A TWO-TIME-SCALE CONTROL SCHEME FOR FAST SYSTEMS

S. Gros, D. Buccieri, Ph. Mullhaupt and D. Bonvin

Laboratoire d'Automatique
École Polytechnique Fédérale de Lausanne
CH-1015 Lausanne, Switzerland

Abstract: Model predictive control (MPC) is a very efficient approach to control nonlinear systems, especially if the systems are high dimensional and/or constrained. MPC formulates the problem of input trajectory generation as an optimization problem. However, due to model mismatch and disturbances, a frequent re-calculation of the trajectories is typically called for. This paper proposes a two-time-scale control scheme that uses repeated trajectory generation in a slow loop and time-varying linear feedback in faster loop. The latter reduces the effect of uncertainty, which allows reducing considerably the sampling frequency of the slow loop. The problem of trajectory generation is treated using two approaches: (i) optimization-based MPC, and (ii) flatness-based system inversion. The two-time-scale control scheme is illustrated via the simulation of a flying robotic structure. It is seen that the scheme combining optimization-based MPC and linear feedback is efficient and robust, but too slow to be considered in a real application. In contrast, the scheme combining flatness-based system inversion and linear feedback is fast, efficient and robust.

Keywords: Two-time-scale Control, Model Predictive Control, Flat Systems, Neighboring Extremals, Flying Robots, VTOL.

1. INTRODUCTION

Predictive control is an efficient approach for tackling problems with nonlinear dynamics and constraints, especially when analytical computation of the control law is difficult (Mayne *et al.*, 2000; Scokaert and Mayne, 1998). Standard predictive control involves re-calculating at every sampling instant the inputs that minimize a criterion defined over a horizon window in the future, taking into account the current state of the system.

A crucial point in predictive control is the extensive use of the dynamic model. Since this model is not always accurate, the predicted state evolution may differ from the actual plant evolution, which requires frequent re-calculation of the inputs. Solutions to this problem are proposed in the literature. One possibility is to cast the problem into a robust framework, where optimization is performed by taking the uncertainty into account explicitly. Standard robust predictive control computes a input trajectories that represent a compromise solution for the range of uncertainty considered (Bemporad and Morari, 1999; Lee and Yu, 1997; Kouvaritakis *et al.*, 2000). Such a methodology is widely used in the process industry where system dynamics are sufficiently slow to permit its implementation. However, due to the complexity of the calculations involved in robust predictive control, its applications to fast dynamics systems are rather limited.

Another solution consists of tracking the system trajectories with a fast feedback loop. If the local dynamics are nearly constant, linear control theory provides efficient tools to design this feedback loop. However, for systems having strongly-varying local dynamics, there is no systematic way of designing such a feedback law (Ronco *et al.*, 2001; Morari and Lee, 1999; Bemporad, 1998).

This trajectory-tracking problem can be tackled by the neighboring-extremal theory. For small deviations from the optimal solution, a linear approximation of the system and a quadratic approximation of the cost are quite reasonable. In such a case, a neighboring-extremal (NE) controller provides a closed-form solution to the optimization problem. Hence, the optimal inputs can be approximated using state feedback, i.e. without explicit numerical re-optimization.

This paper presents two approaches to control a simulated robotic flying structure known as VTOL (Vertical Take-Off and Landing). The structure has 4 inputs and 16 states. It is a fast and strongly nonlinear system. The control schemes is computed based on a simplified model of the system, while the simulation uses the original model.

The first control approach combines an optimization-based predictive controller in a slow loop and a linear time-varying NE-controller in a faster loop. The slow loop generates the reference input and state trajectories, while the fast loop ensures good tracking of the state trajectories. This control scheme is sufficiently efficient to make the re-optimization of the reference trajectories unnecessary. However, the computation time for the optimization tends to be large in comparison with the system dynamics.

Since the simplified VTOL model is flat (Fliess *et al.*, 1995; Fliess *et al.*, 1999), the optimization-based MPC can be replaced by an algebraic generation of the trajectories. This is done in the second approach, in which trajectory tracking is also ensured by a NE-controller.

