CEAS EuroGNC 2019
"Conference on Guidance, Navigation and Control"
3-5 April 2019 @ Politecnico di Milano, Milan, Italy.

CEAS-GNC-2019-041

Closed-loop MIMO data-driven attitude control design for a multirotor UAV

Angelo Zangarini, Davide Invernizzi, Pietro Panizza, Marco Lovera

Abstract Data-driven controller design methods allow a fast tuning of controller parameters directly from data, relying on limited prior knowledge of the plant dynamics. In this paper, the problem of tuning the attitude control system of a multirotor UAV is tackled and a data-driven approach is proposed. With respect to previous work, in this paper data collected in flight, during closed-loop experiments, is used to tune the controller gains. Furthermore, the simultaneous tuning of roll and pitch attitude control loops is demonstrated, thus paving the way to MIMO data-driven attitude control design. The results, based on experimental work carried out on a quadrotor UAV, show that a performance level comparable to that of model-based methods can be achieved.

1 Introduction

Small-scale Unmanned Aerial Vehicles (UAVs), and in particular multirotor ones, have been studied extensively in view of the great potential for a large number of applications. For most problems of practical interest, requirements in terms of pointing and positioning performance require a careful tuning of the control laws. While nonlinear control design approaches have been considered in the literature (see, *e.g.*,[1] for a recent survey), for civil applications such as surveillance, mapping, video and

Angelo Zangarini
Politecnico di Milano, e-mail: angelo.zangarini@mail.polimi.it

Davide Invernizzi
Politecnico di Milano, e-mail: davide.invernizzi@polimi.it

Pietro Panizza
Politecnico di Milano, e-mail: pietro.panizza@polimi.it

Marco Lovera
Politecnico di Milano, e-mail: marco.lovera@polimi.it

photography linear controllers are usually adopted. In these settings, considering also that hover and near-hover operations are representative conditions, cascaded PID laws are usually employed for attitude control thanks to their inherent reliability and ease of implementation. As far as controller tuning is concerned, model-based methods suffer from the fact that the mathematical modelling of quadrotors is particularly challenging due to the non-trivial characterization of the aerodynamics and of the actuators and sensors dynamics (see[2]). For this reason data-driven tuning methods, which have been developed in the last two decades in the control community, offer an interesting alternative. These control design tools are especially appealing when *a priori* knowledge about the plant model is limited, when an accurate modeling of the system is too expensive or when fast deployment of the control system is an important requirement, since they allow the direct tuning of the controller parameters from experimental input-output data. Among the data-driven methods available in the literature, a coarse classification can be made between iterative (*e.g.*, the Iterative Feedback Tuning (IFT)[3]) and single-shot (non-iterative) methods (*e.g.*, the Virtual Reference Feedback Tuning (VRFT)[4], the Correlation-Based Tuning (CbT)[5,6]). Non-iterative methods are particularly attractive for a fast re-tuning of the controller when the plant performance is reduced (*e.g.*, components aging) and/or operating conditions change (*e.g.*, different payloads, environment). Recently ([7]) the VRFT algorithm has been considered to tune the attitude controller parameters of a variable-pitch quadrotor, based on data collected in indoor experiments on a single degree-of-freedom test-bed. The results have shown improvements in the tracking and disturbance rejection capabilities compared to those obtained with a manual tuning. Furthermore, comparable results with respect to a model-based structured H_∞ synthesis ([8]), made data-driven methods a promising tool for this kind of applications. In particular, an extension of VRFT allowing the direct tuning of a cascade controller configuration with a single set of input-output data, following the precedure outlined for the VRFT (see[9]), has been employed. The possibility of tuning the control laws directly from flight-test data has been subsequently explored in[10], as this, among other things, would pave the way to the design and tuning of MIMO attitude controllers. Experiments for the tuning of attitude controllers however can be executed safely only in closed-loop conditions. In view of this, in this paper a closed-loop approach to data-driven tuning of the attitude control laws for a multirotor UAV is presented. With respect to previous work, the pitch and roll axes are tuned in a single experiment. The achievable performance is illustrated by means of experimental results obtained on a small-scale quadrotor.

The paper is organized as follows. The data-driven framework is presented in Section 2. In Section 3.1 the considered quadrotor platform and its controller architecture are introduced in detail. Finally, simulation and experimental results are presented and discussed in Section 3.3 and 3.4 respectively.

2 Data-driven control law design

2.1 VRFT with open-loop data

Consider a linear time-invariant discrete-time system $G(z)$, where z denotes the forward time-shift unit (*i.e.*, $zx(t) = x(t+1)$), a parametrized controller class $\mathscr{C}(\theta) = \{C(z,\theta), \theta \in \mathbb{R}^n\}$, and a given target closed-loop behaviour $M(z)$. The control aim of data-driven methods is the minimization of the weighted $\mathscr{L}_2$-norm of the mismatch between $M(z)$ and the actual closed-loop system:

$$J_{MR}(\theta) = \left\| \left(\frac{G(z)C(z,\theta)}{1+G(z)C(z,\theta)} - M(z) \right) W(z) \right\|_2^2, \tag{1}$$

where $W(z)$ is a weighting function chosen by the user. In data-driven approaches the model-reference problem (1) is solved with limited knowledge of the system and using only a set of available measurements $d_N = \{u(t), y(t)\}_{t=1..N}$, where N is the length of the data-set.

The main idea of VRFT can be described as follows. Consider the reference signal $r(t)$ that would feed the system in closed-loop operation when the closed-loop model is $M(z)$ and the output is the measured $y(t)$. Such a signal is called *virtual reference* and can be computed off-line from the output data as $r(t) = M^{-1}(z)y(t)$. A good controller (making the closed-loop as close as possible to $M(z)$) is then the one that produces the input sequence of the experiment $u(t)$ when it is fed by the error signal $e(t) = r(t) - y(t)$.

