Abstract—This paper proposes data-driven Model-Free Adaptive Control (MFAC) algorithms for a Multi Input-Multi Output (MIMO) twin rotor aerodynamic systems. The azimuth and pitch position control are carried out using a MIMO control system structure with two control loops, one for each axis of motion. A set of experimental results on laboratory equipment is given to show the ability of the MFAC algorithms to control the mechanical ensemble even without decoupling and in spite of disturbances.

I. INTRODUCTION

Multi Input-Multi Output (MIMO) twin rotor systems are nonlinear process benchmarks which illustrate several MIMO and Single Input-Single Output (SISO) control system (c.s.) structures and algorithms. Some current approaches to the azimuth and pitch position control of MIMO twin rotor systems include sliding mode control also combined with fuzzy control [1], [2], fuzzy and neuro-fuzzy control [3], [4], linear parameter varying control [5], and evolutionary-based tuning of linear controllers [6], [7].

A data-driven Model-Free Adaptive Control (MFAC) scheme has been proposed and reported in [8]–[12]. MFAC has attractive features for control applications since it only uses the online input-output data collected from the process. The c.s. structure relies on dynamic linearized models of the process upon which the control algorithm is formulated in a similar manner to predictive control. The theoretical framework is also attractive because it uses a general MIMO formulation, and it ensures, under several assumptions, the convergence and stability of the c.s. [11], [12]. In this sense, MFAC is different from other similar data-driven iterative schemes as Iterative Feedback Tuning [13], Correlation-based Tuning [14], Frequency-domain Tuning [15], [16], Iterative Regression Tuning [17] and Simultaneous Perturbation Stochastic Approximation [18], [19], non-iterative schemes as Virtual Reference Feedback Tuning [20], [21], and adaptive schemes as Model-Free Control [22], Unfalsified Control [23] and Adaptive Dynamic Programming [24]. MFAC has been implemented in chemical industry [25], linear motor control in injection molding [10], pH value control [26], robotic welding [27], industrial boilers [28], pneumatic muscle actuators [29], flash butt welding [30], magnetic levitation [31], power converters [32] and wind turbines pitch control [33].

The results of this paper extend the applicability of the MFAC algorithm to a MIMO twin rotor aerodynamic system. The paper is focused on experimental validation offering a simple and cost-effective control solution. This paper is organized as follows: Section II presents the MIMO twin rotor aerodynamic process model and formulates the reference trajectory tracking problem in a MFAC setting. The process models and the c.s. structure are shown. Two MFAC algorithms are presented in Section III. The experimental results given in Section IV show the ability of the two algorithms to control the process. The conclusions are outlined in Section V.

II. THE CONTROL PROBLEM

The nonlinear process in the MIMO twin rotor aerodynamic system is characterized by the following state-space model [34]:

\[
\begin{align*}
\dot{\omega}_t &= f_1(\omega, \phi, \omega_0) + J_h \alpha_h, \\
\dot{\omega}_m &= f_2(\omega, \phi, \omega_0) + J_h \alpha_m, \\
\dot{y}_1 &= \dot{\theta}_t + \phi, \\
\dot{y}_2 &= \dot{\theta}_m + \phi,
\end{align*}
\]

where: \(u_t\) [%] – the first control input, i.e., the PWM duty cycle of the horizontal (main) DC motor, \(u_t\) [%] – the second control input, i.e., the PWM duty cycle of the vertical (tail) DC motor, \(\alpha_h\) [rad/s] – the first process output, i.e., the azimuth (horizontal) position of the beam which supports the main and the tail rotor, \(\alpha_m\) [rad/s] – the second process output, i.e., the pitch (vertical) position of the beam, \(\Omega_h\) [rad/s] – the azimuth angular velocity of the beam, \(\Omega_m\) [rad/s] – the pitch angular velocity of the beam, \(l_t\) [m] – the length of the tail part of the beam, \(F_h\) [N] – the aerodynamic force from the tail rotor, \(\Omega_m\) [rad/s] – the angular velocity of the tail rotor, \(f_1\) [N m/s/rad] – the friction coefficient in the vertical axis, \(J_h\) [kg m^2] – the sum of moments of inertia relative to the vertical axis, \(I_m\) [m] – the length of the main part of the beam, \(F_m\) [N] – the aerodynamic force from the main rotor, \(\Omega_m\) [rad/s] – the angular velocity of the main rotor, \(f_2\) [N m/s/rad] – the friction coefficient in the horizontal axis, \(g\) [m/s^2] – the gravitational acceleration, \(J_t\) [kg m^2] – the sum of moments of inertia relative to the horizontal axis, \(k_{in}\) [N m] – the coefficient of the cross moment from main rotor to azimuth, \(I_e\) [kg m^2] – the moment of inertia of the tail rotor, \(k_{in}\) [rad/s] – the velocity gain of the tail rotor, \(\alpha_m\) [rad/s] – the moment of inertia of the main rotor, \(k_{in}\) [rad/s] – the velocity gain of the main rotor, and \(A, B, \ldots\)
C [kg m] – constant parameters. The linearization of (1) at the equilibrium point leads to the following linearized state-space model of the process

