Automatic guidance of an aircraft using predictive control in a visual servoing scheme

Satja Lumbar*, Gregor Dolanc*, Darko Vrečko*
Stanko Strmčnik*, Drago Matko**

*Jozef Stefan Institute, Jamova 39, Ljubljana, Slovenia, EU (email: satja.lumbar@ijs.si)
**Faculty of Electrical Engineering, University of Ljubljana

Abstract: In this paper we have designed and implemented a predictive automatic flight controller for an airplane that takes as input the estimated pose obtained from the visual data and provides control outputs for an inner-loop augmented aircraft. In order to make the predictive controller suitable for flight control, it had to be modified. The most important modification was the introduction of the short term reference trajectories, linking the time-based and the position-based control strategy. Furthermore a controller switching mechanism was introduced to enable the aircraft to fly at different flight conditions. The parameters of the controller have been derived by experimental optimization. The tests have shown good overall tracking and disturbance rejection properties; the aircraft is able to perform various maneuvers and the touchdown in all the tested weather (wind, turbulence) conditions.

Keywords: model predictive control, automatic landing, flight control, visual servoing

NOMENCLATURE

\( \text{thr} \) – throttle command \[%\]
\( \text{spoil} \) – spoiler command \[rad\]
\( \text{ldg} \) – landing gear extracted \[0,1\]
\( \text{flaps} \) – flaps extension \[rad\]
\( c\text{Ail}, c\text{Elev}, c\text{Rud} \) – command for surface deflections \( \text{(Aileron, Elevator, Rudder)} \) \[rad\]
\( p, q, r \) – angular velocities \( \text{(roll, pitch, yaw)} \) \[rad/s\]
\( p\text{\_dem}, q\text{\_dem} \) – references for angular velocity \( \text{(roll, pitch)} \) \[rad/s\]
\( V\text{\_dem} \) – airspeed reference \[m/s\]
\( V \) – airspeed \[m/s\]
\( x \) – north geographical coordinate \( \text{(unmeasured)} \) \[m\]
\( y \) – east geographical coordinate \( \text{(unmeasured)} \) \[m\]
\( h \) – altitude above sea level \[m\]
\( AGL \) – altitude above ground level \[m\]
\( \text{pos=} \text{[x\_est, y\_est, h\_est]} \) – estimated geographical coordinates and altitude \[m\]
\( \text{image} \) – image captured with the vision sensor
\( \phi, \theta, \psi \) – Euler angles \[rad\]
\( \text{WP} \) – list of 3D waypoints \( \text{[(x\_wp, y\_wp, h\_wp, V\_wp, flaps\_wp)]} \)
\( \text{WP}_{\text{p}} \) – a waypoint that the aircraft is flying away from \((i\text{-th row of WP})\)
\( \text{WP}_{\text{d}} \) – destination waypoint, the aircraft is flying towards this point \((i+1\text{-th row of WP})\)
\( \text{STRh} \) – Short term trajectory reference vector for altitude \[m\]
\( \text{STRpsi} \) – Short term trajectory reference vector for heading \[rad\]
\( \text{STRv} \) – Short term trajectory reference vector for airspeed \[m/s\]

\( a_x, a_y, a_z \) – accelerations of the aircraft in the body axes \[m/s^2\]
\( v_x, v_y, v_z \) – velocities of the aircraft in the body axes \[m/s\]
\( v_x, v_y, v_z \) – velocities of the aircraft in the earth axes \[m/s\]

1. INTRODUCTION

In this paper a predictive position based visual servoing control scheme for automatic guidance of the aircraft in the terminal zone, where the position measurements are extracted from vision sensors mounted on the aircraft fuselage, will be presented. Today, the most common navigation systems available for landings, ground rolls and take-offs are the ILS (Instrument Landing System) and MLS (Microwave Landing System) which require airport infrastructures. In case of absence or malfunction of such infrastructure, no automatic guidance of the aircraft can be performed. This highlights the need for new systems which could either replace or complement the existing systems.

This topic has been dealt with in the scope of the PEGASE project (PEGASE, 2006), coordinated by Dassault Aviation. The main objective of this project was to perform a feasibility study and develop a simulation prototype of a system for automatic landing of aerial vehicles. The basic motivation was to design a new and completely autonomous system for automatic landing that does not depend on any kind of off-board equipment.