Note that a controller based on feedback linearization can be computed for the simplified model. Though this controller is very efficient at controlling the original model of the system, it is not robust to variations in some of the parameters that are known to be uncertain in flying structures.

The paper is organized as follows. Section 2 briefly revisits optimization-based MPC and NE-control. The proposed two-time-scale control scheme is detailed in Section 3. Section 4 presents the simulated operation of a VTOL structure. Finally, conclusions are provided in Section 5.

2. PRELIMINARIES

2.1 Optimization-based MPC

Consider the nonlinear dynamic process:

$$\dot{x} = F(x, u), \qquad x(0) = x_0 \tag{1}$$

where the state x and the input u are vectors of dimension n and m, respectively. x_0 represents the initial conditions, and F the process dynamics.

Predictive control of (1) is based on repeatedly solving the following optimization problem:

$$\begin{aligned}\min_{u[t_k, t_k+T_f)} \; & J = \Phi(x(t_k + T_f)) \\ & \qquad + \frac{1}{2}\int_{t_k}^{t_k+T_f} L(x(\tau), u(\tau))d\tau \qquad (2)\\ s.t. \quad & \dot{x} = F(x, u), \qquad x(t_k) = x_m(t_k) \qquad (3)\end{aligned}$$

where Φ is an arbitrary scalar function of the states and L an arbitrary scalar function of the states and the inputs. $x_m(t)$ represents the measured or estimated value of $x(t)$. The re-optimization interval, $t_{k+1} - t_k$, is limited by the performance of the available optimization tools.

2.2 Neighboring Extremals

Upon including the dynamic constraints of the optimization problem in the cost function, the augmented cost function, $\bar{J}$, reads:

$$\bar{J} = \Phi(x(t_k + T_f)) + \int_{t_k}^{t_k+T_f} \left(H - \lambda^T \dot{x}\right) dt \tag{4}$$

where $H = L + \lambda^T F(x, u)$, and $\lambda(t)$ is the n-dimensional vector of adjoint states or Lagrange multipliers for the system equations.

The first-order variation of $\bar{J}$ is zero at the optimum. For a variation $\Delta x(t) = x(t) - x^\star(t)$ of the states of the system, minimizing the second-order variation of $\bar{J}$, $\Delta^2 \bar{J}$, with respect to $\Delta u = u - u^*$ represents a time-varying Linear Quadratic Regulator (LQR) problem for which a closed-form solution is available (Bryson, 1999):

$$\begin{aligned}\Delta u(t) &= -K(t)\Delta x(t) && (5)\\ K &= H_{uu}^{-1}\left(H_{ux} + F_u^T S\right) && (6)\\ \dot{S} &= -H_{xx} + S(F_u H_{uu}^{-1} H_{ux} - F_x) && (7)\\ &\quad + (H_{xu} H_{uu}^{-1} F_u^T - F_x^T)S && \\ &\quad + S F_u H_{uu}^{-1} F_u^T S + H_{xu} H_{uu}^{-1} H_{ux} && \\ S(t_k + T_f) &= \Phi(x(t_k + T_f)) && (8)\end{aligned}$$

Controller (5)-(8) is termed NE-controller.

3. TWO-TIME-SCALE CONTROL SCHEME

The repeated solution of (2)-(3) provides feedback to the system. Yet, since the time necessary to perform the optimization can be rather large compared to the system dynamics, the feedback provided by the re-optimization tends to be too slow to guarantee performance and robustness. Hence, it is proposed to add a fast feedback loop in the form of a NE-controller. The resulting control scheme is displayed in Figure 1. The NE-controller operates in the fast loop at the sampling frequency of the system, while the trajectory is generated in the slow loop at a frequency allowed by its computation. The generation of the reference trajectories $u_{ref}(t)$ and $x_{ref}(t)$ can be computed via optimization (e.g. MPC) or direct system inversion (as is possible for example for flat systems). Note that if the time-scale separation between the two loops is sufficient, $u_{ref}(t)$ can be considered as a feedforward term for the fast loop.