Formally, the cost criterion minimized by the VRFT algorithm is the following:

$$J_{VR}^N(\theta) = \frac{1}{N}\sum_{t=1}^{N} \left(u_L(t) - C(z,\theta)e_L(t)\right)^2, \tag{2}$$

where $u_L(t)$ and $e_L(t)$ are suitably filtered versions of $u(t)$ and $e(t)$. The filter $L(z)$ is chosen such that the cost function (2) is a local approximation of the criterion (1) in the neighborhood of the minimum point ([4]). Recent advances on the VRFT method can be found, *e.g.*, in [11,12,13], while application studies are available, *e.g.*, in [14,15].

Remark 1. Both VRFT and CbT have been extended to deal with multiple nested loops architectures in [9,16]. Consider the cascade control scheme in Figure 1, given two reference models $M_i(z)$ and $M_o(z)$, for the inner loop and the outer loop respectively, and consider two families of linear proper controllers $\mathscr{C}_i(\Theta_i) = \{C_i(z,\Theta_i)\ ,\ \Theta_i \in R_i^n\}$ and $\mathscr{C}_o(\Theta_o) = \{C_o(z,\Theta_o)\ ,\ \Theta_o \in R_o^n\}$ and the set of data $D_N = \{u(t), y_i(t), y_o(t)\}_{t=1,\ldots,N}$ being $u(t)$ the control variable, $y_i(t)$ the output of the inner loop, $y_o(t)$ the output of the outer loop. The inner controller can be tuned by applying VRFT or CbT while for the outer controller the approach needs to be different, as the input of the system to control is the reference $r_i(t)$, that is not available in the dataset, since measurements are collected during open-loop operation. Nevertheless, the reference signal

$r_i(t)$ can be derived as follows: once $C_i(z,\Theta_i)$ is fixed, the input of the inner loop can be calculated as $r_i(t) = e_i(t) + y_i(t)$, where the tracking error comes from the result of the inner design: $e_i(t) = C_i^{-1}(z,\Theta_i)u(t)$. With such a choice, $r_i(t)$ is exactly the signal that would feed the inner loop in closed-loop working conditions when the output is $y_i(t)$. Then, the outer controller can be easily found by using the set of I/O data $D_N^o = \{r_i(t), y_o(t)\}_{t=1,\dots,N}$.

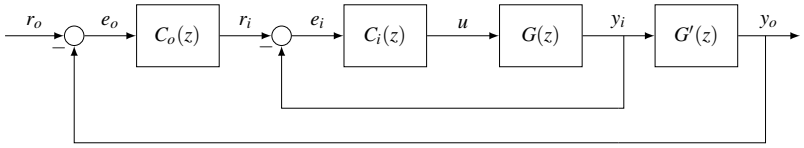


Fig. 1. Cascade control scheme with two nested loops.

2.2 VRFT with closed-loop data

If the test to collect data is performed in flight, then for safety reasons the data must be collected in closed-loop, allowing the user to control the system also during the experiment. Furthermore, closed-loop tests allow to perform the experiment to collect the data without exploiting a test-bed and without modifying the system, thus significantly simplifying the tuning process.

As illustrated in Figure 2, the excitation input $\bar{u}$ is added to the output of the controller $C_d(z)$. $C_d(z)$ is a stabilising controller adopted to carry out the in-flight test. The user can act on the set-point r to control the behaviour of the system also during the experiment.

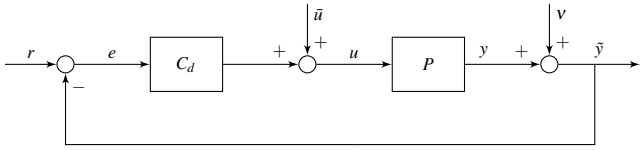


Fig. 2. VRFT experiment in closed-loop operation.

The standard VRFT method cannot be applied to obtain a new controller exploiting the measurements $d_N = \{u(t), \tilde{y}(t)\}_{t=1,\dots,N}$: specific problems arise when the instrumental variable is constructed because u and v are now correlated. Indeed, the user cannot directly act on the input of the plant as in the standard VRFT, but he can operate on the setpoint r and on the excitation input $\bar{u}$, and the input of the plant is now affected by this action:

$$u(t) = \frac{1}{1+C_d(z)P(z)}\bar{u}(t) + \frac{C_d(z)}{1+C_d(z)P(z)}\left(r(t)-\nu(t)\right). \tag{3}$$

For the sake of simplicity, the assumption that the user does not provide a setpoint during the experiment can be made ($r(t) = 0, \quad \forall\, t$) and (3) can be rewritten as:

$$u(t) = \frac{1}{1+C_d(z)P(z)}\bar{u}(t) - \frac{C_d(z)}{1+C_d(z)P(z)}\nu(t). \tag{4}$$

Using (4) to build the instrumental variable leads to a biased controller parameter vector since the instrumental variable is no longer uncorrelated with the noise $\nu(t)$. Indeed, if the instrumental variable is built as $\hat{y}(t) = \hat{P}(z)u(t)$, then using (4) we get

$$\hat{y}(t) = \hat{P}(z)\left(\frac{1}{1+C_d(z)P(z)}\bar{u}(t) - \frac{C_d(z)}{1+C_d(z)P(z)}\nu(t)\right).$$

Following[4], the instrumental variable is given by

$$\begin{aligned}\zeta(t) &= \beta(z)L(z)\left(M(z)^{-1}-1\right)\hat{y}(t)\\ &= \beta(z)L(z)\left(M(z)^{-1}-1\right)\\ &\quad \hat{P}(z)\left(\frac{1}{1+C_d(z)P(z)}\bar{u}(t) - \frac{C_d(z)}{1+C_d(z)P(z)}\nu(t)\right).\end{aligned}$$

The previous equation clearly shows the correlation between $\zeta(t)$ and $\nu(t)$. To solve this problem a different instrumental variable must be chosen, to ensure correlation with the regression variable and incorrelation with the noise. A detailed overview on the choice of the instrumental variable can be found in[10].