\[
\dot{\Omega}_s = a_{i_s} \alpha_s + a_{i_w} \omega_s, \quad \dot{\omega}_s = \Omega_s, \quad \dot{\alpha}_s = \Omega_s, \quad \dot{\omega}_s = \frac{(\alpha_i - \omega_s / k_{th}) / I_s}{},
\]

(2)

where all variables are actually expressed as deviations with respect to the equilibrium point. The transfer functions (t.f.s) of the linearized process model can be computed easily from (2) [35].

The multivariable and nonlinear character of the process, the un-modeled dynamics and the poor identification of the model’s parameters recommend the MFAC algorithm for control purposes.

The MIMO c.s. structure dedicated to azimuth and pitch position reference tracking MFAC control is presented in Fig. 1. Fig. 1 points out the two SISO control loops (12)

where \( \Phi(k) \) is the desired reference trajectory to be tracked. Another assumption about the PPD matrix \( \Phi(k) \) requires that it is a diagonally dominant matrix such that

\[
|\Phi_i(k)| \leq b_1, \quad b_2 \leq |\Phi_i(k)| \leq a_2 b_1, \quad i = 1, p, j = 1, p, i \neq j, \quad a_1 \geq 1, b_2 > b_1 (2a_1 + p - 1),
\]

and the sign of all the elements of \( \Phi(k) \) will remain unchanged. The structure of such a PPD matrix cannot be verified in practice but it should be enforced as an assumption.

The CFDL-MFAC algorithm is summarized as [12]:

\[
\Phi(k) = \Phi(k - 1)
\]

(7)

\[
\Phi_i(k) = \Phi_i(1), \text{ if } |\Phi_i(k)| < b_2 \text{ or } |\Phi_i(k)| > b_2 \text{ or } \text{sgn}(\Phi_i(k)) \neq \text{sgn}(\Phi_i(1)),
\]

(8)

\[
\Phi_i(k) = \Phi_i(1), \text{ if } |\Phi_i(k)| > b_2 \text{ or } \text{sgn}(\Phi_i(k)) \neq \text{sgn}(\Phi_i(1)), \quad i \neq j,
\]

(9)

where \( \eta \in (0, 1) \) is a step size constant, \( \mu > 0 \) is a weighting factor, \( \Phi_i(1) \) is the initial value of \( \Phi_i(k) \), \( i = 1, p \). The parameter \( \rho \) is regarded as a step size scaling factor and \( \lambda \) is considered as a degree of freedom in the design. The above mentioned assumptions together with the estimation mechanism (7), the resetting conditions (8) and with the recurrent control law (9) ensure the stability of the c.s. structure as stated in Theorem 3 in [12]. The Partial Form Dynamic Linearization (PFDL)-MFAC algorithm is next presented.

B. PFDL-MFAC Algorithm

In the PFDL version of the MFAC algorithm, the system (3) is also assumed to be generalized Lipschitz, that is,

\[
\|\Delta y(k + 1)\| \leq b \|\Delta u(k)\| \quad \text{for each fixed } k, \quad \|\Delta u(k)\| \neq 0,
\]

where \( \Delta y(k + 1) = y(k + 1) - y(k) \), \( \Delta u(k) = u(k) - u(k - 1) \) and \( b \) is a positive constant. For the nonlinear system (3), satisfying above requirements there must exist \( \Phi(k) \), called the PPD matrix, such that, (3) can be transformed into the following equivalent CFDL data model with the proof given in [11] and [12]:

\[
\Delta y(k + 1) = \Phi(k) \Delta u(k),
\]

(4)

where \( \Phi(k) = [\Phi_i(k)]_{1 \times p}, \quad \|\Phi(k)\| \leq h \). The objective in the original formulation of MFAC [12] is to solve the optimization problem

\[
\hat{u}(k) = \arg \min_{u(k)} J(u(k)),
\]