One of the most important parts of the PEGASE project was the design of visual servoing control (Hutchinson, 1996). The idea of this part was the development of the control subsystem that would take as input visual information from the vision sensors and guide the aircraft along the reference flight trajectory. The problem was approached by different groups using different methods and algorithms. The task of our research group was to solve this control problem by using some form of advanced control. Due to its ability of taking
into account future reference values and constraints on inputs, outputs and states, the model predictive control (MPC) (Maciejowski, 2002) strategy was selected.

With the development of fast dedicated hardware and efficient algorithms in recent years, predictive control has become more common also in mechanical systems with fast dynamics, also aircrafts (Heisse, 1996, Singh, 1996). The early studies (Hess, 1989, Jung, 1991) used predictive control for flight trajectory following where the aircraft is represented by a linear, discrete time model. In (Huzmazan, 1997) the control problem is solved by predictive control inner loop longitudinal controller and conventional outer loop controllers, while (Schram, 1997) uses classical controllers in inner loop and a predictive controller with a variable time interval in outer loop. A flight test of a predictive controller on an unmanned aerial vehicle is described in (Keviczky, 2005). In (Keviczky, 2006) linear, scheduled and nonlinear predictive approaches are compared with regard to the computational time consumed.

The aim of this paper is to present a MPC path following controller for an aircraft in order to be able to perform maneuvers, exclusively based on information that is available onboard the aircraft, equipped with vision sensors. The objective of this control problem is to provide altitude, airspeed and heading angle tracking autopilot using a pitch-rate and roll-rate inner loop controller.

The paper is organized as follows. The control scheme and all its elements are presented in Section 2, while special focus on the predictive controller is provided in Section 3. The results of the performance tests are discussed in Section 4, whereas some conclusions are given in Section 5.

2. CONTROL SCHEME

Fig. 1 represents a position based visual servoing (PBVS) control scheme for an inner loop augmented aircraft used to guide the aircraft along the reference flight trajectory.

The aircraft flight dynamics model that was used in the simulation is a detailed first principle nonlinear model. This model refers to a middle size business jet aircraft. Table 1 contains descriptions of inputs, outputs and states of this model.

<table>
<thead>
<tr>
<th>Name</th>
<th>Flight model (FAM)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td>$c_{A_{flight}}$, $c_{Elev}$, $c_{Rad}$, $spoil$, $flaps$, $thr$, $ldg$, $AGL$</td>
</tr>
<tr>
<td>Outputs</td>
<td>$V$, $p$, $q$, $r$, $phi$, $theta$, $psi$, $x$, $y$, $h$</td>
</tr>
<tr>
<td>States</td>
<td>$V_{x}$, $V_{y}$, $V_{z}$, $V_{wp}$, $V_{wp}$, $\alpha_{wp}$, $\alpha_{wp}$, $\alpha_{wp}$</td>
</tr>
</tbody>
</table>

2.2 Flight control system (FCS)

The nonlinear aircraft model was augmented with an inner loop classical controller based on pitch rate and roll rate feedback. A benefit of such augmentation is improved stability of the closed loop vehicle, which is particularly important when applying predictive control, since the output predictions of an unstable system can be numerically inaccurate and can cause numerical problems in optimization software (Maciejowski, 2002). Hence, an unstable prediction model should be avoided whenever possible. The inputs and outputs of this FCS are presented in Table 2.

<table>
<thead>
<tr>
<th>Name</th>
<th>Flight Control System (FCS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td>$V$, $p$, $q$, $r$, $h$, $p_{dem}$, $q_{dem}$</td>
</tr>
<tr>
<td>Outputs</td>
<td>$c_{A_{flight}}$, $c_{Elev}$, $c_{Rad}$</td>
</tr>
</tbody>
</table>

2.3 Short term trajectory reference (STR)

The independent variable in predictive control theory is time, but the reference flight trajectory is always position based. This underlines the need for a link between time based and position based control which is provided by the STR block. The inputs and outputs of this function are presented in Table 3. A detailed description of this block is given in Section 3.3.

<table>
<thead>
<tr>
<th>Name</th>
<th>Short term trajectory generator</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td>$WP_{r}$, $WP_{p}$, $WP_{y}$, $V_{dem}$</td>
</tr>
<tr>
<td>Outputs</td>
<td>$STRh$, $STRpsi$, $STRv$</td>
</tr>
</tbody>
</table>

2.4 Automatic flight control system (AFCS)

Automatic flight control system represents the outer loop controller of the cascade structure depicted on Fig. 1. Its basic task is to minimize the distance errors between aircraft position and the reference flight trajectory and also to minimize the error between the reference and the actual airspeed. Detailed description of the developed model predictive controller is given in next section. The proposed input/output/state structure is given in Table 4.