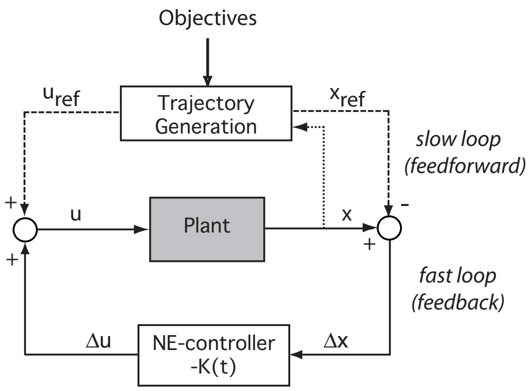

Fig. 1. Scheme combining trajectory generation (via optimization or system inversion) and NE-control

3.1 *Time-varying LQR*

The NE-controller (5)-(8) can be numerically difficult to compute. However, its computation is much easier if the optimization problem consists of trajectory tracking. Indeed, for tracking the trajectories $u_{ref}(t)$ and $x_{ref}(t)$, Φ and L can be chosen as:

$$\Phi = \frac{1}{2}(x - x_{ref})^T P(x - x_{ref}) \tag{9}$$

$$L = \frac{1}{2}(x - x_{ref})^T Q(x - x_{ref}) + \frac{1}{2}(u - u_{ref})^T R(u - u_{ref}) \tag{10}$$

for which the the adjoints become:

$$\dot{\lambda} = -H_x^T = -F_x^T \lambda - Q(x - x_{ref}) \tag{11}$$

$$\lambda(T_f) = \Phi_x(T_f) = 0 \tag{12}$$

i.e. they are zero along the whole trajectory. Hence, the NE-controller reduces to:

$$\Delta u(t) = -K(t)\Delta x(t) \tag{13}$$

$$K = R^{-1} F_u^T S \tag{14}$$

$$\dot{S} = -Q - SF_x - F_x^T S + SF_u R^{-1} F_u^T S \tag{15}$$

$$S(T_f) = P \tag{16}$$

which can be viewed as a time-varying LQR. Note that, if the local system dynamics are nearly constant, the NE-controller is well approximated by a LQR with a constant gain matrix K. In contrast, if the system is strongly time varying, it is necessary to compute the time varying NE-controller (13)-(16).

4. APPLICATION TO A VTOL STRUCTURE

4.1 *System dynamics*

The simulated example is a VTOL structure. The structure is made of four propellers mounted on the four ends of an orthogonal cross. Each propeller is motorized independently. The propeller rotational velocities are opposed as follows (when top-viewed, counted counterclockwise): propellers 1 and 3 are rotating counterclockwise, while propellers 2 and 4 are rotating clockwise. The angle of attack (AoA) of the blades and the positions of the propellers are fixed relative to the structure. The VTOL is controlled through the four motor torques. The states of the system are:

$$X = \begin{bmatrix} x & y & z & \eta_1 & \eta_2 & \phi & \dot{x} & \dot{y} & \dot{z} & \dot{\eta}_1 & \dot{\eta}_2 & \dot{\phi} & \dot{\rho}_1 & \dot{\rho}_2 & \dot{\rho}_3 & \dot{\rho}_4 \end{bmatrix} \tag{17}$$

Variables $\begin{bmatrix} x & y & z \end{bmatrix}$ give the position of the center of gravity G in $[m]$ within the laboratory referential (e_1, e_2, e_3). Variables $\begin{bmatrix} \eta_1 & \eta_2 & \phi \end{bmatrix}$ give the angular attitude of the structure in $[rad]$, with the transformation from the laboratory referential to the VTOL referential, $(e_1, e_2, e_3) \to (E_1, E_2, E_3)$, being described by the matrix $\Phi(\eta_1, \eta_2, \phi) = R_{e_3}(\phi) R_{e_2}(\eta_2) R_{e_1}(\eta_1)$, where $R_e(\alpha)$ is a rotation of angle α around the basis vector e.

Variable $\dot{\rho}_k$ is the speed of the propeller k in $[rad/s]$. The model of the VTOL can be computed through analytical mechanics. The aerodynamical forces and torques generated by the propellers are modeled using the standard squared velocity law. The resulting model is rather complicated and will not be explicited here. The reader is referred to (Gros, 2005) for details. The model is nonlinear, and its local dynamics are strongly time varying.