2.3 Multivariable extension

The VRFT algorithm can be extended to the multivariable case[17], where the initial formulation is the same, but an additional step is introduced. A different instrumental variable method is employed, the extended instrumental variable (EIV), which is easily implemented for multivariable problems[18].

The discrete MIMO LTI problem requires a redefinition of the state matrices and transfer functions, where $u \in \mathbb{R}^{n_u}$, $y, r \in \mathbb{R}^{n_y}$, $G(z) \in \mathbb{R}^{n_y \times n_u}$, $C(z) \in \mathbb{R}^{n_u \times n_y}$, $T(z) \in \mathbb{R}^{n_y \times n_y}$. The model reference problem is then formulated with respect to the input complementary sensitivity $T(z)$, thus the reference model is such that $M(z) \in \mathbb{R}^{n_y \times n_y}$.

By analysing the frequency-wise counterpart of the cost function defined in (2) and the model reference one in Equation (1), the filters which make them equivalent can be defined as:

$$L_u(z) = M(z)\Phi_{uu}^{-1/2}(z), \quad L_e = C^{-1}(z,\theta)M(z), \quad L_y = C(z,\theta)\Phi_{uu}^{-1/2}(z), \tag{5}$$

where Φ_{uu} is the power spectral density of the input u. These filters however require the knowledge of the controller, turning the problem into a nonlinear one. A convex approximation of the problem can be achieved by making the following assumptions:

1. the sensitivity function $S(z) = I - M(z)$ is close to the closed-loop sensitivity function for $\theta = \hat{\theta}$;
2. the controller family $C(z,\theta)$ can be linearly parametrized with the vectors of parameters $\theta \in \mathbb{R}^n$, such that $\mathscr{C}(\theta) = \left\{C(z,\theta) = \beta^T(z)\theta\right\}$.

Finally, by replacing these assumptions in Equation (1), the cost function becomes:

$$J_{MR}(\theta) = \left\| M(z) - (I - M(z))\,\beta^T(z)\theta \right\|_2^2. \tag{6}$$

By choosing the above filters as:

$$L_u(z) = L_e(z) = L(z) = M(z) and L_y(z) = I, \tag{7}$$

the cost function becomes linear in the parameter vector, and is equivalent to the approximated model reference cost function shown in Equation (6).

It is noted that the filters for the MIMO extension differ from the ones derived for the SISO problem, as obtained in[4]. This is due to Assumption 1 being used at the beginning of the derivation, obtaining the filter for the convex model reference problem instead of deriving the optimal filter first for the original model reference problem.

The structure of the regressor vector $\varphi(k)$ must be defined. Another parametrization of the controller class is introduced for the MIMO problem, such that:

$$u(k) = u(k-1) + \sum_{i=0}^{n} B_i e(k-i) \tag{8}$$

$$= u(k-1) + B_0 e(k) + B_1 e(k-1) + \cdots + B_n e(k-n), \tag{9}$$

where $B_i \in \mathbb{R}^{n_u \times n_y}, i = 1,\ldots,n$. The linearly parametrized PID class can be obtained by exploiting the properties of the Kronecker product, denoted with $\otimes$, as follows:

$$u(k) = u(k-1) + \sum_{i=0}^{n} B_i e(k-i) = u(k-1) + \varphi^T(k)\theta \tag{10}$$

$$\begin{aligned} \sum_{i=0}^{n} B_i e(k-i) &= \left[e^T(k)\otimes I, \cdots, e^T(k-n)\otimes I\right]\ \mathrm{vec}\left([B_0,\cdots,B_n]\right) \\ &= \varphi^T(k)\theta, \end{aligned} \tag{11}$$

where:

$$\theta = \mathrm{vec}\left([B_0,\cdots B_n]\right) \tag{12}$$

$$\varphi(k) = \left[e^T(k)\otimes I, \cdots, e^T(k-n)\otimes I\right]^T. \tag{13}$$

The definition of the regressor φ and the parameter vector $\theta \in \mathbb{R}^{n_\theta}, n_\theta = n \times n_u \times n_y$ in Equation (10) can be further manipulated obtaining:

$$u(z) = \frac{1}{1-z^{-1}}\varphi^T(z)\theta = \frac{z}{z-1}\varphi^T(z)\theta = \varphi_F^T(z)\theta. \tag{14}$$

Now the Extended Instrumental Variable (EIV) is added to develop a new cost function. Unlike the SISO case, the length of the instrumental variable is not the same as the length of the input vector. This is due to the fact that it is not built using data from another experiment, as presented in Section 2.2. In the case of EIV, the same control input is used with a window of length $\pm\, l$:

$$\zeta(k) = \begin{Bmatrix} u(k+l) \\ \vdots \\ u(k-l) \end{Bmatrix} \qquad \zeta_L(k) = \begin{Bmatrix} u_L(k+l) \\ \vdots \\ u_L(k-l) \end{Bmatrix}. \tag{15}$$

