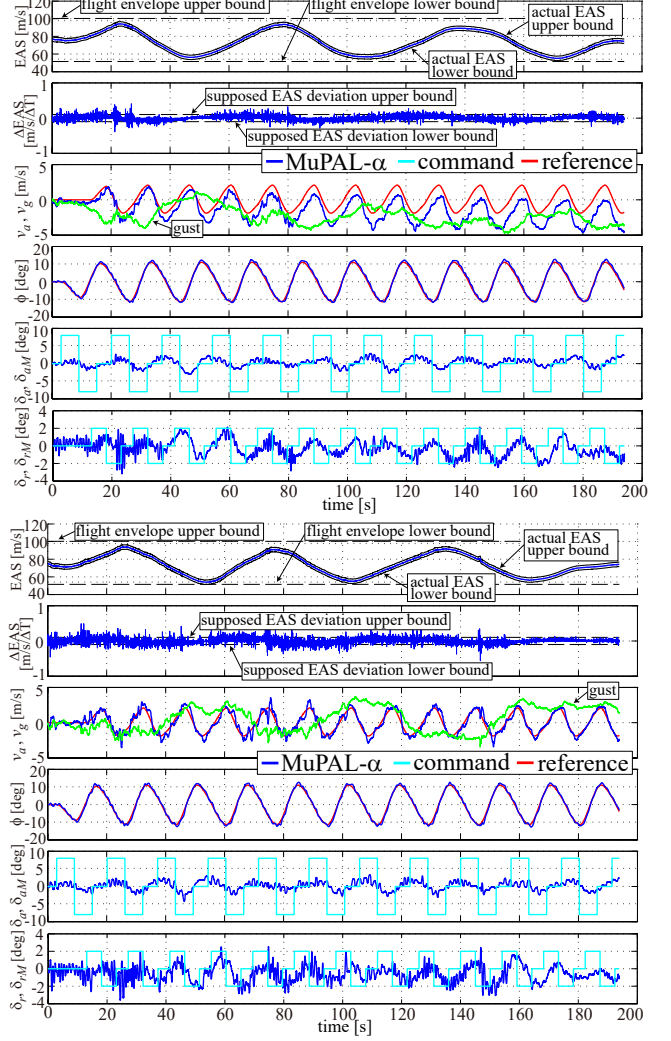

Fig. 7. Flight test results using non-robust LPV GS controller (top) and robust LPV GS controller (bottom) in (Sato, 2018)

On the other hand, control performance improvement of an LPV GS controller, which is designed to be robust against the uncertainties in the provided scheduling parameters, compared to an LPV GS controller, which is designed without consideration of the uncertainties in the provided scheduling parameters, is interesting as clearly demonstrated in (Sato, 2018). The sideway airspeed v_a of MuPAL-α controlled by the latter LPV GS controller cannot follow the reference and has a bias error as shown in the the third rows in the top figures in Figs. 7 and 8. In a sharp contrast, the sideway airspeed v_a and roll angle ϕ are both well controlled by the former LPV GS controller as shown in bottom figures in Figs. 7 and 8. This performance improvement clearly illustrates that robustness against the uncertainties in the provided scheduling parameters should be considered to design practical LPV GS controllers.