A simplified model can be computed by removing certain non-linearities, which is well justified in practice. Introducing the notations:

$$v_1 = \sum_{k=1}^{4} \dot{\rho}_k^2 \qquad v_2 = \sum_{k=1}^{4} (-1)^k \dot{\rho}_k^2$$
$$v_3 = \dot{\rho}_1^2 - \dot{\rho}_3^2 \qquad v_4 = \dot{\rho}_2^2 - \dot{\rho}_4^2$$
$$v_5 = \sum_{k=1}^{4} \dot{\rho}_k$$

the simplified model can be written as:

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = C_{xyz} \begin{bmatrix} sin(\eta_2) \\ -cos(\eta_2)sin(\eta_1) \\ cos(\eta_2)cos(\eta_1) \end{bmatrix} v_1 - \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} \tag{18}$$
$$\ddot{\phi} = C_\phi v_2 \tag{19}$$
$$\begin{aligned} \ddot{\eta}_1 = {} & \frac{1}{C_{\eta_1}^1 cos(\eta_2)^2 + C_{\eta_1}^2} (C_d sin(\eta_2) v_2 \\ & + C_s d cos(\eta_2)(sin(\phi) v_3 + cos(\phi) v_4) \\ & - I_A^M cos(\eta_2) \dot{\eta}_2 v_5) \end{aligned} \tag{20}$$
$$\begin{aligned} \ddot{\eta}_2 = {} & C_{\eta_2}(-cos(\phi) v_3 + sin(\phi) v_4 \\ & + I_M^A cos(\eta_2) \dot{\eta}_1 v_5) \end{aligned} \tag{21}$$
$$\dot{v}_k = u_k \qquad k = 1, ..., 4 \tag{22}$$

with the constants

$$C_{xyz} = \frac{C_s}{M_5 + 4m}$$
$$C_\phi = \frac{C_d}{4I_M^A + 4md^2 + I_M^5}$$
$$C_{\eta_1}^1 = -I_M^5 + I_S^5 - 4I_M^A + 4I_S^A - 2md^2 + 4mh^2$$
$$C_{\eta_1}^2 = I_5^M + 4I_M^A + 4md^2$$
$$C_{\eta_2} = \frac{C_s d}{4I_S^A + 2md^2 + 4mh^2 + I_S^5}$$

The inputs u_k, $k = 1...4$, do not represent the physical inputs (the motor torques M_k), but are related to them by invertible algebraic relationships. The parameter C_s is the aerodynamical coefficient for the sustentation of the propellers, C_d is the aerodynamical drag coefficient of the propellers. The parameter M_5 is the mass of the main body, without the propellers, m is the mass of a propeller. Parameters h and d are the horizontal and vertical distances, respectively, from the center of gravity of the main structure to the centers of the propellers. Parameter I_M^5 is the inertia of the main body (without the propellers) along the vertical axis, I_S^5 is the inertia along the two secondary axes. Parameter I_M^A is the inertia of a propeller along its vertical axis, while I_S^A is the inertia along the two secondary axes. The inertias of the propellers along their horizontal axis are considered identical. This assumption is reasonable since, as the rotation speeds of the propellers are high, the propellers can be considered as discs. The numerical values of the parameters used in the simulations are given in Table 1. They do not correspond to any particular real VTOL structure, yet they are realistic.