<table>
<thead>
<tr>
<th>Name</th>
<th>Automatic Flight Control System (AFCS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td>$STRh$, $STRpsi$, $STRv$, $x_{est}$, $y_{est}$, $h_{est}$, $phi_{est}$, $theta_{est}$, $psi_{est}$, $AGL$, $p$, $q$, $r$</td>
</tr>
</tbody>
</table>
2.5 Pose estimation algorithm

A pose estimation algorithm has a task of estimating the current position and orientation of the aircraft from the image acquired by the vision sensor. This is a complex, multistage computer vision procedure that represents one of the most important parts in the PEGASE project. Inputs and outputs of this block are presented in Table 5. In this paper we do not deal with image processing but consider that this preliminary step is already done, (see e.g. Marchand, 1999 for example).

### Table 5 – Inputs and outputs of the pose estimation algorithm block

<table>
<thead>
<tr>
<th>Name</th>
<th>Pose estimation algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs</td>
<td>Image of the environment</td>
</tr>
<tr>
<td>Outputs</td>
<td>x_est, y_est, h_est, phi_est, theta_est, psi_est</td>
</tr>
</tbody>
</table>

2.6 Vision sensor

Vision sensor (array) is mounted on the aircraft fuselage. The output of the sensor is an image of the environment passed to the pose estimation algorithm. Note that the sensor has to be able to acquire a usable image in all weather and visibility conditions. In the simulation environment sensors were supplemented by sensor models, which can include effects like quantization, sampling rate, color transformation, image noise, etc.

2.7 Flight Trajectory

Reference flight trajectory defines a flight path that the aircraft should follow as closely as possible. Reference flight trajectory is given in terms of 3 dimensional waypoints, defining the segments of the prescribed flight path. Each waypoint has prescribed reference airspeed V_dem and aircraft configuration (flaps and landing gear), which are valid for the next segment of the reference flight trajectory.

### 3. Model Predictive AFCS

#### 3.1 The basics of model predictive control

The MPC strategy is illustrated in Fig. 2, and consists of the following three steps.

**STEP 1:** Define \( \hat{y}(k+j) \) as the j-step ahead prediction of the process output based on data up to time \( k \). Determine \( \hat{y}(k+j) \) for \( j = 1, \ldots, H_p \) where \( H_p \) is the prediction horizon. These predictions of the process output will be a function of future control increments \( \Delta u(k+i) \) for \( i = 1, \ldots, H_c \), where \( \Delta \) denotes the difference operator: \( \Delta u(k+i) = u(k+i) - u(k + i - 1) \). \( H_c \) is called the control horizon and \( \Delta u(k+i) \) is assumed to be equal to zero for all \( i \geq H_c \).

**STEP 2:** Minimize a criterion function with respect to \( [\Delta u(k), \ldots, \Delta u(k+H_c-1)] \), subject to signal constraints. This criterion function will at least contain the difference between the predicted system outputs and their desired values which are specified by the reference trajectory \( w(k+j) \) for \( j = 1, \ldots, H_p \).

**STEP 3:** Apply the first control signal of the sequence to the system, displace the horizon one time-step towards the future and repeat the procedure.

In most predictive controllers the controller performance is specified by the following quadratic criterion function, which will also be used in this paper:

\[
J(k) = \frac{1}{2} \sum_{j=0}^{H_p} ||y(k+j) - w(k+j)||_R^2 + \sum_{i=0}^{H_c-1} ||\Delta u(k+i)||_Q^2.
\]

#### 3.2 Model predictive AFCS implementation

For the implementation of model predictive controller the Model Predictive Control toolbox (Bemporad, 2004) for MATLAB has been used. This is a collection of software that helps to design, analyze, and implement a predictive algorithm. An approximate, linear plant model is used to provide the predictions.

As it is common in flight control, the control system was divided into the longitudinal and the lateral controllers depicted in Fig. 3. The longitudinal controller is in charge of tracking altitude and velocity, influencing the pitch rate command on the FCS and the throttle command on FAM, while the lateral controller tracks the heading of the aircraft, with its output connected to the roll rate command of the FCS. The parameters of both controllers are gathered in Tables 6 and 7.