(5)

\[
J(u(k)) = \|y^*(k + 1) - y(k + 1)\|^2 + \lambda \|\Delta u(k)\|^2,
\]

where \( y^*(k + 1) \) is the desired reference trajectory to be tracked. Another assumption about the PPD matrix \( \Phi(k) \) requires that it is a diagonally dominant matrix such that

\[
|\Phi_i(k)| \leq b_1, \quad b_2 \leq |\Phi_i(k)| \leq a_2 b_1, \quad i = 1, p, j = 1, p, i \neq j, \quad a_1 \geq 1, b_2 > b_1 (2a_1 + p - 1),
\]

and the sign of all the elements of \( \Phi(k) \) will remain unchanged. The structure of such a PPD matrix cannot be verified in practice but it should be enforced as an assumption.

The CFDL-MFAC algorithm is summarized as [12]:

\[
\Phi(k) = \Phi(k - 1)
\]

(7)

\[
\Phi_i(k) = \Phi_i(1), \text{ if } |\Phi_i(k)| < b_2 \text{ or } |\Phi_i(k)| > b_2 \text{ or } \text{sgn}(\Phi_i(k)) \neq \text{sgn}(\Phi_i(1)),
\]

(8)

\[
\Phi_i(k) = \Phi_i(1), \text{ if } |\Phi_i(k)| > b_2 \text{ or } \text{sgn}(\Phi_i(k)) \neq \text{sgn}(\Phi_i(1)), \quad i \neq j,
\]

(9)

where \( \eta \in (0, 1) \) is a step size constant, \( \mu > 0 \) is a weighting factor, \( \Phi_i(1) \) is the initial value of \( \Phi_i(k) \), \( i = 1, p \). The parameter \( \rho \) is regarded as a step size scaling factor and \( \lambda \) is considered as a degree of freedom in the design. The above mentioned assumptions together with the estimation mechanism (7), the resetting conditions (8) and with the recurrent control law (9) ensure the stability of the c.s. structure as stated in Theorem 3 in [12]. The Partial Form Dynamic Linearization (PFDL)-MFAC algorithm is next presented.

B. PFDL-MFAC Algorithm

In the PFDL version of the MFAC algorithm, the system (3) is also assumed to be generalized Lipschitz, that is,

\[
\|\Delta y(k + 1)\| \leq b \|\Delta U(k)\| \quad \text{for each fixed } k, \quad \|\Delta U(k)\| \neq 0,
\]

where \( \Delta y(k) = [\Delta y(k)]_{p \times 1} \), \( \Delta u(k) = [\Delta u(k)]_{p \times 1} \), \( \Delta u(k - 1) = [\Delta u(k - 1)]_{p \times 1} \), the backward shift of the control input is \( i = 0, L - 1 \), \( \Delta y(k + 1) = [\Delta y(k + 1)]_{p \times 1} \), \( \Delta u(k) = 0 \) for all \( k \leq 0 \) and \( b \) is a positive constant [12].
For the nonlinear system (3), satisfying the previous requirements with \( \| \Delta \mathbf{U}(k) \| \neq 0 \) for all \( k \), a similar description to (4) is employed for the PFDL-MFAC [12]:

\[
\Delta y(k+1) = \Phi^T(k) \Delta \mathbf{U}(k),
\]

where

\[
\Phi(k) = [\Phi_1(k) \Phi_2(k) \ldots \Phi_L(k)]^T, \quad \| \Phi(k) \| \leq b,
\]

\[
\Phi_n(k) = [\Phi_\theta(k) \downarrow_{i=1}^n \mathbf{m}]_m = \overline{1:L}.
\]

In the SISO case, the PPD matrix becomes a vector \( \Phi(k) = [\Phi_\theta(k) \Phi_z(k) \ldots \Phi_L(k)]^T \). With the same objective defined in (5), the PFDL-MFAC-SISO algorithm is summarized as [12]:

\[
\dot{\Phi}(k) = \Phi(0) - (1) + \frac{\eta \Delta \mathbf{U}(k-1)}{\mu + \| \Delta \mathbf{U}(k-1) \|^2} \Delta y(k) - \Phi^T(k-1) \Delta \mathbf{U}(k-1) - \Phi(k-1) \Delta \mathbf{U}(k-1),
\]