5. CONCLUDING REMARKS AND FUTURE RESEARCH TOPICS

We review the design method of classical GS controllers with a design example of flight controllers for QTWUAV, and confirm that the most laborious work is many "trial-and-errors" in controller design. To reduce "trial-and-
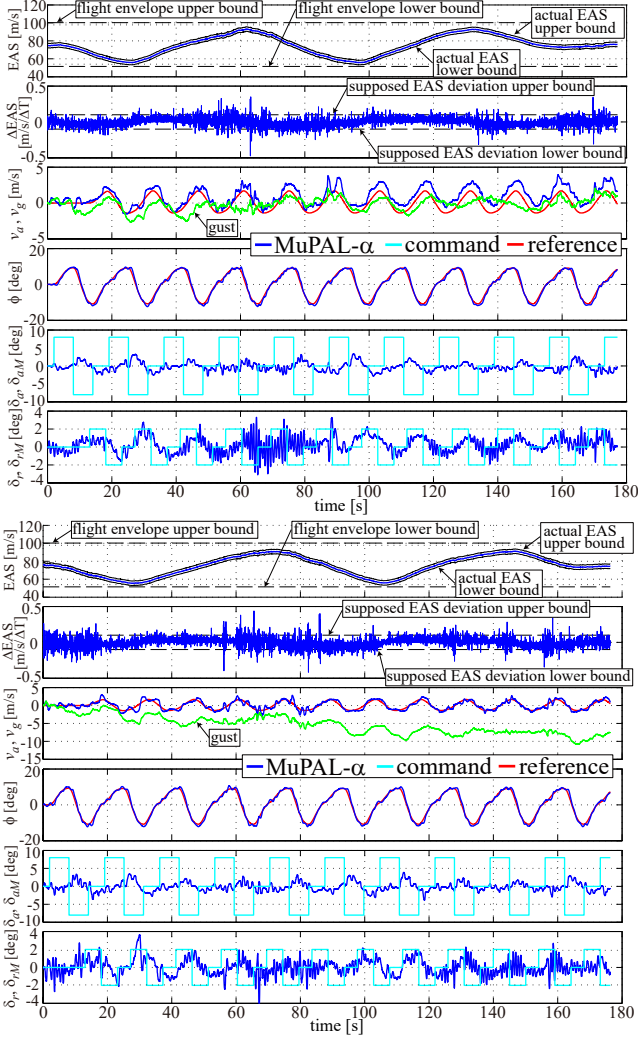
Fig. 8. Flight test results using non-robust LPV GS controller (top) and robust LPV GS controller (bottom) in (Sato, 2018)

errors", LPV GS controller design is then addressed, because the overall control performance is rigorously evaluated in controller design phase. However, in exchange for conservatism reduction, "implementbility" is deteriorated by using Parametrically Dependent Lyapunov Functions (PDLFs), and practicality (e.g. inexactness of provided scheduling parameters) is not well considered in controller design. On these critical issues for implementation, our proposed methods are reviewed with flight controller design example for JAXA's research airplane MuPAL-α.

Although practical and implementable LPV GS controllers can be designed using the currently available methods, the maturity is not sufficient and further research is still necessary, which is discussed in the following.

5.1 Verification with Practical Systems

As the survey paper (Hoffmann and Werner, 2015) indicates, the application of LPV GS controllers to real systems has been increasing even for space satellite (Hamada et al., 2011); however, it seems to be not yet matured compared to H_∞ controllers. There are several reasons for this; that is, although some toolbox tackling modeling and design have already been proposed (Hjartarson et al., 2015),

globally standard one has not yet come, implementability of the designed LPV GS controllers is not matured since online inversion calculations for parameter-dependent matrices are required, however, such possibly problematic calculations cannot be accepted in industry, etc. Although some progress has been seen using Weierstrass approximation theorem (Bliman, 2004; Sato and Peaucelle, 2012), the maturity is not enough for airworthiness certification. Thus, from the viewpoint of practical system applications, theoretical development is still required. On the other hand, it is believed that theoretical development increases the applications of LPV GS controllers to real systems.

5.2 Theoretical Research

As mentioned above, polynomially parameter-dependent LPV GS controllers are much more acceptable than rationally parameter-dependent LPV GS controllers. This kind of research is important to enhance the applicability of LPV GS controllers. Thus, theoretical research development for improving the implementability of LPV GS controllers is still one of the future research topics.

Similarly, to improve the implementability of LPV GS controllers, designing low-dimensional LPV GS controllers is also important, similarly reduced-order H_∞ controllers.

On the other hand, there is a possibility to exploit the high-dimensions of full-order LPV GS controllers to embed property to the designed LPV GS controllers. Accordingly to this idea, we have been studying to embed observer property to *a priori* designed LPV GS controllers without changing control performance (Sato and Sebe, 2022; Sato and Sebe). If observer property is successfully embedded to existing LPV GS controllers, then the estimated state can be used as plant health monitoring, fault detection, fault information, etc.

