LPV Gain-Scheduling Control of an Electromechanically Driven Landing Gear for a Commercial Aircraft

H. P. Lüdders, H. Abbas, D. Doberstein, F. Thielecke and H. Werner

Abstract—This paper presents an application of Linear-Parameter-Varying (LPV) control to an electromechanically driven landing gear of an aircraft. The LPV approach is motivated by the highly nonlinear characteristics of the kinematic. First, a nonlinear model is derived using physical modeling tools. A quasi-LPV model is then derived by using an approach which is based on linearisation and additional measurements of the nonlinear model. Using mixed sensitivity loop shaping, a polytopic LPV controller based on a single Lyapunov function is designed. Then a method is developed for this controller which compensates any wind up effects due to a reconstruction and without any further calculation. Finally the designed controller is tested and compared to a Linear-Time-Invariant (LTI) controller in simulation studies.

I. INTRODUCTION

Landing gears of commercial aircrafts usually operate with hydraulic energy [8]. Due to the new technology-trend “More Electric Aircraft” it is one goal to substitute the hydraulic components with electromechanical solutions. For this reason a new approach was developed which enables to retract and to extend the front landing gear of an aircraft with a rotatory electromechanical actuator [4]. This innovative principle is shown in Fig. 1. The new concept fulfills all the necessary safety requirements.

If there is a total loss of electrical power-supply for the landing gear, it is still possible to execute a “freefall” due to the spring. The motor of the actuator has to be connected to a resistant to limit the speed of the freefall. Moreover, the landing gear cannot be driven reversely in the fully extended position because the ratio \( \frac{\theta_{\text{gear}}}{\theta_{\text{actuator}}} \) between actuator angle and landing gear angle becomes zero. This electromechanically driven landing gear has several advantages. Due to reduced load peaks it is more energy-efficient. Moreover the complexity can be reduced compared to the hydraulic solution.

In order to preserve the landing gear it is required to control the extension and the retraction. The challenge in designing a suitable controller is the high nonlinearity of the system. Fig. 2 shows the changing ratio between the actuator and landing gear angles over the operating range. The ratio \( \frac{\theta_{\text{gear}}}{\theta_{\text{actuator}}} \) changes in a range of 0 to 1.02 and so the inertias change as well.

Due to the nonlinearity a cascade LTI controller did not show the required performance as landing gears has to be extended and retracted in a short time in order to minimize drag impact.

This paper presents the application of LPV-control for such electromechanically driven landing gear.

II. MODELING OF THE LANDING GEAR

Fig. 1 shows a scheme of the kinematic model. The kinematics can be interpreted as a 6-link-chain. For a mathematical modeling five differential equations (each for one body) have to be coupled. The mathematical description is not an easy task. For this reason it is common in mechanical engineering to use multi-body-simulation-programs [4], for example SIMMECHANICS. With SIMMECHANICS a nonlinear model can be created in an efficient way. The electromechanical actuator can be built in SIMULINK. This model can be directly linked to the kinematic model, as SIMMECHANICS is part of the SIMULINK environment. Fig. 3 shows how the sub models are linked for the nonlinear simulation. In all models coulomb and viscous frictions are considered. The result of the modeling is a nonlinear model.
of order eight. The kinematic model consists of rigid bodies and has one degree of freedom.

Fig. 3. Complete nonlinear simulation model

A. LPV-Modelling

A linear parameter varying model is defined by a parameter dependent state-space model

\[
G(\theta) := \begin{cases} 
\dot{x}(t) = A(\theta(t))x(t) + B(\theta(t))u(t) \\
y(t) = C(\theta(t))x(t) + D(\theta(t))u(t)
\end{cases} \quad (1)
\]

with a compact set

\[
\mathcal{P} \subseteq \mathbb{R}^l : \theta \in \mathcal{P}, \forall t > 0.
\]

where \(x \in \mathbb{R}^n\) is the state vector, \(u \in \mathbb{R}^m\) the input vector, \(y \in \mathbb{R}^p\) the output vector. The mappings \(A(\theta), B(\theta), C(\theta), D(\theta)\) are analytic functions of \(\theta\). The parameter vector \(\theta(t)\), see (2), represents a time-varying parameter vector and depends on a vector of measurable signals \(\rho(t)\) referred to as scheduling signals, according to

\[
\theta(t) = f_\theta(\rho(t)),
\]

where \(f : \mathbb{R}^k \rightarrow \mathbb{R}^l\) is an analytic mapping. If \(\rho(t)\) depends on the system state, output or input, the LPV model is called quasi-LPV. In this section, the task is to derive a quasi-LPV model of the landing gear in the form (1).