Said instrumental variable can now be used to define a decorrelation cost function, as described in[18]:

$$J_D(\theta) = (r - R\theta)^T \hat{W}^{-1}(r - R\theta) \tag{16}$$

$$R = \frac{1}{N}\sum_{k=1}^{N} \zeta_L(k) \otimes \varphi_L(k) \tag{17}$$

$$r = \frac{1}{N}\sum_{k=1}^{N} \zeta_L(k) \otimes u_L(k), \tag{18}$$

where φ_L is the regressor defined from signals filtered with (7) and $\hat{W}$ is a positive semi-definite weight, optimally a consistent estimate of the residual covariance matrix $\bar{W}$:

$$\bar{W} = \mathbb{E}\left[(r - R\theta)(r - R\theta)^T\right]. \tag{19}$$

The decorrelation function in the absence of noise, for large windows l, leads asymptotically to $R\theta - r = 0$. Thus, the minima of the decorrelation cost function (16) are equivalent to the minima of the virtual reference cost function (2), and are given by:

$$\hat{\theta} = \arg\min_{\theta} J_D(\theta) = \left(R^T W^{-1} R\right)^{-1}\left(R^T W^{-1} r\right). \tag{20}$$

The length of the window for the EIV method represents a tuning knob of the algorithm, however an arbitrarily large number can be used.

3 Results

3.1 Multirotor platform

The considered multirotor platform, called ADAM-0 (see Figure 3), is a fixed-pitch quadrotor with the following characteristics:

- Take-Off Weight (TOW): approximately 1450 grams;
- Battery: 4S Li-Po 4000 mAh;
- Flight time: 12 minutes;
- Frame dimensions (footprint): 500 mm (excluding rotors).

As is common practice in the multirotor literature, as far as linear controllers are concerned symmetry arguments are used to reduce the attitude control problem to a set of three separate problems for, respectively, the pitch, roll and yaw axes. In this work the pitch and roll attitude controllers of the ADAM-0 quadrotor are first tuned using the SISO algorithm with individual data.

Fig. 3. The ADAM-0 UAV.

3.2 Controller structure

Concerning the control architecture, the ADAM-0 platform adopts an attitude control scheme based on decoupled cascaded PID loops for the pitch, roll and yaw axes, running at 250 Hz. Focusing on the pitch axis, the outer loop (measured angle ϑ, set-point ϑ^o) is a P controller, while the inner controller is a complete PID with

an additional feed-forward term. More specifically, the feed-forward gain is directly computed on the pitch angle set-point and the derivative action of the inner loop is computed starting from the pitch rate q and not from the pitch angular rate error (see the block diagram in Figure 4, where the pitch control loop is represented).

This structure can be converted into the regressor form in equation (10) in a straightforward step, resulting in:

$$u(k) = u(k-1) + \sum_{i=0}^{n_e} B_i e(k-i) + \sum_{j=0}^{n_y} B_j y(k-j) + \sum_{m=0}^{n_r} B_m r(k-m) \tag{21}$$

$$= u(k-1) + \varphi^T(k)\theta \tag{22}$$

The default parameters of the controller are shown in Table 1, which will be used for the experiment that follow.

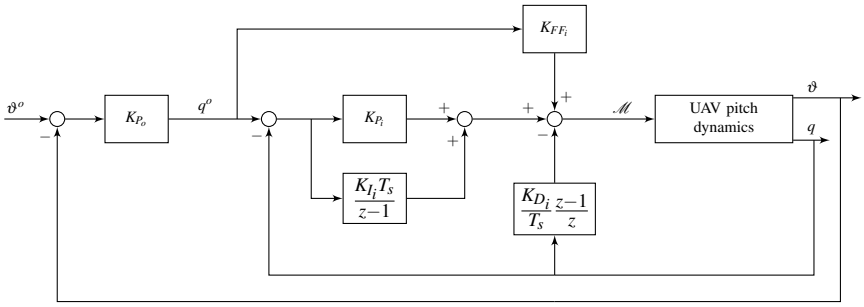


Fig. 4. Block diagram of the pitch control system, with a feed-forward gain and a derivative action based on angular rate feedback.

Table 1. ADAM-0: default controller parameters.

$\mathbf{K_P^O}$	$\mathbf{K_{FF}^I}$	$\mathbf{K_P^I}$	$\mathbf{K_I^I}$	$\mathbf{K_D^I}$
$\begin{bmatrix} 6.5 & 0 \\ 0 & 6.5 \end{bmatrix}$	$\begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 0.15 & 0 \\ 0 & 0.15 \end{bmatrix}$	$\begin{bmatrix} 0.05 & 0 \\ 0 & 0.05 \end{bmatrix}$	$\begin{bmatrix} 0.003 & 0 \\ 0 & 0.003 \end{bmatrix}$

Until now only the pitch attitude controller was considered. As is common practice in multirotor UAVs, the roll Degree of Freedom (DoF) is controlled with the same regulator scheme in Figure 4 thanks to the geometrical symmetry of the quadrotor. Obviously, the involved signals are different: the user provides the roll setpoint $\phi^o(t)$ and the proportional outer controller generates the roll angular rate reference signal. The inner regulator, starting from this information, computes the roll pitch moment $\mathscr{L}(t)$.

This decoupled architecture for the pitch and roll axes is justified by the fact that if the body axes are principal axes of inertia, then when the quadrotor is in near-

hovering conditions the roll and pitch DoFs could be assumed decoupled. This, in turn, implies that the pitch and the roll can be tuned independently. If, however, the system does not have decoupled attitude dynamics between the DoFs, the tuning problem for the different DoFs is coupled and the corresponding controllers must be tuned at the same time.