One example is briefly given below. We embed observer property of the control device states (i.e., aileron and rudder deflection angles) to the LPV GS controller in (Sato and Peaucelle, 2020), and we conduct numerical simulations using the recorded EAS and wind gust. The results are shown in Fig. 9. It confirms that it is possible to embed observer property to *a priori* designed LPV GS controllers. In particular, although the deflection angles are not large in the bottom figure in Fig 9, the converted controller can well estimate the control device states.

Such kinds of theoretical research would be appreciated from the viewpoint of the enhancement of practicality.

5.3 Extension Together with Other Control Technique

As mentioned above, we still have research topics which should be addressed; however, the basic design method for LPV GS controllers is well established. Therefore, LPV GS control method combined with other techniques becomes a good candidate for further good control performance. For example, LPV sliding mode control is proven to be a powerful candidate for Fault Tolerant Control (FTC), as demonstrated with MuPAL-α (Chen et al., 2020) in Europe-Japan international joint research project entitled "Validation of Integrated Safety-enhanced Intelligent flight cONtrol (VISION)" (Sato et al., 2018). Such kind of
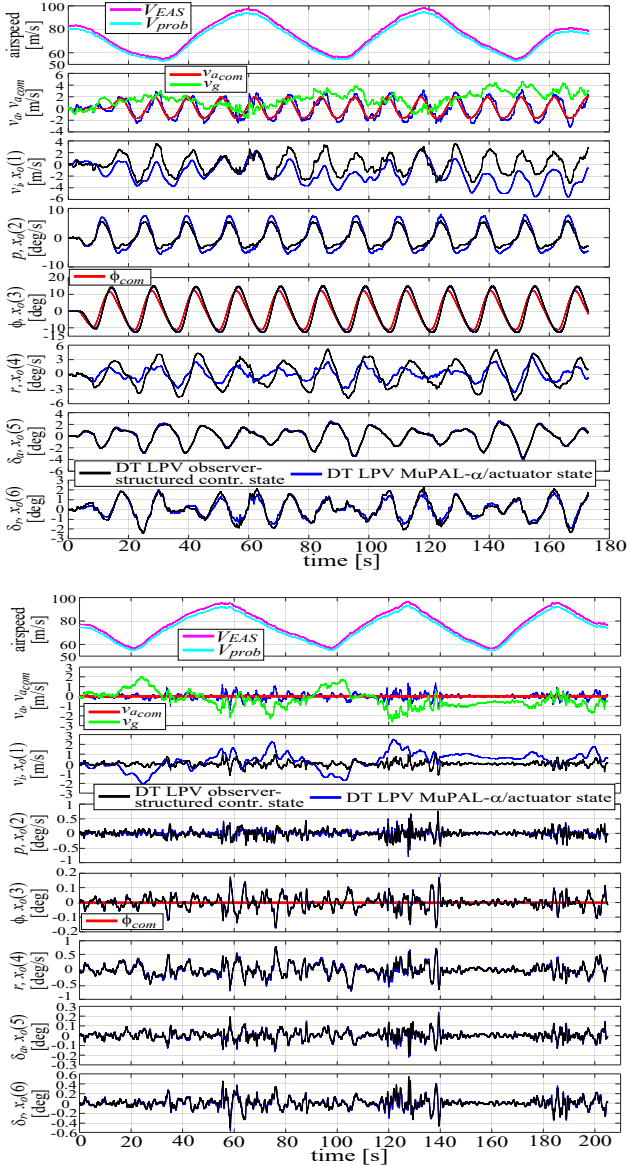
Fig. 9. Simulation results using LPV controller converted from the LPV controller in (Sato and Peaucelle, 2020) using EAS and wind gust in Fig. 6 (top) and Fig. 4 (bottom) in (Sato and Peaucelle, 2020)

extension together with other control techniques might be a breakthrough to the current control performance limit.

ACKNOWLEDGEMENTS

REFERENCES

Ackermann, J. (1985). *Uncertainty and Control*, volume 70 of *Lecture Notes in Control and Information Sciences*, chapter Multi-Model Approaches to Robust Control System Design, 108–130. Springer-Verlag, Berlin Heidelberg. doi:10.1007/BFb0007278.

Amato, F., Mattei, M., and Pironti, A. (2005). Gain scheduled control for discrete-time systems depending on bounded rate parameters. *Int. J. Robust and Nonlinear Control*, 15, 473–494. doi:10.1002/rnc.1001.