A LPV-model is called polytopic if it can be presented by a linear combination of a finite number of state-space matrices of the LPV-model at frozen values of \(\theta\) called the vertices \(\theta_\nu\). A matrix polytope is defined as a convex hull of these finite number of matrices [2]. The properties are shown as follows:

\[
C_\theta \left\{ \begin{bmatrix} A(\theta_{1}) & B(\theta_{1}) \\ C(\theta_{1}) & D(\theta_{1}) \end{bmatrix}, \ldots, \begin{bmatrix} A(\theta_{N}) & B(\theta_{N}) \\ C(\theta_{N}) & D(\theta_{N}) \end{bmatrix} \right\} \quad (4)
\]

\[
= \sum_{i=1}^{N} \alpha_i \begin{bmatrix} A(\theta_{i}) \\ C(\theta_{i}) \end{bmatrix} \begin{bmatrix} B(\theta_{i}) \\ D(\theta_{i}) \end{bmatrix} \quad \text{with} \quad \alpha_i \geq 0, \quad \sum_{i=1}^{N} \alpha_i = 1 \quad (5)
\]

In order to design a low order LPV controller, it is reasonable to build up a low order LPV-model for the landing gear, as the order of the LPV controller depends on the order of the model with the synthesis strategy [2] utilized here. In this context, the dynamics of the actuator, which consists of the brushless DC motor and the two gears, are not taken into account. In order to improve the accuracy, the inertia and the viscous friction of the actuator are added to the first body of the kinematic model, respectively. The Coulomb friction of this simplified model is approximated by adding a viscous friction term in order to avoid discontinuities. The actuator is thus reduced to a static gain. Fig. 4 summarizes this procedure. Due to this order reduction a simplified nonlinear description of the landing gear can be obtained.

The simplified nonlinear model has an order of two and is described with SIMMECHANICS. Now a methodology is presented to obtain a LPV model for the SIMMECHANICS simulation.

SIMMECHANICS allows to linearize the nonlinear model at any operating point. Hence it is possible to build an LPV model based on Jacobi-linearized models, and interpolating between them. However it was shown in [5] that this method may lead to a poor closed loop performance when controllers are designed based on such a model. This stems from the fact that this model is an interpolation between individual dynamic systems, each with its distinct states, inputs and outputs, rather than a single dynamic system. Nevertheless, the methodology used here to construct a LPV model is based on the Jacobi-linearization approach, too. However, further information, like steady input gain and others, are used to improve the modeling.

For the landing gear system which has one degree of freedom and one connected actuator, the general nonlinear differential equation is given as follows (only viscous friction is assumed and outer influences are neglected):

\[
J(\varphi_{Act,LG}) \cdot \varphi_{Act,LG} = G(\varphi_{Act,LG}) + d(\varphi_{Act,LG}) \cdot \varphi_{Act,LG} + M_{Act,LG}
\]

\[
\varphi_{LG} = i_{Pos}(\varphi_{Act,LG}) \cdot \varphi_{Act,LG}
\]

All quantities in (7) and (8) are scalars. Here \(J\) is the inertia of the system which varies with the actual angle of the actuator. \(G\) contains all gravitational torques and torques due to internal springs. This factor varies with the actual angle of the actuator as well. The \(d\)-term summarizes the viscous friction of the kinematics and depends on the actual angle. \(M_{Act,LG}\) is the drive torque of the actuator and \(\varphi_{LG}\) describes the output angle of the kinematic. To calculate \(\varphi_{LG}\) from the actuator angle \(\varphi_{Act,LG}\), it has to be multiplied by the varying position ratio \(i_{Pos}\). The problem is that the varying functions \(J, G, d, i_{Pos}\) are unknown and have to be identified.
The general differential equation is shown as

\[
\begin{bmatrix}
\dot{\phi}_{Act, LG}
\end{bmatrix}
+ \begin{bmatrix}
0 & \frac{c(\phi_{Act, LG})}{J(\phi_{Act, LG})} & \frac{d(\phi_{Act, LG})}{J(\phi_{Act, LG})}
\end{bmatrix}
\begin{bmatrix}
\phi_{Act, LG}
\end{bmatrix}
= - \begin{bmatrix}
0 & 0 & \frac{1}{J(\phi_{Act, LG})}
\end{bmatrix}
\begin{bmatrix}
\varphi_{Act, LG}
\end{bmatrix}
\]

(9)

Hence, four scheduling variables are introduced as

\[
\theta_{A21} = \frac{G(\phi_{Act, LG})}{J(\phi_{Act, LG})} \phi_{Act, LG}, \quad \theta_{B21} = \frac{d(\phi_{Act, LG})}{J(\phi_{Act, LG})}.
\]