\[
\phi_i(l) = \Phi(l), \text{if } \| \Phi(k) \| \leq \varepsilon \text{ or } \text{sgn}(\hat{\phi}_i(k)) \neq \text{sgn}(\hat{\phi}_i(l)), \quad \varepsilon \text{ is a small positive constant},
\]

\[
u(k) = u(k-1) + \rho \hat{\phi}_i(k)[y^i(k-1) - y^i(k)]
\]

\[
\hat{\phi}_i(k) = \frac{\sum_{i=1}^{L} \rho \hat{\phi}_i(k) \Delta u(k-i+1)}{\lambda + \| \hat{\phi}_i(k) \|^2},
\]

where \( \eta \in (0,1) \) is a step size constant, \( \mu > 0 \) is a weighting factor, \( \rho = [\rho_1 \ldots \rho_L]^T \) is a step size vector, \( \rho_i \in (0,1], \quad i = \overline{1:L} \), which is added to make (14) more general, \( \Phi(k) \) is the estimate of \( \Phi(k) \), \( \varepsilon \) is a small positive constant, and \( \hat{\phi}_i(l) \) is an initial value of \( \hat{\phi}_i(k) \) in the SISO case. As in the CFDL scheme, \( \lambda \) is a degree of freedom in the design. The PPD matrix estimation mechanism (12), the resetting condition (13) and the recurrent control law (14) fulfill the stability conditions formulated in Theorem 2 in [12].

According to [11] and [12], the recommendations on choosing \( L \) suggest that it should be connected to the approximated orders of the unknown process or simply proportional to the complexity of the process.

IV. EXPERIMENTAL RESULTS

The CFDL-MFAC and PFDL-MFAC algorithms are applied to the nonlinear twin rotor aerodynamic system laboratory equipment. Three scenarios are used. In the first SISO scenario, the pitch is fixed and the azimuth is controlled. In the second SISO scenario, the azimuth is fixed and the pitch is controlled. In the third MIMO scenario, both pitch and azimuth are controlled.

The main results related to the application of these two MFAC algorithms are presented as follows. The sampling period is \( T_s = 0.01 \) s for all the experiments. The number of samples is \( N = 9000 \), the desired reference trajectories used in all scenarios are:

\[
y^1(k) = 0.2 \text{ if } k \in [0,4500], \quad 0.2 \text{ if } k \in (4500,7500), \quad 0 \text{ if } k \in (7500,9000), \quad y^2(k) = 0.2 \text{ if } k \in [0,3000], \quad 0.2 \text{ if } k \in (3000,6000), \quad 0 \text{ if } k \in (6000,9000).
\]

As performance index for the next experiments we use \( J^p = \sum_{k=1}^{N} \varepsilon^1_k(k) \) and \( J^p = \sum_{k=1}^{N} \varepsilon^2_k(k) \) in the SISO scenarios where \( J^p \) corresponds to the pitch motion. \( J^p = \sum_{k=1}^{N} (\varepsilon^1_k(k) + \varepsilon^2_k(k)) \) measures the performance in the MIMO scenario, where the control errors \( \varepsilon^1_k(k) \) and \( \varepsilon^2_k(k) \) from Fig. 1 are not related to \( \varepsilon \) from (13). Different criteria can be used in combination with other applications [36]–[42].

For the PFDL-MFAC algorithm in the SISO and MIMO scenarios, the value \( L=3 \) is chosen for computing the recurrent control law from (14).

For the c.s. with CFDL-MFAC algorithm in the SISO scenarios, a 0.1 step output additive disturbance is applied to \( y_1 \) at 35 s, and a –0.1 step output additive disturbance is applied to \( y_2 \) at 50 s. For the c.s. with PFDL algorithm in the SISO scenarios, a 0.1 step output additive disturbance is applied to \( y_1 \) at 35 s, and a 0.1 step output additive disturbance is applied to \( y_2 \) at 50 s.

Since there are many parameters of both CFDL-MFAC and PFDL-MFAC algorithms and their experimental heuristic tuning is costly, we start with an optimization based approach using a Gravitational Search Algorithm [43] in the pre-tuning phase. Other optimization algorithms can be used as well [44]–[48].

