

Louis Gagnon provides this scientific self-archived article free of charge.
For more research info see LouisGagnon.com

This article is the final submission, post-review, version of the following article:

Gagnon, L., Masarati, P., « Autonomous Untethered Flight of Multibody Dynamics Rotorcraft With Cycloidal Rotors », DETC2018-85152, ASME IDETC/CIE 2018, Québec City, Canada, August 26-29, 2018

Copyright © 2018 by the American Society of Mechanical Engineers (ASME)

Note: The version of this document may differ in format from the official version distributed by the publisher. The scientific content should nevertheless be identical as this version was created after conclusion of the peer-review process. For the official version, please consult the publishers website. Do keep in mind that a subscription or fee may be asked for the official version.

DETC2018-85152

AUTONOMOUS UNTETHERED FLIGHT OF MULTIBODY DYNAMICS ROTORCRAFT WITH CYCLOIDAL ROTORS

Louis Gagnon¹

Pierangelo Masarati²

Department of Aerospace Science and Technology
Politecnico di Milano
20156 Milano, Italy

¹Email: louis.gagnon@polimi.it

²Email: pierangelo.masarati@polimi.it

ABSTRACT

Multibody dynamics models of a helicopter and two cycloidal rotor aircraft concepts capable of vertical take-off and landing (VTOL) are constructed. The first concept aircraft is a helicopter equipped with two lateral cycloidal rotors acting as a replacement for its tail rotor and is named the Heligyro. The other concept is named the Quadricyclogyro and is propelled exclusively by four cycloidal rotors whose axes are aligned. The autopilot algorithm is implemented as a proportional, integral, and derivative (PID) controller and is tuned using a genetic optimization algorithm directly on the multibody models. Aircraft vibration and energy requirements are monitored and fed as penalty functions to the genetic algorithm. The time-domain responses of the aircraft attempting to follow mission paths of variable complexity obtained from the literature are studied. Overall, the tuned VTOL aircraft are able to reproduce the requested routes with good accuracy if a certain speed threshold is respected.

INTRODUCTION

There are different reasons why semi or fully automated pilot systems for aircraft simulations are developed. Some attempt to avoid collisions, others provide assistance for night and bad weather or nap-of-the-earth flight [1, 2]. Some autopilot algorithms are developed to study particular trim conditions [3, 4],

to provide fully automated flight [5], or to study innovative aircraft [6].

The purpose of the autopilot model developed here fits in the last category. It provides a repeatable method to preliminarily evaluate the flight comfort, handling characteristics, and energy consumption of aircraft concepts that do not currently materially exist. This autopilot is thus an indispensable tool when evaluating the feasibility of an aircraft concept at stage of early design and low technology readiness level. It is developed with the purpose of allowing comparison between a traditional helicopter and novel cycloidal rotor concepts while maintaining the bias induced by a pilot's ability to control a novel aircraft to a minimum.

In their most general expression, cycloidal rotors are sets of constant section and null twist blades which are equidistantly distributed to form a cylindrical shape. They rotate around the central axis of this cylinder and provide thrust by cyclically pitching around their own pivot axis. A schematic example is shown in Fig. 1.

Aside from their commercial use as boat propellers [7], they are used mostly for research purposes as the main thrust source for small aircraft [8–11] and airships [12]. They receive a lot of attention because of their ability to instantly change the direction of their thrust in a 360° plane perpendicular to the cylinder's axis. Recent research by the authors identified a strong potential for better handling and lower energy consumption of aircraft

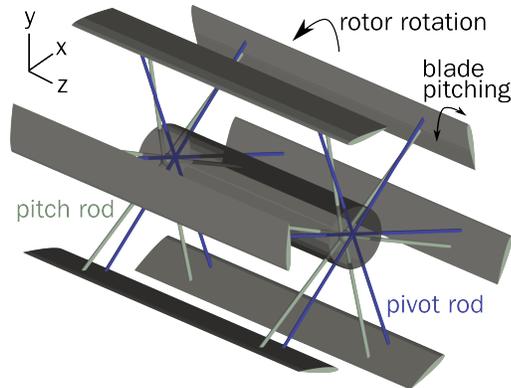


FIGURE 1. CYCLOIDAL ROTOR.