Amongst data-driven techniques, the Virtual Reference Feedback Tuning (VRFT) method has been considered. As described in Section 2, this method is not limited to Single Input Single Output (SISO) control systems and VRFT can deal also with Multiple Input Multiple Output (MIMO) regulators.

In this section only the inner loop controllers, based on the roll and pitch angular feedbacks, are taken into account. In particular the considered controller has four inputs (pitch and roll angular setpoints, and pitch and roll angular rate measurements) and two outputs (pitch and roll moments). Internally it has four independent regulators as displayed in Figure 4: one for the pitch DoF, one for the roll DoF and two for the coupled dynamics. In this case the controller parameters are not scalar but they are 2×2 matrices: on the main diagonal there are the parameters for the pitch and the roll DoF controllers and on the secondary diagonal the parameters of the coupled dynamics controllers.

3.3 Simulation results

A complete Simulink ADAM-0 nonlinear simulator has been used to validate the results of the algorithm. The simulator is able to replicate the attitude dynamics under feedback control on all axes. An artificial inertial coupling has been introduced, where the off-diagonal terms represent 10% of the diagonal terms. These terms will introduce gyroscopic effects since the pitch and roll axis are no longer principal axes of inertia.

Simulation data is collected in closed-loop in order to create the input and output dataset required for the VRFT algorithm. Two Pseudo Random Binary Sequence (PRBS) excitation signals, one for the pitching moment and one for the rolling moment, are applied consecutively. The input $\bar{u}(t) = \{\bar{\mathscr{L}}, \bar{\mathscr{M}}\}^T$ is injected in the system as shown in Figure 2. The two signals are different but they share the same PRBS parameters (signal amplitude and min/max switching interval). In this case, for each axis, a total excitation time of 20 s has been used, with a frequency of 50 rad/s and an amplitude of 0.15. The latter is a non-dimensional amplitude, referred to the maximum moment that can be applied.

As illustrated in Section 2.2, an initial controller $C_0(z)$ that stabilizes the system must be available, with parameters collected in Table 1.

Reference models

For both the pitch and roll inner loops, the reference model is a second order model, with a desired bandwidth and damping ratio of 20 rad/s and 0.4 respectively:

$$M_i(z) = \frac{0.003131z + 0.003065}{z^2 - 1.932z + 0.9380}.$$

In this specific case no filtering action was needed, thus the weighting function has been defined as $W_i(z) = I$. Therefore, considering the MIMO case the reference models are 2×2 matrices of transfer functions, with the transfer function $M_i(z)$ on the main diagonal and zeros on the secondary diagonal, requiring full decoupling.

Similarly, requirements have been set for the outer loop, once again a second order model, and a slower response. The desired bandwidth is 10 rad/s with a damping ratio of 0.7:

$$M_o(z) = \frac{0.0007852z + 0.0007706}{z^2 - 1.9440z + 0.9455}.$$

Controller parameters comparison

Results from the SISO algorithm applied to both pitch and roll data are then compared to the full MIMO formulation shown in Section 2.3, for the given set of reference models for the inner and outer dynamics. Noise has been introduced in the system, modelled as white noise with a standard deviation obtained from hovering endurance tests to account for the uncertainty of the state estimates.

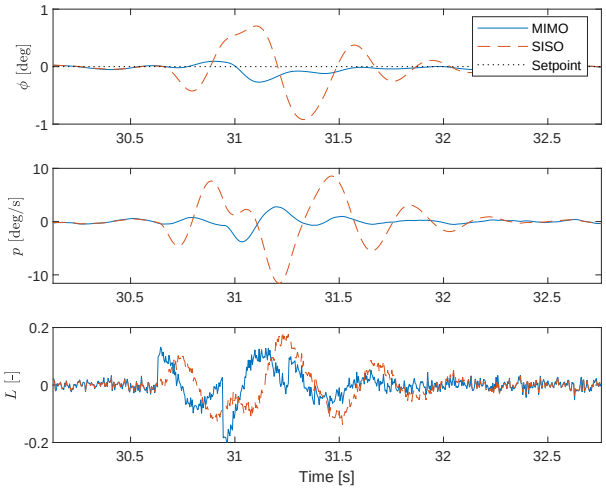
The resulting parameters of both algorithms are illustrated in Table 2. It is showed that the diagonal terms are almost the same. The gains of the outer loop P controller feature an almost identical term on the diagonal, and the outer diagonal terms which are smaller by two orders of magnitude. This means that the inner loop controller is able to decouple effectively the dynamics from the attitude rate and making them similar.

Table 2. ADAM-0: optimal controller parameters for outer and inner controllers considering the VRFT method with closed-loop simulation data.