Apkarian, P. (1997). On the discretization of LMI-synthesized linear parameter-varying controllers. *Automatica*, 33(4), 655–661. doi:10.1016/S0005-1098(96)00211-7.

Apkarian, P. (2011). Nonsmooth μ synthesis. *International Journal of Robust and Nonlinear Control*, 21(13), 1493–1508.

Apkarian, P. and Adams, R.J. (1998). Advanced gain-scheduling techniques for uncertain systems. *IEEE Transactions on Control Systems Technology*, 6(1), 21–32. doi:10.1109/87.654874.

Apkarian, P. and Gahinet, P. (1995). A convex characterization of gain-scheduled H_∞ controllers. *IEEE Transactions on Automatic Control*, 40(5), 853–864. doi:10.1109/9.384219.

Apkarian, P., Gahinet, P., and Becker, G. (1995). Self-scheduled H_∞ control of linear parameter-varying systems: a design example. *Automatica*, 31(9), 1251–1261. doi:10.1016/0005-1098(95)00038-X.

Apkarian, P. and Noll, D. (2006). Nonsmooth H_∞ synthesis. *IEEE Transactions on Automatic Control*, 51(1), 71–86. doi:10.1109/TAC.2005.860290.

Balas, G., Packard, A.K., Renfrow, J., Mullaney, C., and M'closkey, R.T. (1998). Control of the F-14 aircraft lateral-directional axis during powered approach. *Journal of Guidance, Control, and Dynamics*, 21(6), 899–908. doi:10.2514/2.4323.

Balini, H.M.N.K., Witte, J., and Scherer, C.W. (2012). Synthesis and implementation of gain-scheduling and LPV controllers for an AMB system. *Automatica*, 48(3), 521–527. doi:10.1016/j.automatica.2011.08.061.

Barker, J.M. and Balas, G.J. (2000). Comparing linear parameter-varying gain-scheduled control techniques for active flutter suppression. *Journal of Guidance, Control, and Dynamics*, 23(5), 948–955. doi:10.2514/2.4637.

Beckar, G. and Packard, A. (1994). Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback. *Systems & Control Letters*, 23(3), 205–215. doi:10.1016/0167-6911(94)90006-X.

Bliman, P.A. (2004). An existence result for polynomial solutions of parameter-dependent LMIs. *Systems & Con-

trol Letters, 51(3–4), 165–169. doi:10.1016/j.sysconle.2003.08.001.

Briat, C. (2015). *Linear Parameter-Varying and Time-Delay Systems*, volume 3 of *Advances in Delays and Dynamics*. Springer-Verlag, Heidelberg, Germany. doi:10.1007/978-3-662-44050-6.

Chen, L., Edwards, C., Alwi, H., and Sato, M. (2020). Flight evaluation of a sliding mode online control allocation scheme for fault tolerant control. *Automatica*, 114. doi:10.1016/j.automatica.2020.108829. Article No. 108829.

Daafouz, J., Bernussou, J., and Geromel, J.C. (2008). On inexact LPV control design of continuous-time polytopic systems. *IEEE Transactions on Automatic Control*, 53(7), 1674–1678. doi:10.1109/TAC.2008.928119.

de Aguiar, R.S.S., Apkarian, P., and Noll, D. (2018). Structured robust control against mixed uncertainty. *IEEE Transactions on Control Systems Technology*, 26(5), 1771–1781. doi:10.1109/TCST.2017.2723864.

de Caigny, J., Camino, J.F., Oliveira, R.C.L.F., Peres, P.L.D., and Swevers", J. (2012). Gain-scheduled dynamic output-feedback control for discrete-time LPV systems. *Int. J. Robust and Nonlinear Control*, 22(5), 535–558. doi:10.1002/rnc.1711.

Fialho, I., Balas, G.J., Packard, A.K., Renfrow, J., and Mullaney, C. (2000). Gain-scheduled lateral control of the F-14 aircraft during powered approach landing. *Journal of Guidance, Control, and Dynamics*, 23(3), 450–458. doi:10.2514/2.4550.

Gahinet, P. and Apkarian, P. (1994). A linear matrix inequality approach to H_∞ control. *Int. J. Robust and Nonlinear Control*, 4(4), 421–448. doi:10.1002/rnc.4590040403.