In the differential equation high order terms like \( \ddot{\phi}_{Act, LG} \) can be neglected. Hence the linearized version of (7) is given as

\[
J(\phi_{Act, LG}) \left( \ddot{\phi}_{Act, LG} \right) = G(\phi_{Act, LG}) \cdot \dot{\phi}_{Act, LG} + \frac{d(\phi_{Act, LG})}{J(\phi_{Act, LG})} \cdot \dot{\phi}_{Act, LG} + M_{Act, LG}
\]

(21)

with \( M_{Act, LG} = M_{Act, LG} - M_{Act, LG} \) and \( M_{Act, LG} = M_{Act, LG} - (-G(\phi_{Act, LG})) \).

(22)

Transforming (21) into a state space model leads to the following expression

\[
\begin{bmatrix}
\dot{\phi}_{Act, LG}
\end{bmatrix}
+ \begin{bmatrix}
0 & \frac{c(\phi_{Act, LG})}{J(\phi_{Act, LG})} & \frac{d(\phi_{Act, LG})}{J(\phi_{Act, LG})}
\end{bmatrix}
\begin{bmatrix}
\phi_{Act, LG}
\end{bmatrix}
= - \begin{bmatrix}
0 & 0 & \frac{1}{J(\phi_{Act, LG})}
\end{bmatrix}
\begin{bmatrix}
\varphi_{Act, LG}
\end{bmatrix}
\]

(23)

\[
\dot{\varphi}_{LG} = \begin{bmatrix}
0 & 0 & \frac{1}{J(\phi_{Act, LG})}
\end{bmatrix}
\begin{bmatrix}
\phi_{Act, LG}
\end{bmatrix}
\]

(24)

By comparing (9) and (23) it can be seen that the varying parameters \( \theta_{A22} \) and \( \theta_{B21} \) and the scheduling signals \( \rho_{A22} \) and \( \rho_{B21} \) can be measured if the kinematic model is linearized in a sufficient number of linearization points to cover the whole range of operation. However, the underlying functions of the parameters \( \theta_{A21} \) and \( \theta_{C31} \) have to be established using other methods.

In SimMechanics it is possible to measure the static drive torque \( M_{Act, LG} \) required to hold the kinematic model in the linearization point. According to (22) this torque is equivalent to \( -G(\phi_{Act, LG}) \). With \( \frac{1}{J(\phi_{Act, LG})} \theta_{B21} \) and \( \phi_{Act, LG} \) as the linearization angle, all terms are known to calculate \( \theta_{A21} \) as

\[
\theta_{A21} = - \frac{M_{Act, LG}}{\phi_{Act, LG}} \theta_{B21}.
\]

(25)

The function of \( \theta_{C31} \) cannot be identified based on the linearization approach, as shifting the linearization point \( i_{pos}(\phi_{Act, LG}) \) only describes the position ratio between \( \phi_{Act, LG} \) and \( \phi_{LG} \) which is not identical to \( i_{pos}(\phi_{Act, LG}) \). However, with additional measurement of the output angle \( \phi_{LG} \) at each linearization point all information is given to calculate \( \theta_{C31} \) as

\[
\theta_{C31} = i_{pos}(\phi_{Act, LG}) = \frac{\phi_{LG}(\phi_{Act, LG})}{\phi_{Act, LG}}.
\]

(26)

This described procedure is summarized in Fig. 5. By analyzing the evolution of these four parameters it can be shown that the parameter \( \theta_{A22} \) can be approximated as an affine function of \( \theta_{B21} \).

\[
\theta_{A22} \approx \theta_{A22}^* = -k_1 \theta_{B21} - k_2
\]

(27)

The factors \( k_1 \) and \( k_2 \) are constants, therefore the polytopic character of the LPV model can be obtained.

It is worth to mention that the above described method to construct the quasi-LPV model for the landing gear can be adapted to any other physically modeled kinematics with one degree of freedom.
B. Verification of the Quasi-LPV Model

According to the procedure in the previous subsection, a 2nd order quasi-LPV model has been derived with three scheduling parameters for the landing gear system. Before this model is used to design a controller, it has to be validated against the nonlinear model. For this reason the closed loop behavior of the LPV model and of the nonlinear model are compared. The comparison is performed with a linear controller that was designed beforehand. Fig. 6 shows the results of the validation. A multisine-signal is commanded to both closed loop systems, considering the anticipated frequency of the landing gear. In Fig. 6, the first diagram compares the outputs of both systems, while the second one compares the control inputs. The LPV model describes the nonlinear model in a reasonable manner. However, a difference can be seen in the controller effort, since coulomb-friction is not considered in the LPV model, the controller output shows unsteady behavior when the angle velocity of the landing gear goes to zero. In the LPV model only viscous friction is implemented, therefore the controller output of the model is smooth. Nevertheless, the LPV model will be used in the following to design a LPV controller for the landing gear.