First, the performance indices \( J^p \), \( J^p \) and \( J^p \) are minimized for the best combination of MFAC algorithm parameters. These parameters are then manually tuned on the experimental equipment. The values of the parameters of CFDL-MFAC and PFDL-MFAC algorithms corresponding to the first two SISO scenarios are shown in Table I.

| TABLE I. VALUES OF PARAMETERS OF CFDL-MFAC AND PFDL-MFAC ALGORITHMS IN THE FIRST TWO SISO SCENARIOS FOR EACH MOTION |
|-----------|-----------|-----------|-----------|-----------|
|           | Azimuth   | Pitch     | Azimuth   | Pitch     |
| CFDL-MFAC | 2501      | 2001      | 1800      | 261       |
| PFDL-MFAC |           |           | 0         | 0         |
|           |           |           | 0         | 0         |
| \eta, \lambda, \mu | \eta = 0.1, \lambda = 5, \mu = 0.99 |
| \rho | 0.79      | 5.53      | 0.79      | 0.79      |
| \gamma | 2500      | 2000      | 10^7      | 10^7      |
| \Gamma | 2501.5    | 2001.9    | 644.79    | 106.2     |
| \tau | 638.2     | 93.24     | 638.2     | 93.24     |

In the CFDL-MFAC algorithm, the parameter \( \gamma \) is the lower limit and it is \( b_1 \) from (6), and the parameter \( \Gamma \) is the upper limit and it is \( b_2 \) from (6). In the PFDL-MFAC
algorithm, $\gamma$ is $\varepsilon$, and an upper limit $\Gamma$ of $|\hat{\phi}(k)|$ is not needed in the resetting mechanism.

Fig. 2 shows the experimental results as the controlled output and the control input for the c.s. with CFDL-MFAC algorithm applied in the SISO azimuth position control. Fig. 3 shows the experimental results as the controlled output and the control input for the c.s. with CFDL-MFAC algorithm applied in the SISO pitch position control.

Fig. 4 presents the experimental results for the c.s. with CFDL-MFAC algorithm in the MIMO scenario. Fig. 5 presents the experimental results for the c.s. with PFDL-MFAC algorithm in the MIMO scenario.

Table II. Values of Parameters of CFDL-MFAC and PFDL-MFAC Algorithms in the MIMO Scenario

<table>
<thead>
<tr>
<th>CFDL-MFAC</th>
<th>PFDL-MFAC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Azimuth</td>
<td>Pitch</td>
</tr>
<tr>
<td>$\Phi(1)$</td>
<td>1001</td>
</tr>
<tr>
<td>$\eta$, $\lambda$, $\mu$</td>
<td>$\eta = 0.1$, $\lambda = 5$, $\mu = 0.99$</td>
</tr>
<tr>
<td>$\rho$</td>
<td>0.79</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>1000</td>
</tr>
<tr>
<td>$\Gamma$</td>
<td>1001.5</td>
</tr>
<tr>
<td>$J_{p}^{\ast}$</td>
<td>407.72</td>
</tr>
</tbody>
</table>

In Fig. 6 the results show the output and the control input for the c.s. with CFDL-MFAC algorithm in the MIMO azimuth position control. Fig. 7 presents the output and the control input for the c.s. with CFDL-MFAC algorithm in the MIMO pitch position control.

In Fig. 8 the results show the output and the control input for the c.s. with PFDL-MFAC algorithm in the MIMO azimuth position control. Fig. 9 presents the output and the control input for the c.s. with PFDL-MFAC algorithm in the MIMO pitch position control.
V. CONCLUSION

This paper has suggested an application of CFDL-MFAC and PFDL-MFAC algorithms to MIMO and SISO systems for the azimuth and pitch control in twin rotor systems. The experimental results showed that there are no significant differences between the results of CFDL and PFDL-MFAC algorithms in the SISO scenarios. Some better results were obtained for the PFDL-MFAC algorithm in the SISO scenario and for the CFDL-MFAC algorithm in the MIMO scenario. The dependence on the PPD matrix order is not relevant in our tests.

The azimuth control showed increased sensitivity for disturbances and poor c.s. performance even from the pre-tuning phase and after the experimental phase. The reasons have to be further investigated. In general, the high number of parameters proves to be difficult to tune systematically but this has to be compared against the lack of knowledge on the process model.

Future research will be dedicated to the design of c.s.s with MFAC algorithms and reduced parametric sensitivity. The algorithms must cope with the nonlinearities of many processes [49]–[56].

ACKNOWLEDGMENT

This work was supported by a grant of the Romanian National Authority for Scientific Research, CNCS - UEFISCDI, project number PN-II-ID-PCE-2011-3-0109.

REFERENCES