![Fig. 2 – MPC strategy principle](image)

![Fig. 3 – Block diagrams of the longitudinal and the lateral MPC controllers](image)

### Table 6 – Longitudinal controller

<table>
<thead>
<tr>
<th>Variable</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{STh} )</td>
<td>Longitudinal MPC controller</td>
</tr>
<tr>
<td>( \text{h}_{\text{est}} )</td>
<td>( q_{\text{dem}} )</td>
</tr>
<tr>
<td>( \text{V} )</td>
<td>( \text{hr} )</td>
</tr>
<tr>
<td>( \text{STPw} )</td>
<td>Lateral MPC controller</td>
</tr>
<tr>
<td>( \text{p}_{\text{est}} )</td>
<td>( p_{\text{dem}} )</td>
</tr>
</tbody>
</table>
3.3 Short term reference trajectories

studied in the frame of this research. Note that the subject of closed-loop stability has not been added in a manner described by (Maciejowski, 2002). Step disturbances and the measurement noise models have been In order to further increase the robustness qualities of the controllers has been improved. White noise models have been used for measurement output disturbance modeling while white noise models have been used for measurement noise modeling (Bemporad, 2004).

Table 7 – Lateral controller

| Parameters | Prediction horizon \( H_p = 50 \) | Control horizon (blocking) \( H_c = [10 \ 40] \) | Sample time \( T_s = 0.2 \) s |
| Weights | Weight on airspeed = 0.3 | Weight on altitude = 1 | Weight on demanded pitch rate = 10 |
| Constraints | Pitch angle interval = [-10...20] *π/180 rad | Pitch rate interval = [-0.4...0.4] rad/s | Demanded pitch rate interval = [10...40] rad/s |

Note that the parameters, weights and constraints have been derived by experimental optimization using a common simulation environment that was provided in the scope of the PEGASE project. This environment is able to run s-functions written with different programming languages, the most important being that of C++ and MATLAB.

Furthermore it can be observed that both controllers employ a blocking strategy on the control horizon. This means, that in optimization (1) it is considered that the calculated controller output will not only be applied in the next control sample, but rather in a block of next control samples. In our case we use the blocking strategy with the control horizon equal to \( H_c = [10 \ 40] \), which means that the first value of the manipulated variable will be held constant for 10 samples, and the second value of the manipulated variable for the next 40 samples. This strategy was employed due to the fact that the environment prescribed sample time was too small which resulted in a controller with a “nervous” output. Blocking strategy ensured the elimination of redundant control moves, effectively emulating a controller with a longer sampling time and shorter prediction horizon. In that sense the robustness of the controllers has been improved.

In order to further increase the robustness qualities of the controllers, the output disturbance models for nonmeasurable disturbances and the measurement noise models have been added in a manner described by (Maciejowski, 2002). Step models have been used for output disturbance modeling while white noise models have been used for measurement noise modeling (Bemporad, 2004).

Note that the subject of closed-loop stability has not been studied in the frame of this research.

3.4 Linearization of the inner loop augmented FAM

As previously explained, the Model Predictive Control Toolbox for MATLAB uses a linear model for predicting the future outputs and the states of the aircraft. For that purpose the linear models of the controlled plant are required. In the case of guiding an aircraft with an inner loop controller, the plant is represented with a connection of the FCS and the FAM.

The linearization has been done in two steps. Firstly the aircraft was trimmed in a specific flight condition. Secondly small perturbations on the relevant FCS and FAM inputs were applied. From responses of the perturbations the linear model was obtained. Note that in reality the experimental data needed for the linearization could be obtained from experimental flights of a real aircraft.

The linear models of the longitudinal and lateral FAM+PC models are depicted in Fig. 4:

Since the aircraft is a highly nonlinear system, one linear model cannot be equally accurate for all flight conditions, particularly in the longitudinal direction. To achieve a good prediction over the entire flight envelope, a set of different linear longitudinal models must be used and a mechanism for
switching these linear models online has to be applied. In our case controllers have been switched instead of models.

![Diagram of controllers]

Fig. 4 – Longitudinal and lateral linear model of the FAM+FCS

The principle of switching controllers is depicted in Fig. 5.

![Diagram of model switching mechanism]

Fig. 5 – Model switching mechanism

The controller switching mechanism is further explained in Fig 6. If we define two switching criterions, sw1 and sw2, with equations:

\[
sw1 = (WP_D - WP_p) \times (WP_D - p_{hp})^T, \tag{2}
\]

\[
sw2 = (WP_D - WP_p) \times (WP_D - pos)^T, \tag{3}
\]

where \(p_{hp}\) is the end point of the short term trajectory, then sw1 becomes negative when the STRs cross \(WP_D\) and sw2 becomes negative when the aircraft passes \(WP_D\).