III. Controller Synthesis

LPV gain-scheduling can be used to design controllers for nonlinear systems which automatically adapt to the actual state of the system. With the existence of a quadratic Lyapunov function $P$ the designed LPV-controller guarantees stability and a level of control performance for the whole parameter set $\mathcal{P}$. In [2] the transformation of the controller synthesis problem to a set of Linear Matrix Inequalities (LMIs) is presented. With this, a LPV-controller for the landing gear can be designed using a mixed sensitivity loop shaping approach. The generalized plant for this synthesis is shown in Fig. 7. The block $G_{LPV,LG}$ represents the second order LPV-model for the landing gear. $G_{PreF}$ and $G_{PostF}$ are pre- and postfilter respectively, which are added to remove the parameter dependence of the B and C matrices of the generalized plant [2]. They are of order one and have a sufficient bandwidth to ensure that they will not interact with the dynamics of the landing gear. $W_1$ and $W_2$ are shaping filters of order one which are used to define design criteria for the controller-synthesis. With the structure in Fig. 7 for the general plant, it is possible to influence the input-disturbance rejection of the closed loop system directly. This approach showed better results for this system compared to the "standard" loop-shaping approach where the sensitivity and the control sensitivity are weighted [6]. With

$$W_1 = \frac{103530}{125s + 1}, \quad W_2 = \frac{s + 100}{s + 10^2}$$

the controller is computed and the performance criteria are met. The result of this synthesis is a LPV-controller $k(\theta)$ of order five. Due to the rank condition which can be utilized to decrease the order of the controller, see [2], the order of the controller is one less than the order of the generalized plant. The controller $k(\theta)$ is given as

$$\dot{x}_k(t) = A_k(\theta(t))x(t) + B_k(\theta(t))e(t)$$

$$u_k(t) = C_k(\theta(t))x_k(t) + D_k(\theta(t))e(t).$$

A. Anti-Windup-Design

In order to achieve an anti-windup behavior of the designed LPV controller, we extend the method which has been proposed previously in [3] for LTI systems, to the LPV case. This method, under the assumption that $D_k(\theta)$ is invertible, $\forall \theta \in \mathcal{P}$, is simply based on changing the structure of the controller in the closed loop system to deal with the windup problem.

Fig. 8 displays the general form of the LPV-controller with a saturation block. A windup occurs when the states of the controller are driven by the control error $e$ while the saturation is already reached. The goal is to avoid windup by
using a new formulation for the controller. By multiplying (30) with $H(\theta) \in \mathbb{R}^{n \times m}$ and subtracting from (29) we get:

$$\dot{x}_k = (A_k(\theta) - B_k(\theta)D_k(\theta))x_k + B_k(\theta)D_k(\theta)e + H(\theta)u$$

$$u = C_k(\theta)x_k + D_k(\theta)e.\tag{32}$$

Using the actual plant input $\dot{u}$ instead of $u$ and selecting $H(\theta) = B_k(\theta)D_k^{-1}(\theta)$, the states of the controller are driven in the same way as the states of the plant, namely by $\dot{u}$. Hence the state space model of the controller is given as

$$\dot{x}_k = (A_k(\theta) - B_k(\theta)D_k^{-1}(\theta)C_k(\theta))x_k + B_k(\theta)D_k^{-1}(\theta)\dot{u}$$

$$u = C_k(\theta)x_k + D_k(\theta)e,\tag{33}$$

$$\dot{u} = \text{sat}(u).\tag{34}$$

The new structure of the controller is shown in Fig. 9. Now, the controller has to receive the additional signal $\dot{u}$. This method is an efficient way which is based on just changing the structure of the LPV controller without adding any complexity in order to achieve Anti-Windup behavior. Moreover, no extra design variables are required to be computed. The only requirement is that $D_k(\theta)$ should have full rank for $\forall \theta \in \mathcal{P}$. Hence a necessary condition is that the system has the same number of inputs and outputs; however, this is not sufficient to insure full rank. This will be investigated more in future contributions.

**IV. RESULTS**

This section presents the implementation of the designed LPV-controller and simulation results. This LPV-controller is compared to a LTI cascade controller which was designed in preliminary work.