FIGURE 2. HELIGYRO CONCEPT.

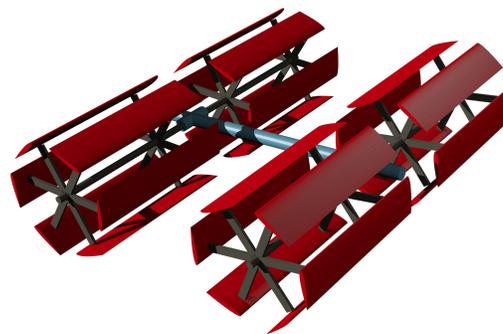


FIGURE 3. QUADRICYCLOGYRO CONCEPT.

through the use of cycloidal rotors using analytical rigid [13] and flexible [14] solutions, fixed [15] and free (untethered) [16] multibody dynamics models, and bi- and tridimensional computational fluid dynamics [17, 18]. This paper thus attempts to further investigate two promising aircraft models which rely on cycloidal rotors to provide better control and potentially reduce energy consumption. This research builds upon the preliminary models described in a previous conference paper [16]. The two concepts beyond the helicopter model studied in this paper are shown in Fig. 2 and Fig. 3.

The motivation to rely on cycloidal rotors in this study comes from their near-total absence from industrial research. Although different attempts were made by the industry to go beyond the prototype stage, the challenging control system design prevented their further development. However, the lack of a good directional thrust capability for traditional rotors and the ability of cycloidal rotors to immediately vector the thrust in a circular path, make them ideal for scenarios where agility is required. Even though quadrotor aircraft with traditional rotors are easier to control than helicopters because of the absence of gyroscopic effects and their multiple independent rotors, they are still unable to uncouple asset and position controls. A cycloidal rotor aircraft can do so and maintain a pitch or roll condition regardless of the path it has to follow. There is thus a large potential for these rotors to be demonstrated and expanded through pure research.

The innovative aspects of this paper are to automatically fly totally free helicopter and cyclogyro comprehensive multibody dynamics models. Also, the autopilot uses a control system that is calibrated within a genetic optimization procedure and is able to fly successfully in many different flight paths. Finally, the cycloidal rotor concepts are tested for their agility and power usage while following challenging flight paths. This article briefly presents the multibody models, then describes the control system, then discusses the optimization method, and finally presents the performance of the aircraft tested on different flight paths.

MULTIBODY MODELS

The performance of each new aircraft configuration is evaluated using time-marching multibody dynamics simulations which rely on the MBDyn software [19, 20]. All multibody dynamics models developed for this paper use rigid nodes. This choice comes from the need to evaluate a large number of flight scenarios in little time within a controller optimization process. For every aircraft, the evaluation timestep is chosen as to rotate the fastest rotor by 2° . This is a stricter requirement than previous research [3] and is chosen as an affordable means to reduce controller response time and increase accuracy. The tolerance on the convergence and the tolerance on the initial derivatives of the simulation are constant for the helicopter case and dependent on the cycloidal rotor's geometry and angular velocities for the other aircraft. The dependence is calculated according to the contributions to the residual coming from both aerodynamic and inertia forces. At the onset of the simulation, when the aircraft multibody models are launched, the rotors are gradually brought up to speed and the viscoelastic connexion which holds the aircraft in place is gradually removed.

Cycloidal rotors

These particular rotors are present in the models of the Quadricyclogyro and the Heligyro and are modeled using the same node arrangements. For simplicity of the analysis and be-

cause prior research considers it a fairly safe choice, the rotors each have 6 blades. The mass and inertia of the aircraft are distributed between the airframe node and each blade node. The masses are proportional to the area of the blades and the inertias to both mass and span of the blades. The blades have constant chords, no twist, and have aerodynamic forces dependent on the operating condition and calculated from a lookup table. This cycloidal rotor simulation method was previously validated against 3 different experimental sources [13].