Cs	3.64×10^{-6}	Ns^2
Cd	1.26×10^{-6}	Nms^2
M_5	0.5	kg
m	2.5×10^{-2}	kg
h	0.03	m
d	0.3	m
I_M^5	181×10^{-4}	Nms^2
I_S^5	96×10^{-4}	Nms^2
I_M^A	6.26×10^{-6}	Nms^2
I_S^A	1.25×10^{-6}	Nms^2

Table 1. Model parameters

4.2 Control problem

The control problem is of the tracking type: the VTOL structure must be driven smoothly from some initial configuration to another predefined configuration. A translation from the position $(x = 0\,[m], y = 0\,[m], z = 0\,[m], \phi = 0\,[rad])$ to the position $(x = 1\,[m], y = 1\,[m], z = 1\,[m], \phi = 2\pi\,[rad])$ will be considered. The speeds and accelerations are zero initially as well at the final time $T_f = 4\,[s]$. The problem is here unconstrained. The various controllers are computed using the simplified model, while the simulations are done with the original model. Perturbations are introduced in the aerodynamical parameters C_s and C_d to represent the uncertainty resulting from self-induced turbulences and surface effects: C_s^1, C_d^1, C_s^2, C_d^2 are perturbed +50 percent and C_s^3, C_d^3, C_s^4, C_d^4 −50 percent, C_d^k, C_s^k being the aerodynamical coefficients of propeller k.

4.3 Control based on feedback linearization

Since the simplified model is flat, with the flat outputs being (x, y, z, ϕ), it is possible to compute a controller based on feedback linearization (Fliess *et al.*, 1995; Fliess *et al.*, 1999). However, the resulting controller was found to lack robustness with respect to the uncertain parameters C_s and C_d.

4.4 Control based on MPC

With no parametric uncertainty in C_s and C_d, the MPC scheme moves nicely to the desired state setpoints despite the fact that the inputs are computed based on the simplified model. However, MPC struggles when the parametric uncertainty exceeds 10 percent on the aerodynamical coefficient. Figure 2 shows the control performance for a 10 percent perturbation, the perturbation being applied to the four propellers as indicated in Subsection 4.2. A sampling time of $0.1\,[s]$ is used. The control is rather slow and exhibits a large overshoot.

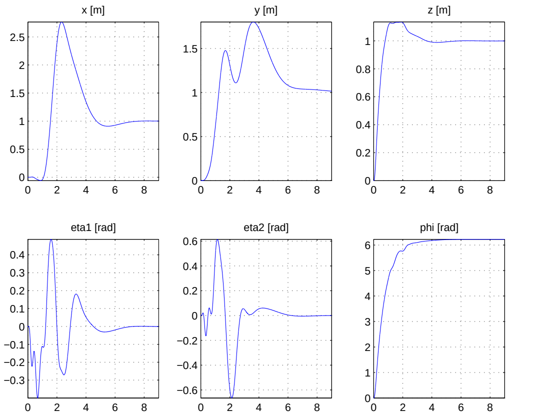

Fig. 2. MPC control performance

4.5 Control based on MPC + NE-controller

Next, the control scheme presented in Section 3 is applied to the VTOL structure. MPC is used in the slow loop and a NE-controller in the fast loop. The optimization in MPC is performed using a *modified* version of the simplified model of the VTOL. The cost function is chosen quadratic:

$$J = \frac{1}{2}\int_0^{T_f} [(\bar{x} - \bar{x}_{sp})^T Q(\bar{x} - \bar{x}_{sp}) + \bar{u}^T R\bar{u}]dt$$

with $\bar{x}_{sp}$ the state setpoints and $\bar{u} = \begin{bmatrix} u_1 & u_2 & \bar{u}_3 & \bar{u}_4 \end{bmatrix}^T$, where

$$\bar{v}_3 = sin(\phi)v_3 + cos(\phi)v_4 \quad \dot{\bar{v}}_3 = \bar{u}_3 \qquad (23)$$

$$\bar{v}_4 = -cos(\phi)v_3 + sin(\phi)v_4 \quad \dot{\bar{v}}_4 = \bar{u}_4 \qquad (24)$$

and $\bar{x} = \begin{bmatrix} X(1:12) & v_1 & v_2 & \bar{v}_3 & \bar{v}_4 \end{bmatrix}^T$.