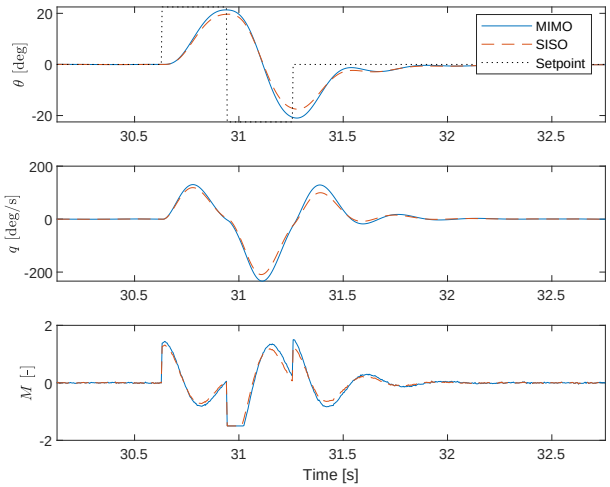
	$\mathbf{K_P^O}$	$\mathbf{K_P^I}$	$\mathbf{K_I^I}$	$\mathbf{K_D^I}$
MIMO	$\begin{bmatrix} 5.2090 & 0.0104 \\ 0.0114 & 5.1874 \end{bmatrix}$	$\begin{bmatrix} 0.1464 & 0.0094 \\ 0.0094 & 0.1275 \end{bmatrix}$	$\begin{bmatrix} 0.2630 & 0.0038 \\ 0.0038 & 0.2555 \end{bmatrix}$	$\begin{bmatrix} 0.0005 & 0 \\ 0 & 0.0004 \end{bmatrix}$
SISO	$\begin{bmatrix} 4.6650 & 0 \\ 0 & 4.6772 \end{bmatrix}$	$\begin{bmatrix} 0.1490 & 0 \\ 0 & 0.1300 \end{bmatrix}$	$\begin{bmatrix} 0.1765 & 0 \\ 0 & 0.1807 \end{bmatrix}$	$\begin{bmatrix} 0.0001 & 0 \\ 0 & 0.0001 \end{bmatrix}$

In order to compare the results, a doublet benchmark has been considered, that is a quick consecutive variation of the attitude that has a zero mean. The doublet period $T = 0.42s$ and amplitude $A = 22.5$ deg is held constant amongst tests.

The simulated pitch doublet is shown in Figure 5, where it is highlighted that the full MIMO controller is able to significantly reduce the coupling effects. It can be seen that the control effort is comparable, while achieving a better attitude tracking.



(a) Roll response



(b) Pitch response

Fig. 5. Simulation of pitch attitude doublet

3.4 Experimental results

The experimental data is collected in the same way as presented for the simulation result. The default controller parameters are used as the initial regulator, illustrated in Table 1.

Figure 6a and Figure 6b show the involved signals in the data-driven tuning procedure. These signals share the same specifications of the simulated experiment in Section 3.3. For the sake of clarity, the signals are represented in two figures but were collected sequentially during the same flight.

Reference models

As seen in Section 3.3, the chosen reference models are second order models with the addition of a delay. The choice of the reference model in data-driven methods can affect the stability of the feedback system, thus it might need adjustments between tests and lead to slightly different results. In this case, the reference model chosen for the SISO algorithm is different from the one used for the simulation and MIMO formulation.

Table 3. Experimental data model references for the inner and outer loops.

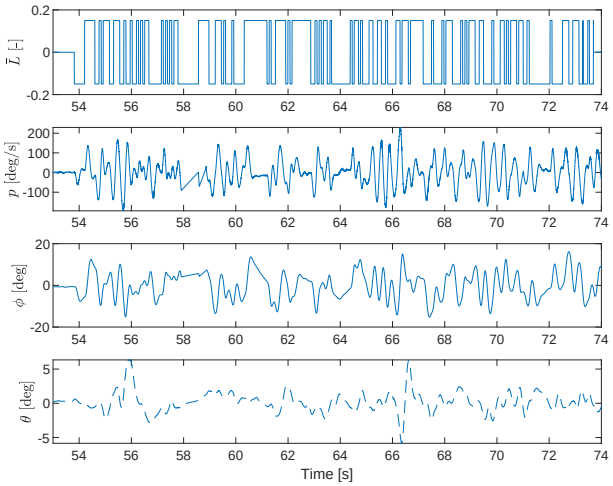
		ω **[rad/s]**	ζ	**Delay**
MIMO	Inner loop	18	0.3	1
	Outer loop	10	0.8	1
SISO	Inner loop	20	0.4	3
	Outer loop	10	0.7	3

Controller parameter comparison

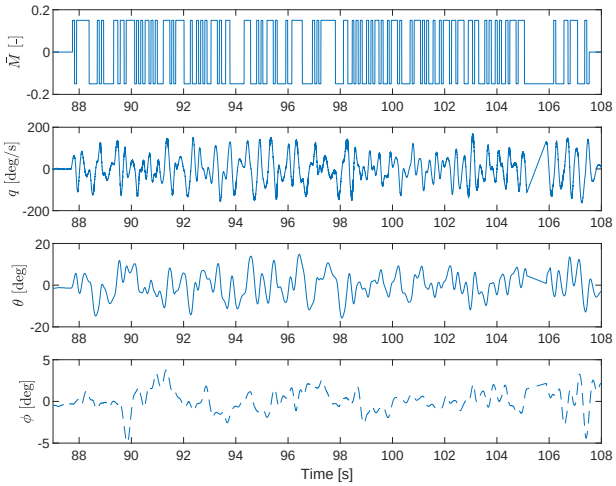
The benchmark for the performance comparison is a doublet, with period $T = 0.45s$ and amplitude $A = 40$ deg, similarly to the simulation section. Exploiting the reference models and closed-loop experimental data, the VRFT method leads to the parameter values reported in Table 4.

Since the doublet experiment requires the position and velocity outer feedback loops to be disabled, experiments have been carried out manually by a pilot, leading to difficulties in replicating the exact input and conditions. For the sake of brevity, only one of the excitations is shown, as in this platform the results are identical.

Preliminary tests are shown in Figures 7 and 8, where the roll doublet response shows a coupling, which in both cases is very limited, as expected. The pitch angle variations are about $\pm$ 3 deg, however they are smoother for the MIMO solution.



(a) Roll excitation.



(b) Pitch excitation

Fig. 6. ADAM-0: closed-loop experimental dataset used by MIMO data-driven method.