Quadricyclogyro

The Quadricyclogyro multibody model is a simple arrangement of four cycloidal rotors rigidly attached together at a central node. The rotors turn a constant angular velocity and a function with variable magnitude and phase imposes the local blade pitch angles.

Helicopter and Heligyro

A previously validated aeroelastic multibody model [3] of the Bo105 helicopter serves as the basis for the helicopter and Heligyro models of this research. The difference is that, here, the main rotor blades are rigid elements. The blades of the main rotor are modeled as aerodynamic surface elements with NACA23012 airfoil profiles and blade twist along the span. The drag, lift, and moment forces on the blade at Mach numbers up to 0.8 are taken from experimentally obtained coefficients for angles of attack in the $[-20,20]^\circ$ range and estimated for magnitudes up to 180° . The fuselage and stabilizers are modeled as generic aerodynamic 3D forces and moments. They are obtained from coefficients and the dynamic pressure. The coefficients are obtained from bidimensional lookup tables providing data for the expected operation sideslip and incidence angle ranges. All aerodynamic data comes from [21]. The tail rotor is modeled as an applied anti-torque force, partly because the time step required to model the tail rotor is smaller than the one for the main rotor [3]. The power consumed by the tail rotor is taken to be 10% of the main rotor's power. The details of the multibody model connexions are shown in Fig. 4.

CONTROLLER

The control system used to fly the 3 aircraft is embedded within the multibody simulation, thus creating a monolithic solver. The basis of the controller is the same for each model and uses a non-predictive PID algorithm which has complete control over the active parts of the aircraft. It takes its reading from the frequency-filtered position and angles of the aircraft at the previous timestep. The filter used is a Butterworth of 2nd order. The purpose of filtering the signals is to reduce the sensitivity of the controller to the simulation timestep and avoid an excessively

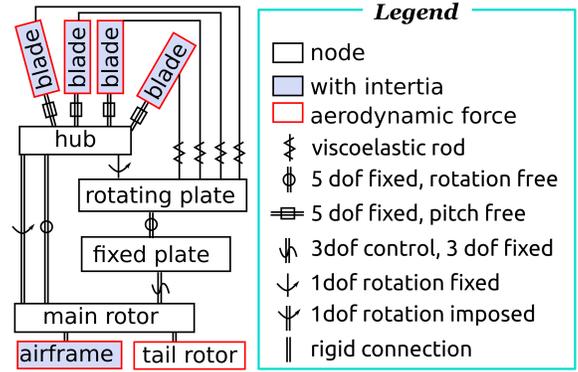


FIGURE 4. Helicopter nodes.

aggressive response. Being in a simulation environment, the relative positions and their derivatives, which are taken directly as velocities, are measured with the *model* namespace functions of the MBDyn software. Although not planned at this level of research, for a flying prototype, the dead reckoning method could be used to compute position data from the aircraft's inertial measurement unit.

Thus, the controller implements the following equation to compute the correction to be applied for each degree of freedom (DOF),

$$U_j(t) = S_j T_j + C_{j,n} \quad (1)$$

where S_j is a sign associated with the j DOF, $C_{j,n}$ is the neutral value imposed by the controller for the j DOF, and

$$T_j = C_{j,p} \epsilon_j + C_{j,d} \dot{x}_j + C_{j,i} \int \epsilon_j dt, \epsilon_j \in [-C_{j,m}, C_{j,m}] \quad (2)$$

where $C_{j,p}$, $C_{j,d}$, and $C_{j,i}$ are the proportional, derivative, and integral coefficients of the PID controller for the j DOF. The neutral values are those assumed by the rotorcraft aerodynamic surfaces when no control response is requested. They correspond to null relative pitch angles for all the Quadricyclogyro blades. For the Helicopter and Heligyro, they correspond to the default Bo105 control positions at rest, with a slight increase of the anti-torque control on the Helicopter and of the collective control for both the Helicopter and Heligyro. At the same DOF, $C_{j,m}$ is the maximum amplitude the controller can impose to the commands. Also,

$$\epsilon_j = x_j + \eta_j - \zeta_j \quad (3)$$

where x_j is the filtered, previous timestep, value of the current DOF and ζ_j is the value it is asked to take, as dictated by the given flight path. In other words, ε_j is an error function where x_j is the value of the DOF and ζ_j is the desired value of that DOF.