![Diagram of controller switching mechanism]

Fig. 6 – Controller switching mechanism. The segments are defined by waypoints. The cross represents the current aircraft position and the dotted line represents the STRs.

When sw1 becomes negative, the second controller (AFCS_{s+1}) becomes active, but its outputs are unconnected. The controller is chosen based on its linear model, which has to be in accordance with the aircraft configuration and demanded airspeed of the next segment (defined in \(WP_n\)). During this time, the state of previous controller is being forced into the controller AFCS_{s+1}. When sw2 becomes negative, controller AFCS_{s+1} becomes active, and controller AFCS, is deactivated.

Note that the mechanism described above guarantees a bumpless switch of controllers only in the case of continuous (smooth) change in flight condition (airspeed, altitude). In case of discrete events (flaps extension, landing gear extension) additional compensation of the consequential disturbance would be needed. Development of a mechanism for elimination of the consequences of discontinuous changes in flight condition represents an additional research problem which is beyond the scope of this paper.

4. TESTS AND RESULTS

Having the controller implemented some tests to evaluate the quality of reference flight trajectory tracking have been performed. Results of two tests will be presented in this paper. All the tests were performed in simulated synthetic environment of Marseille Marignane airport.

Fig. 7 depicts the tracking of airspeed, altitude and heading over the whole prescribed flight trajectory. Fig. 8 shows the outputs of the FCS (\(p_{dem}, q_{dem}, thr\)) and tracking of angular rates \(p\) and \(q\).

![Graphs of tracking results]

Fig. 7 – Tracking of reference flight trajectory. Top: heading angle (bird view); Middle: altitude; Bottom: airspeed. Reference is marked with red color; tracked signals are marked with blue colour

We can see that with the use of the predictive controller a very good quality of tracking the reference flight trajectory has been achieved. The predictive controller tracks all three signals from beginning to the touchdown without significant deviations from the reference. The maximum deviations occur when the effect of extending the flaps, making a turn and applying large step in airspeed all happen at the same time instant. Largest altitude error is smaller than 30 meters, largest lateral error is smaller than 50 meters and largest airspeed error is smaller than 2 meters per second.

Fig. 9 is a close-up of the first subplot in Fig. 7. It depicts the reaction of the aircraft equipped with the predictive AFCS in a lateral turn.

Note the pre-emptive reaction of the predictive AFCS to a lateral turn in the reference flight trajectory. The aircraft starts the turn, when the short term reference trajectory for heading \((STR_{psi})\) crosses the waypoint instead of starting the turn when the aircraft crosses this waypoint. This results in
smaller tracking error in lateral turns and therefore better tracking properties of the controller.

The second test is composed of only the touchdown segment of the landing scenario with front wind of 3 m/s and a Dryden turbulence atmospheric model (Hoblit, 1988). Fig. 10 depicts the flare maneuver and touchdown of the aircraft under these deteriorated weather conditions.

We can see that in presence of wind and turbulence the real flight trajectory of the aircraft is less smooth than before, but this non-smoothness is really insignificant. The flare maneuver and touchdown are still successfully performed. This leads us to believe that the predictive controller has the necessary robustness to perform well in case of deteriorated weather conditions. Note that the runway is not horizontal. Therefore the aircraft goes slightly upwards after touchdown.

In the end it should be mentioned, that all the tests were performed under non-ideal visible conditions with mild rain and fog present. This affects the level of noise in the vision algorithms and therefore the quality of position estimates. Further deterioration of visibility conditions could lead to worse control properties, but the analysis of this phenomenon is beyond the scope of this paper.

5. CONCLUSIONS

It has been shown that using MPC in the frame of position based visual servoing control scheme renders good results in terms of tracking and disturbance rejection properties. Our future work will be devoted to further improvements of the presented algorithm and to the design of an image based visual servoing algorithm in order to guide an aircraft without knowing its current pose (no pose estimation).

ACKNOWLEDGEMENT

The PEGASE project was financially supported by EC under contract No. AST5-CT-2006-030939. The authors are also grateful for financial support of Slovenian Research Agency.

REFERENCES


Keviczky T., Balas G.J. (2006), Receding horizon control on F-16 aircraft: A comparative study, *Control Engineering Practice*, 14, 1023-1033

Maciejowski J.M. (2002). *Predictive control with constraints*, Prentice Hall, New Jersey, USA