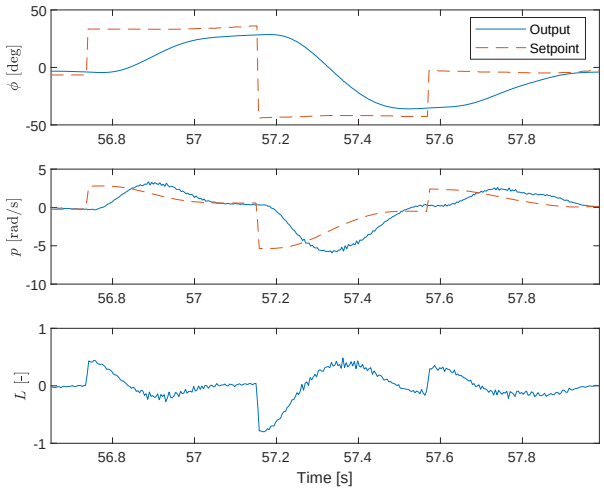
Table 4. ADAM-0: optimal controller parameters for outer and inner controllers considering the VRFT method with closed-loop experimental data.

	$\mathbf{K_P^O}$	$\mathbf{K_P^I}$	$\mathbf{K_I^I}$	$\mathbf{K_D^I}$
MIMO	$\begin{bmatrix} 4.4484 & -0.1287 \\ 0.2364 & 5.2220 \end{bmatrix}$	$\begin{bmatrix} 0.1187 & -0.0004 \\ 0.0044 & 0.1252 \end{bmatrix}$	$\begin{bmatrix} 0.1749 & 0.0023 \\ 0.0090 & 0.1164 \end{bmatrix}$	$\begin{bmatrix} 0.0007 & 0.0001 \\ 0 & 0.0011 \end{bmatrix}$
SISO	$\begin{bmatrix} 4.2505 & 0 \\ 0 & 4.2061 \end{bmatrix}$	$\begin{bmatrix} 0.1381 & 0 \\ 0 & 0.1495 \end{bmatrix}$	$\begin{bmatrix} 0.1039 & 0 \\ 0 & 0.3039 \end{bmatrix}$	$\begin{bmatrix} 0.0015 & 0 \\ 0 & 0.0027 \end{bmatrix}$

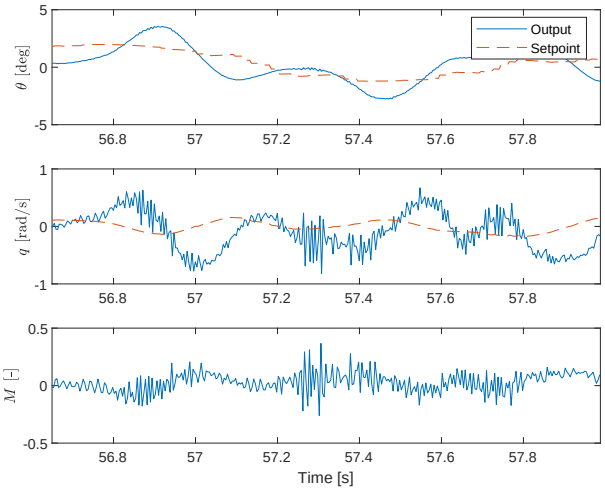
Note that the regulator obtained from the MIMO algorithm leads to a quicker response.

The MIMO controller features a quick suppression of the oscillations in the pitch rate loop, which is able to follow the rate setpoint given from the attitude feedback loop. Finally, the control effort on the pitch axis is reduced with respect to the SISO controller.

Since the data-collecting experiments are conducted in near-hovering conditions, the secondary diagonal of the parameters in Table 4 is always one or more orders of magnitude smaller than the primary terms, confirming an almost decoupled dynamics between the DoFs in the quadrotor platform. The symmetry of the build is more evident when the SISO results are analysed, since pitch and roll results are very similar.

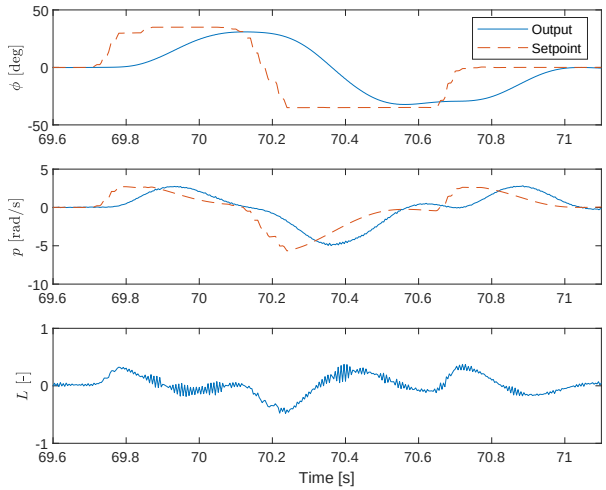


(a) Roll response

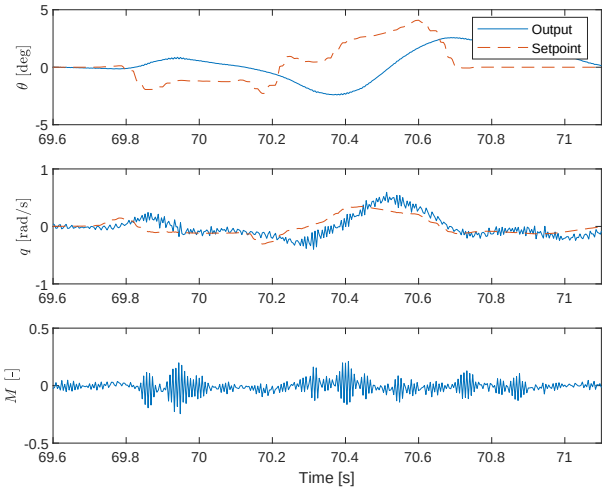


(b) Pitch response

Fig. 7. ADAM-0: Roll doublet experiment with SISO method parameters.