This change of variables removes the coupling between the states and allows the optimization algorithm to perform much better. The corresponding reference torques and states trajectories for the original model can be computed algebraically from $\bar{x}$ and $\bar{u}$. The weighting matrices R and Q are chosen to obtain the desired dynamics. Here, they are chosen diagonal, with the diagonal terms R_D and Q_D given as:

$$R_D = 10^{-9} \cdot \begin{bmatrix} 1.8 \cdot 10^{-3} & 0.1058 & 28.125 & 27.38 \end{bmatrix}$$

$$Q_D(1:8) = 10^3 \cdot \begin{bmatrix} 1 & 1.5 & 1 & 0.5 & 0.1 & 1 & 0.5 & 0.45 \end{bmatrix}$$

$$Q_D(9:16) = 10^3 \cdot \begin{bmatrix} 0.5 & 0.5 & 0.5 & 0.5 & 0 & 0 & 0 & 0 \end{bmatrix}$$

The computation of the NE-controller on the basis of the *modified* simplified model (i.e. the one used to compute $\bar{u}_{ref}(t)$ and $\bar{x}_{ref}(t)$) leads to robustness problems (as the resulting torques would depend on the uncertain parameters). Hence, the computation is based on the simplified model, i.e. without the modification introduced in (23)-(24). The fast-loop controller is described in Subsection 3.1. Since the two models do not use the same inputs (u and $\bar{u}$, respectively), R has to be redefined for the NE-controller computation. The following weights are chosen: 10 for the states x, y, z, ϕ and 0.5 for $\dot{x}$ and $\dot{y}$. The other states are not weighted. Though these weights result in a slightly better tracking performance than the weights given by Q_D, they are not crucial to the efficiency of the fast loop.

This control scheme exhibits a nice behavior as shown in Figure 3. No re-optimization is necessary in this simulation due to the robustness provided by the NE-controller in the fast loop. However, since the computation time needed to solve the optimization problem largely exceeds the final time, this control scheme can hardly be an option in practice, i.e. when re-optimization might become necessary.

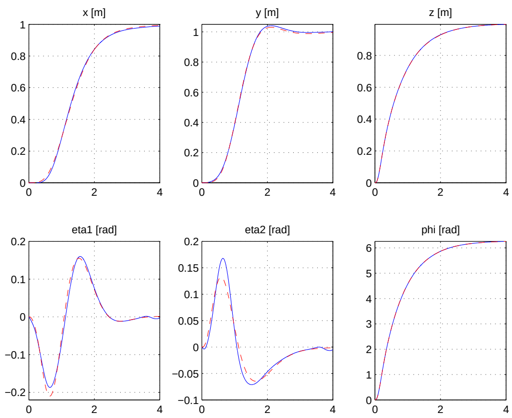

Fig. 3. Control based on MPC + NE-controller (reference trajectories: dashed line, system trajectories: plain line)

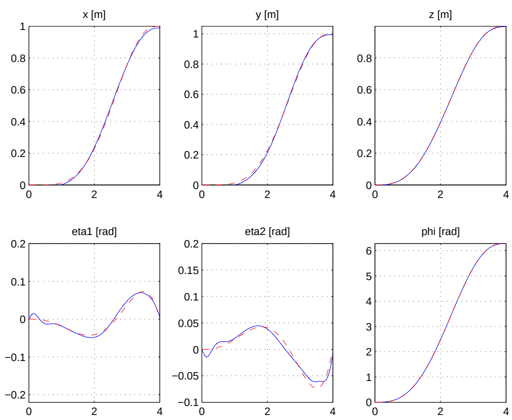

Fig. 4. Control based on system inversion + NE-controller (reference trajectories: dashed line, system trajectories: plain line)

4.6 Control based on system inversion + NE-controller

The flatness property of the simplified model allows generating the reference inputs and states algebraically, which reduces the computation time

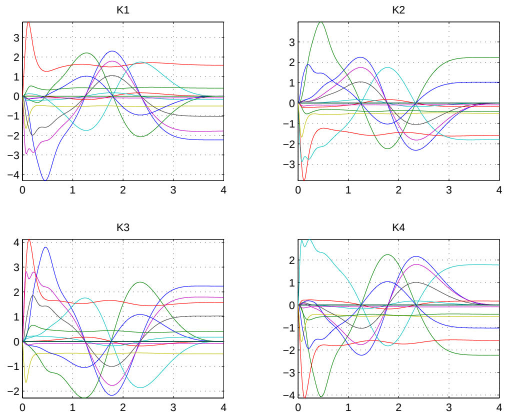

Fig. 5. Gains of the NE-controller for the flatness-based trajectory generation

significantly. The NE-controller in the fast loop ensures good tracking of the state references. The NE-controller is the same as in Sub-section 4.5.