Hamada, Y., Ohtani, T., Kida, T., and Nagashio, T. (2011). Synthesis of a linearly interpolated gain scheduling controller for large flexible spacecraft ETS-VIII. *Control Engineering Practice*, 19(6), 611–625. doi:10.1016/j.conengprac.2011.02.005.

Hashemi, S.M., Abbas, H.S., and Werner, H. (2012). Low-complexity linear parameter-varying modeling and control of a robot manipulator. *Control Engineering Practice*, 20(3), 248–257. doi:10.1016/j.conengprac.2011.11.002.

Hjartarson, A., Seiler, P., and Packard, A. (2015). LPVTools: A toolbox for modeling, analysis, and synthesis of parametervarying control systems. In *1st IFAC Workshop on Linear Parameter Varying Systems*. Grenoble, France. doi:10.1016/j.ifacol.2015.11.127.

Hoffmann, C. and Werner, H. (2015). A survey of linear parameter-varying control applications validated by experiments or high-fidelity simulations. *IEEE Transactions on Control Systems Technology*, 23(2), 416–433. doi:10.1109/TCST.2014.2327584.

Inokuchi, H. (2003). Position error for the airspeed sensor of the multi-purpose aviation laboratory MuPAL-α. Technical report, National Aerospace Laboratory of Japan. URL `https://repository.exst.jaxa.jp/dspace/handle/a-is/38313`. NAL-TM-776 (in Japanese).

Iwasaki, T. and Skelton, R.E. (1994). All controllers for the general H_∞ control problem: LMI existence conditions and state space formulas. *Automatica*, 30(8), 1307–1317. doi:10.1016/0005-1098(94)90110-4.

Kaminer, I., Pascoal, A.M., Khargonekar, P.P., and Coleman, E.E. (1995). A velocity algorithm for the implementation of gain-scheduled controllers. *Automatica*, 31(8), 1185–1191. doi:10.1016/0005-1098(95)00026-S.

Kawaguchi, J., Ninomiya, T., and Miyazawa, Y. (2011). Stochastic approach to robust flight control design using hierarchy-structured dynamic inversion. *Journal of Guidance, Control, and Dynamics*, 34(5), 1573–1577. doi:10.2514/1.53257.

Köroğlu, H. (2010). Robust generalized asymptotic regulation via an LPV controller without parameter derivative dependence. In *Proceedings of the 49th IEEE Conference on Decision and Control*, 4965–4971. IEEE, Atlanta, GA. doi:10.1109/CDC.2010.5717033.

Lane, S.H. and Stengel, R.F. (1988). Flight control design using non-linear inverse dynamics. *Automatica*, 24(4), 471–483. doi:10.1016/0005-1098(88)90092-1.

Leith, D.J. and Leithead, W.E. (2000). Survey of gain-scheduling analysis and design. *International Journal of Control*, 73(11), 1001–1025. doi:10.1080/002071700411304.

Lhachemi, H., Saussié, D., and Zhu, G. (2015). A structured H_∞-based optimization approach for integrated plant and self-scheduled flight control system design. *Aerospace Science and Technology*, 45, 30–38. doi:10.1016/j.ast.2015.04.003.

Lhachemi, H., Saussié, D., and Zhu, G. (2016). Gain-scheduling control design in the presence of hidden coupling terms. *Journal of Guidance, Control, and Dynamics*, 39(8), 1872–1880. doi:10.2514/1.G001830.

Masubuchi, I., Kato, J., Saeki, M., and Ohara, A. (2004). Gain-scheduled controller design based on descriptor representation of LPV systems: Application to flight vehicle control. In *Proceedings of the 43rd IEEE Conference on Decision and Control*, 815–820. IEEE, Paradise Island, Bahamas. doi:10.1109/CDC.2004.1428770.

Masubuchi, I. and Kurata, I. (2011). Gain-scheduled control via filtered scheduling parameters. *Automatica*, 47(8), 1821–1826. doi:10.1016/j.automatica.2011.05.005.