The variable η_j is only used as a corrector to impose a change on a DOF when needed to reduce the error of another DOF. Thus, if the current degree of freedom j is roll, and because increasing roll is the simplest method to produce a lateral translation, then

$$\eta_j = C_{j,p}T_z \quad (4)$$

where the subscript z is the lateral displacement DOF and T_z is evaluated at the current timestep. If j is pitch and the aircraft is a helicopter, then

$$\eta_j = C_{j,p}T_x \quad (5)$$

where the subscript x is the longitudinal displacement DOF and T_x evaluated at the current timestep. A similar pitch correction is also applied, to a lesser extent, to the Heligyro. For any other case, $\eta_j = 0$.

For the helicopter and Heligyro, a simple Stability Augmentation System (SAS) is applied to the pitch and roll control values. The purpose of this SAS is counter the gyroscopic effect of the main rotor. Thus,

$$U'_j(t) = (C_{j,s} + v_a C_{j,v})U_{j'}(t) + U_j(t) \quad (6)$$

with the constraint that

$$U'_j(t) \in [C_{j,n} - C_{j,m}, C_{j,n} + C_{j,m}] \quad (7)$$

where $C_{j,s}$ and $C_{j,v}$ are the SAS coefficients for the *current* DOF, v_a is the airspeed, $U_{j'}(t)$ is the calculated controller input from the *reference* DOF, $U_j(t)$ is the calculated controller input from the *current* DOF. The *current* (j) and *reference* (j') DOFs are either roll and pitch, or pitch and roll, depending on whether the SAS is correcting the roll or pitch, respectively. The calculated controller inputs come from the PID calculations done within the multibody model prior to applying the SAS corrections. Finally, these control values modified by the SAS equation are applied in place of the original controller commands.

Thus, the controller produces six corrective signals, either $U_j(t)$ or $U'_j(t)$, depending on the DOF and the aircraft concept. These signals are redirected by each of the three conceptual aircraft according to the rules of Tab. 1. For space considerations,

the entries of the table are abbreviated and the boldface characters in the following text intend to refer directly to the abbreviations used therein. The table fields indicate either that the controller signal is used to **add** to another DOF's signal, using the variable η_j of Eqn. 3; move the helicopter's or Heligyro's swash-plate's **collective**, **lateral**, or **fore-aft** actuator according to the value of Eqn. 1; **synchronously** change the **phases** or **magnitudes** of the pitch angle function of the blades of every cycloidal rotor together; or **desynchronize** the **phases** or **magnitudes** of the blades between the **Left** and **Right** or **Front** and **Rear** rotors. A more thorough description and an explanatory sketch of the controller logic applied from Eqn. 1 to the Quadricyclogyro is given in a previous article [16].

TABLE 1. REDIRECTION OF CONTROLLER OUTPUTS.

DOF	Helicopter	Heligyro	Quadricyclogyro
long.	add pitch	combined	sync. phases chg.
vert.	coll. act.	coll. act.	sync. mag. chg.
lat.	add roll	add roll	add roll
roll	lateral act.	lateral act.	desync. L&R mag.
yaw	tail rotor	cyclorotors	desync. L&R phases
pitch	fore-aft act.	fore-aft act.	desync. F&R mag.

The Heligyro controller has the particularity of responding by the **combined** action of tilting the aircraft and increasing the cycloidal rotor response when asked to provide longitudinal displacement. It applies the following equation as the local pitch angle function of the blades of its **cyclorotors**,

$$\theta(t) = \pm(U_5(t) \pm U_1(t)/2) \sin(\theta_a \pm \pi/2) \quad (8)$$

where $U_5(t)$ and $U_1(t)$ are the yaw and longitudinal obtained controller outputs, respectively, and θ_a is the relative angle between the blade and the rotor drum's initial's x-axis position. The \pm signs are adapted according to whether the cycloidal rotor is on the right or on the left side on the Heligyro.