This control scheme also exhibits a nice behavior as shown in Figure 4. The reference input and state trajectories are parametrized using polynomials. They are generated once for the whole trajectory, and no re-calculation is needed. The computation time for the flatness-based trajectory generation is fairly low. Figure 5 displays the gains of the NE-controller. Since the gains are strongly time-varying, the NE-controller cannot be approximated by a LQR.

5. CONCLUSION

This paper has proposed a two-time-scale control scheme that uses repeated trajectory generation in a slow loop and time-varying linear feedback based on the neighboring-extremal approach in a faster loop. The slow loop provides reasonable reference trajectories, while the fast loop ensures robustness. Trajectory generation has used two approaches: (i) optimization-based MPC, and (ii) flatness-based system inversion. Both approaches give good results, but the latter is significantly faster.

The proposed approaches as well as feedback-linearization and standard MPC have been used in simulation to control a VTOL flying structure. Though the simplified model of the structure is flat, control based on feedback linearization is not appropriate because of its lack of robustness with respect to the model uncertainties typically encountered in flying structures. Standard MPC requires a high re-optimization frequency and, in addition, cannot accommodate large model uncertainties. In contrast, the proposed two-time-scale control scheme is so robust that it does not require re-calculation of the reference trajectories. The optimization-based trajectory generation performs well, but is slow in comparison with the VTOL dynamics. The flatness-based approach appears to be sufficiently fast, and will be used for experimental implementation on a laboratory-scale VTOL structure, for which re-generation of the reference trajectories might be necessary.

Future work will also investigate the stability and robustness issues of the proposed two-time-scale control scheme.

REFERENCES

Bemporad, A. (1998). Reducing conservatism in predictive control of constrained systems with disturbances. In: *37th IEEE Control and Decision Conference*. Tampa, FL. pp. 1384–89.

Bemporad, A. and M. Morari (1999). *Robust Model Predictive Control: A Survey*. Springer Verlag.

Bryson, A. E. (1999). *Dynamic Optimization*. Addison-Wesley, Menlo Park, California.

Fliess, M., J. Lévine, Ph. Martin and P. Rouchon (1995). Flatness and defect of nonlinear systems: Introductory theory and examples". *International Journal of Control* **61**(6), 1327–1361.

Fliess, M., J. Lévine, Ph. Martin and P. Rouchon (1999). A Lie-Bäcklund approach to equivalence and flatness of nonlinear systems. *IEEE Trans. Automat. Contr.* **38**, 700–716.

Gros, S. (2005). Modeling and control of a VTOL structure. Rapport interne. Laboratoire d'Automatique, Ecole Polytechnique Fédérale de Lausanne.

Kouvaritakis, B., J. A. Rossiter and J. Schuurmans (2000). Efficient robust predictive control. *IEEE Trans. Automat. Contr.* **45**(8), 1545–49.

Lee, J. H. and Z. Yu (1997). Worst-case formulations of model-predictive control for systems with bounded parameters. *Automatica* **33**(5), 763–781.

Mayne, D. Q., J. B. Rawlings, C. V. Rao and P. O. M. Scokaert (2000). Constrained model predictive control: Stability and optimality. *Automatica* **36**(6), 789–814.

Morari, M. and J. H. Lee (1999). Model predictive control: Past, present, and future. *Comp. Chem. Eng.* **23**, 667–682.

Ronco, E., B. Srinivasan, J. Y. Favez and D. Bonvin (2001). Predictive control with added feedback for fast nonlinear systems. In: *European Control Conference*. Porto, Portugal. pp. 3167–3172.

Scokaert, P. O. and D. Q. Mayne (1998). Min-max feedback model predictive control for constrained linear systems. *IEEE Trans. Automat. Contr.* **43**, 1136–1142.