Masubuchi, I. and Yabuki, Y. (2020). Synthesis of memory gain-scheduled controllers for discrete-time LPV systems. *SICE Journal of Control, Measurement, and System Integration*, 13(5), 249–255. doi:10.9746/jcmsi.13.249.

Masui, K. and Tsukano, Y. (2000). Development of a new in-flight simulator MuPAL-α. In *AIAA Modeling and Simulation Technologies Conference*. AIAA, Denver, CO. doi:10.2514/6.2000-4574. AIAA 2000-4574.

Matsumura, F., Namerikawa, T., Hagiwara, K., and Fujita, M. (1996). Application of gain scheduled H_∞ robust controllers to a magnetic bearing. *IEEE Transactions on Control Systems Technology*, 4(5), 484–493. doi:10.1109/87.531915.

Miyazawa, Y. (1992). Robust flight control system design with multiple model approach. *Journal of Guidance, Control, and Dynamics*, 15(3), 785–788. doi:10.2514/3.20908.

Mohammadpour, J. and Scherer, C.W. (eds.) (2012). *Control of Linear Parameter Varying Systems with Applications*. Springer-Verlag, New York, 1st edition.

Ohara, A., Yamaguchi, Y., and Morito, T. (2001). LPV modeling and gain scheduled control of re-entry vehicle

in approach and landing phase. In *AIAA Guidance, Navigation, and Control Conference*. Montreal, Canada. doi:10.2514/6.2001-4038. AIAA 2001-4038.

Ohno, M., Yamaguchi, Y., Hata, T., Takahama, M., Miyazawa, Y., and Izumi, T. (1999). Robust flight control law design for an automatic landing flight experiment. *Control Engineeinrg Practice*, 7(9), 1143–1151. doi:10.1016/S0967-0661(99)00085-4.

Packard, A. (1994). Gain scheduling via linear fractional transformations. *Systems & Control Letters*, 22(2), 79–92. doi:10.1016/0167-6911(94)90102-3.

Petersen, I.R. (1987). A stabilization algorithm for a class of uncertain linear systems. *Systems & Control Letters*, 8(4), 351–357. doi:10.1016/0167-6911(87)90102-2.

Pratt, R.W. (ed.) (2000). *Flight Control Systems: practical issues in design and implementation*. IEE Control Engineering Series. The Institute of Electrical Engineers.

Rugh, W.J. and Shamma, J.S. (2000). Research on gain-scheduling. *Automatica*, 36(10), 1401–1425. doi:10.1016/S0005-1098(00)00058-3.

Sadeghzadeh, A. (2018). Gain-scheduled continuous-time control using polytope-bounded inexact scheduling parameters. *International Journal of Robust and Nonlinear Control*, 28, 5557–5574. doi:10.1002/rnc.4333. Nov.

Sato, M. (2011). Gain-scheduled output-feedback controllers depending solely on scheduling parameters via parameter-dependent Lyapunov functions. *Automatica*, 47(12), 2786–2790. doi:10.1016/j.automatica.2011.09.023.

Sato, M. (2018). Gain-scheduled flight controller using bounded inexact scheduling parameters. *IEEE Transactions on Control Systems Technology*, 26(3), 1074–1082. doi:10.1109/TCST.2017.2692750.

Sato, M. (2021). Dual formulation of causal gain-scheduled output feedback controller design using parameter-dependent Lyapunov functions. *SICE Journal of Control, Measurement, and System Integration*, 14(1), 196–205. doi:10.1080/18824889.2021.1953304.

Sato, M., Hardier, G., Ferreres, G., Edwards, C., Alwi, H., Chen, L., and Marcos, A. (2018). Fault tolerance -background and recent trends. *Journal of the Society of Instrument and Control Engineers*, 57(4), 279–286.

Sato, M. and Muraoka, K. (2015). Flight controller design and demonstration of quad tilt wing unmanned aerial vehicle. *Journal of Guidance, Control, and Dynamics*, 38(6), 1071–1082. doi:10.2514/1.G000263.

Sato, M. and Peaucelle, D. (2011). Gain-scheduled H_∞ controllers being derivative-free of scheduling parameters via parameter-dependent Lyapunov functions. In *Proceedings of the 18th IFAC World Congress*, 7951–7956. Milano, Italy. doi:10.3182/20110828-6-IT-1002.00048.