For the helicopter and the Heligyro, the commands are damped before being applied to any aircraft part. This is done, as for filtering, to reduce the volatility of the controller response and to vaguely emulate the delays induced in a real aircraft by the hydraulic control systems. To do so, a damped joystick is emulated by applying the controller-calculated commands as input forces to a massless node within the multibody simulation. Its linear DOFs displacements are then fed as the actual control commands

and as detailed in Table 1. The rotations of these massless nodes are constrained to zero while their translations are restrained by linear viscoelastic laws having 1 N/m elasticity and 0.01 Ns/m damping. These nodes have no physical connexion to the aircraft. They only serve to yield relative displacements which are used as inputs to the collective, fore-aft, and lateral swashplate controls of the helicopter and Heligyro. A similar damping is also applied to the yaw cycloidal rotor command of the Heligyro. The increase in simulation time which results from adding this damping is negligible and the pre-optimized aircraft behavior becomes considerably smoother with the damping.

OPTIMIZATION

A minimization procedure is run to minimize the error on the prescribed paths, the energy consumed by the aircraft, and the cabin vibrations. This procedure is applied independently to each of the 3 aircraft under study.

Although each multibody model is solved on a single processor, the optimization procedure is run on an high-performance computer cluster using the Many-Task Computing (MTC) approach. This approach runs the genetic optimization algorithm and simultaneously calls multiple instances of the multibody models to refine. The multibody model used during the optimization process is solved by the naïve [22] sparse linear system solver, which is known to be efficient for small multibody dynamics problems. The colamd column rearrangement procedure is also done on the residual matrix to reduce sparsity.

The optimization problem is formulated as follows,

$$\begin{aligned} \min_{\mathbf{y}} F(\mathbf{y}) \\ \text{s.t. } a_i \leq y_i \leq b_i \end{aligned} \quad (9)$$

where $\mathbf{y} = \{y_1, y_2, \dots, y_i, \dots, y_n\}$ is a 28 or 32-variable vector of parameters defined in the following section. For each y_i parameter, a_i and b_i are its lower and upper bounds, respectively. The problem is solved by single-objective method but is formulated internally as a multicriteria optimization by letting the objective function be,

$$F(\mathbf{y}) = f_{ed}(\mathbf{y}) + f_d(\mathbf{y}) + f_E(\mathbf{y}) + f_{vib}(\mathbf{y}) + f_E(\mathbf{y}) + f_{pre}(\mathbf{y}) \quad (10)$$

where $f_{ed}(\mathbf{y})$ and $f_d(\mathbf{y})$ are the penalties applied in case of very early divergence or any divergence, respectively; $f_E(\mathbf{y})$ is the total path error during the untethered portion of the flight; $f_{vib}(\mathbf{y})$ is a function which increases with cabin vibrations; $f_E(\mathbf{y})$ increases nonlinearly with energy consumption; and $f_{pre}(\mathbf{y})$ is a possible pre-simulation interruption and penalty which combines a series analytic checks performed prior to running the models to avoid

running a multibody simulation when the parameters are clearly not a feasible solution. Each function is briefly described in the second next section.

Solution search

The process is implemented inside a constrained minimization which uses a constrained Single-objective Genetic Algorithm (SOGA) of the JEGA library [23]. A series of flight paths, starting from an attached aircraft, fully released in less than one second, and then flown by the autopilot, is fed to the SOGA. For each aircraft, 24 controller parameters are explored by the SOGA. They are the signal input filter cutoff frequency for each DOF and the $C_{j,p}$, $C_{j,i}$, and $C_{j,d}$, which consist of the proportional (P), integral (I), and derivative (D) coefficients of each of the 6 j degrees of freedom of the aircraft. For the helicopter and the Heligyro, the SAS controller adds $C_{j,s}$ and $C_{j,v}$ as 4 additional variables to the SOGA for the two concerned DOFs, as explained by Eqn. 6. Finally, the Heligyro and Quadricyclogyro also allow the minimization algorithm to vary the radius and angular velocity of the rotors and the chord and span of their blades, thus adding 4 optimization parameters. This leaves the Heligyro with 32 exploration variables and the helicopter and Quadricyclogyro with 28 such variables. The population sizes of the SOGA are chosen as to given each iteration the possibility to evaluate two values of one parameter against every possible two values of the other parameters. Thus, the population sizes are n_v^2 where n_v is the number of parameters and are thus 1024 the Heligyro and 784 for the other aircraft.