(a) Roll response



(b) Pitch response

Fig. 8. ADAM-0: Roll doublet experiment with MIMO method parameters.

4 Conclusions

The problem of data-driven design of the attitude control law for a multirotor UAV has been considered. The VRFT method has been extended to consider a more general class of controllers and by allowing the closed-loop execution of data-collection experiments on the system. Experimental results show that the in-flight tests can be conducted in a safe way and that a satisfactory level of performance can be achieved by using a 20 seconds data sets.

It is highlighted that the MIMO formulation of the problem allows to reduce the effects of coupling that can arise for very aggressive manoeuvres, such as the ones featured in the experiments, even for the case of seemingly symmetrical builds. These couplings typically arise from a number of dynamic and aerodynamic effects which are difficult to model, thus all situations leading to a nonlinear behaviour. These effects make this class of synthesis methods very appealing, since almost no assumptions on the system are made.

As with other data-driven methods, no stability constraint is enforced on the algorithm, making the solution of the method reliant on the choice of a suitable reference model. The main advantages over the SISO formulation are that the instrumental variable parameters (model order and past/future windowing, see[10]) are not needed, reducing the number of tuning variables. Furthermore, for one of the possible choice of instrumental variable for the SISO algorithm, the initial controller must be known, while in the MIMO formulation it is no longer necessary. Finally, this method allows a more general approach to the problem, removing the hypothesis of symmetrical builds and decoupled dynamics.

References

[1] R. Mahony, V. Kumar, and P. Corke. Multirotor Aerial Vehicles: Modeling, Estimation and Control of Quadrotor. *IEEE Robotics & Automation Magazine*, 19(3):20–32, 2012.

[2] F. Riccardi, P. Panizza, and M. Lovera. Identification of the attitude dynamics for a variable-pitch quadrotor UAV. In *40th European Rotorcraft Forum, Southampton, UK*, pages 1–9, 2014.

[3] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin. Iterative feedback tuning: theory and applications. *IEEE Control Systems*, 18(4):26–41, 1998.

[4] M.C. Campi, A. Lecchini, and S.M. Savaresi. Virtual reference feedback tuning: a direct method for the design of feedback controllers. *Automatica*, 38(8):1337–1346, 2002.

[5] K. Van Heusden, A. Karimi, and D. Bonvin. Data-driven model reference control with asymptotically guaranteed stability. *International Journal of Adaptive Control and Signal Processing*, 25(4):331–351, 2011.

[6] S. Formentin, K. Heusden, and A. Karimi. A comparison of model-based and data-driven controller tuning. *International Journal of Adaptive Control and Signal Processing*, 28(10):882–897, 2014.

[7] P. Panizza, D. Invernizzi, F. Riccardi, S. Formentin, and M. Lovera. Data-driven attitude control law design for a variable-pitch quadrotor. In *American Control Conference, Boston, USA*, 2016.

[8] F. Riccardi and M. Lovera. Robust attitude control for a variable-pitch quadrotor. In *IEEE Conference on Control Applications, Antibes, France*, pages 730–735, 2014.

[9] S. Formentin, A. Cologni, D. Belloli, F. Previdi, and S.M. Savaresi. Fast tuning of cascade control systems. In *18th IFAC World Congress, Milan, Italy*, pages 10243–10248, 2011.

[10] S. Capocchiano, P. Panizza, D. Invernizzi, and M. Lovera. Closed-loop data-driven attitude control design for a multirotor UAV. In *IEEE Conference on Control Technology and Applications, Copenhagen, Denmark*, 2018.

[11] L. Campestrini, M. Gevers, and A.S. Bazanella. Virtual Reference Feedback Tuning for Non Minimum Phase Plants. In *European Control Conference (ECC 2009), Budapest, Hungary, pp. 1955-1960*, 2009.

[12] S. Formentin, M. Corno, S.M. Savaresi, and L. Del Re. Direct data-driven control of linear time-delay systems. *Asian Journal of Control*, 13(5):1–12, 2011.

[13] S. Formentin and A. Karimi. Enhancing statistical performance of data-driven controller tuning via L_2-regularization. *Automatica*, 50(5):1514–1520, 2014.

[14] F. Previdi, T. Schauer, S.M. Savaresi, and K.J. Hunt. Data-driven control design for neuroprotheses: a virtual reference feedback tuning (VRFT) approach. *IEEE Transactions on Control Systems Technology*, 12(1):176–182, 2004.

[15] S. Formentin, M.C. Campi, and S.M. Savaresi. Virtual reference feedback tuning for industrial PID controllers. In *19th IFAC World Congress, Cape Town, South Africa*, pages 11275–11280, 2014.

[16] D. Invernizzi, P. Panizza, F. Riccardi, S. Formentin, and M. Lovera. Data-driven attitude control law of a variable-pitch quadrotor: a comparison study. In *20th IFAC Symposium on Automatic Control in Aerospace (ACA 2016), Sherbrooke, Quebec, Canada*, 2016.

[17] S. Formentin, S.M. Savaresi, and L. Del Re. Non-iterative direct data-driven controller tuning for multivariable systems: theory and application. *IET control theory & applications*, 6(9):1250–1257, 2012.

[18] T. Söderström and P. Stoica. Instrumental variable methods for system identification. *Circuits, Systems and Signal Processing*, 21(1):1–9, 2002.