Sato, M. and Peaucelle, D. (2012). *Control of Linear Parameter Varying Systems with Applications*, chapter Gain-Scheduled Output-Feedback Controllers with Good Implementability and Robustness, 181–215. Springer. doi:{10.1007/978-1-4614-1833-7_8}.

Sato, M. and Peaucelle, D. (2013). Gain-scheduled output-feedback controllers using inexact scheduling parameters for continuous-time LPV systems. *Automatica*, 49(4), 1019–1025. doi:10.1016/j.automatica.2013.01.034.

Sato, M. and Peaucelle, D. (2020). Conservatism reduction for linear parameter-varying control design facing inexact scheduling parameters illustrated on flight tests. *Int. J. Robust and Nonlinear Control*, 30(15), 6130–6148. doi:10.1002/rnc.5037.

Sato, M. and Peaucelle, D. (2021). Causal gain-scheduled outputfeedback controllers using parameter-dependent Lyapunov functions. *Automatica*, 129. doi:10.1016/j.automatica.2021.109569. Article No. 109569.

Sato, M. and Satoh, A. (2008). Simultaneous realization of handling and gust responses: In-flight simulator controller design. *Journal of Guidance, Control, and Dynamics*, 31(6), 1545–1560. doi:10.2514/1.36713.

Sato, M. and Satoh, A. (2011). Flight control experiment of multipurpose-aviation-laboratory-α in-flight simulator. *Journal of Guidance, Control, and Dynamics*, 34(4), 1081–1096. doi:10.2514/1.52400.

Sato, M. and Sebe, N. Observer-structured controller for linear parametric plant: Beyond Luenberger observer-based controller. (submitted).

Sato, M. and Sebe, N. (2022). Conversion from unstructured LPV controllers to observer-structured LPV controllers. In *Proceedings of IEEE Conference on Decision and Control*. IEEE, Cancún, Mexico. (to appear).

Scherer, C.W. (1996). Mixed H_2/H_∞ control for time-varying and linear parametrically-varying systems. *International Journal of Robust and Nonlinear Control*, 6(9-10), 929–952. doi:10.1002/(SICI)1099-1239(199611)6:9/10⟨929::AID-RNC260⟩3.0.CO;2-9.

Scherer, C.W. (2001). LPV control and full block multipliers. *Automatica*, 37(3), 361–375. doi:10.1016/S0005-1098(00)00176-X.

Sename, O., Gáspár, P., and Bokor, J. (eds.) (2013). *Robust Control and Linear Parameter Varying Approaches*, volume 437 of *Lecture Notes in Control and Information Sciences*. Springer-Verlag, Berlin, 1st edition.

Shin, J.Y., Balas, G., and Kaya, M.A. (2002). Blending methodology of linear parameter varying control synthesis of F-16 aircraft system. *Journal of Guidance, Control, and Dynamics*, 25(6), 1040–1048. doi:10.2514/2.5008.

Stengel, R.F. (2004). *Flight Dynamics*. Princeton University Press, Princeton, New Jersey, 1st edition.

Stevens, B.L. and Lewis, F.L. (1992). *Aircraft Control and Simulation*. John Wiley & Sons, Inc., New York.

Wu, F. and Dong, K. (2006). Gain-scheduling control of LFT systems using parameter-dependent Lyapunov functions. *Automatica*, 42(1), 39–50. doi:10.1016/j.automatica.2005.08.020.

Wu, F., Yang, X.H., Packard, A., and Becker, G. (1996). Induced L_2-norm control for LPV systems with bounded parameter variation rates. *International Journal of Robust and Nonlinear Control*, 6(9-10), 983–998. doi:10.1002/(SICI)1099-1239(199611)6:9/10⟨983::AID-RNC263⟩3.0.CO;2-C.

Xie, L. and de Souza, C.E. (1992). Robust H_∞ control for linear systems with norm-bounded time-varying uncertainty. *IEEE Transactions on Automatic Control*, 37(8), 1188–1191. doi:10.1109/9.151101.

Zhao, P. and Nagamune, R. (2017). Switching LPV controller design under uncertain scheduling parameters. *Automatica*, 76, 243–250. doi:10.1016/j.automatica.2016.10.026.