**A. Implementation of the LPV Controller**

The controller has two inputs, the control error $e$ and the plant input $\dot{u}$. Additionally the controller receives the three scheduling signals $\theta_{A21}$, $\theta_{B21}$ and $\theta_{C21}$. These signals are provided with lookup-tables in which the scheduling functions $p_{A21}$, $p_{B21}$ and $p_{C21}$ are stored. The block diagram of the closed loop system is shown in Fig. 10. It is necessary to implement the nonlinear model with pre- and postfilter in the closed loop system to get accurate results, based on the design model [2]. However, the results do not degrade visibly if these filters are omitted.

**B. Implementation of the Cascade Controller**

The LPV-controller is compared to an LTI-cascade controller which was designed in preliminary work. This controller consists of two cascades. The first one controls the angular velocity of the electromechanical actuator $\omega_{Act,LG}$. The outer cascade controls the angle of the landing gear. The two parts of this controller, the velocity controller and the position controller, were designed using root locus and frequency response, based on three linearized models of the landing gear of characteristic positions.

The cascade controller has to be able to stabilize all three linear models and to provide the required performance. For this reason the velocity controller consists of two integrators and two zeros. The position controller is only a proportional controller [6]. Fig. 11 shows the block diagram of the closed loop system with the cascade controller. It is obvious that the cascade controller needs also the velocity signal to be measured for the feedback.

**C. Testing Requirements**

Both controllers are tested on a trajectory that captures the requirements for the landing gear. First the landing gear is completely extended. It has to be retracted in approximately 8 s smoothly without overshooting in order to avoid structural damage. After a short holding time in the retracted position the landing gear has to extend in 8 s, as well. Additionally the landing gear has to reject a load trajectory during the movements. This load trajectory represents the worst-case scenario for the landing gear during actuation, for more details see [6]. In Fig. 12 this trajectory is shown in normalized form. The main effects in this trajectory are loads due to flight manoeuvres and aerodynamic loads.

**D. Comparison**

Fig. 13 demonstrates the results of the comparison between the cascade controller and the LPV-controller designed in the previous section. The solid line in this figure shows the reference signal which defines an optimal behavior of the
landing gear. The dashed line shows the system response if the nonlinear system is controlled by the LPV controller. The dot-dashed line indicates the system response with the LTI-cascade controller. During this testing the controllers have to reject the load trajectory defined in Fig. 12. As can be seen, the LPV controller shows better tracking capability. Especially in the extended position of the landing gear \( \phi_{LG} = 0 \), the LTI-controller has problems to achieve a reasonable tracking performance. The reason for this mainly comes from the nonlinear ratio of the landing gear kinematic. Because of possible system failure the internal ratio of the landing gear in this point becomes zero. The LPV controller knows this effect through the measured scheduling signals and can adjust its control output. Due to the better tracking capability, the LPV controller is able to finish the retraction and the extension procedure 3 s faster than the cascade controller. This improves the drag-performance of the aircraft as the landing gear and the doors which lock the compartment do not disturb the air flow longer than necessary.

\[ M_{\text{Strake}} = \begin{cases} 0 & : 0 < t < 3s \\ M_{\text{Load}} & : 3s \leq t \leq 3.5s \\ 0 & : t > 3.5s \end{cases} \]  \hspace{1cm} (35)

Fig. 14 shows the system response with and without anti-windup system for the LPV controller. It can be seen that the response without anti-windup shows a big overshoot and would crash into the aircraft structure, which would result in a critical damage of the aircraft. However, the Anti-Windup system can completely avoid this effect and thus only the increased control error during the rejection shows the existence of the additional load due to the bird strike.

\[ \text{Fig. 12. Load trajectory} \]

\[ \text{Fig. 13. Comparison of the reference tracking} \]

\[ \text{Fig. 14. Test results for the Anti Windup system} \]

V. CONCLUSIONS AND FUTURE WORK

This paper illustrates the successful application of a LPV gain scheduling controller synthesis for an electromechanically driven landing gear. First, a nonlinear model of the system with \textit{SimMechanics} and \textit{Simulink} is derived. Due to the lack of a complete mathematical description of the system, a technique is presented which combines general system knowledge, information from linearization and additional measurements to obtain an LPV model for the landing gear. With this model, a LPV controller, based on a quadratic Lyapunov-function, is developed. This LPV controller shows better results than an LTI-cascade controller designed previously and is able to track the reference in the desired manner. Additional efforts should be made to design a fixed structure, low order LPV controller, as it is done in [1], in order to simplify the structure and hence to increase the acceptance of using LPV-control techniques in civil aircraft industries. Experimental application on a test rig will be available at the end of 2010.

REFERENCES


4664