Size limits were also applied to the Heligyro so that it would fit within the geometry of the Bo105 helicopter. The angular velocity of the cycloidal rotors is limited to 500 rad/s in order to maintain a reasonable solution time of the multibody dynamics simulation.

Many different paths were attempted as a basis for the optimization, but a restrained selection was retained. The choice was made based on the capability of the different aircraft to attain the requested speeds and accelerations. Thus, the created paths were always verified to respect the maximum allowed velocities of the Bo105 in the 3 linear directions as taken from the helicopter's technical specifications. The *generated* path consists of a combination of lateral, longitudinal, and vertical oscillation movement which was deemed as a good replacement for more generic combinations displacement along one coordinate axis.

Objective functions

The objective functions within $F(\mathbf{y})$ depend on a series of parameters and are slightly different between one aircraft and another. They all have in common a single high value penalty $f_{ed}(\mathbf{y})$ for any simulation that diverges before completing at least enough rotor rotations to allow calculation of the error on the paths. Also, whenever the simulation diverges before reaching

the end of the requested path, a penalty $f_d(\mathbf{y})$ proportional to the ratio of time remaining to complete the trajectory is added to the computed residual. The residual is otherwise calculated as a combination of the distance between the requested path and the aircraft over time, the presence of vibrations in the response, and the energy consumption using the functions $f_e(\mathbf{y})$, $f_{vib}(\mathbf{y})$, and $f_E(\mathbf{y})$, respectively. For the Heligyro, an analytic solution based on prior research [13] is used before starting the multibody simulation to assess whether the weight of the added cycloidal rotors, the power they request, the thrust they generate, their number of blades by chord-to-radius ratio, and their tip Mach number at maximum helicopter advance ratio are feasible designs. In case they are not, the simulation is not run and a penalty function $f_{pre}(\mathbf{y})$ based on the importance of the error is returned. The blockage of the blades over the full cylinder is allowed to reach as much as 0.5 as such a ratio was seen to work well in a prototype [24].

Optimization paths

The Quadricyclogyro is optimized using two paths which are hover and the *generated* trajectory for a 5-minute period. The Helicopter has the same optimization path with an additional mirror version of the *generated* path in order to ensure the ability to fly backwards. The Heligyro was more unstable and the optimization was limited to one 125 second path, which is the *trickier* path published by Aldawoodi [5].

The optimized helicopter and Quadricyclogyro controllers are able to achieve a considerable precision at following the paths given during the optimization. Figs. 5 and 6 show this ability of the helicopter for the *generated* path and its mirror, respectively.

Figures 7 and 8 show the response in both position and inclination, respectively, to the *generated* path trajectory. The lines of the path to follow in position are almost entirely covered by the quadricyclogyro’s actually followed path. Thus, the match is excellent.

Optimization peculiarities

The choice for a genetic algorithm and the multi-task computing (MTC) approach comes from the difficulty encountered when dealing with the particular non-linearity of the problem and the presence of important coupling between the DOFs of the aircraft. The benefits of overcoming the difficulty of dealing with coupling issues by running an optimization process is further highlighted by the fact that the helicopter is the aircraft that most benefits from the optimization procedure while the Quadricyclogyro benefits less. With the availability of computer cluster time, quickly conducting the optimization is possible. In general, the optimizations conducted used 108 processors per path evaluated over a period of 8 to 24 hours. The choice for a genetic algorithm seemed the most natural as it allows for large populations to be evaluated at once using multiple processors, does not

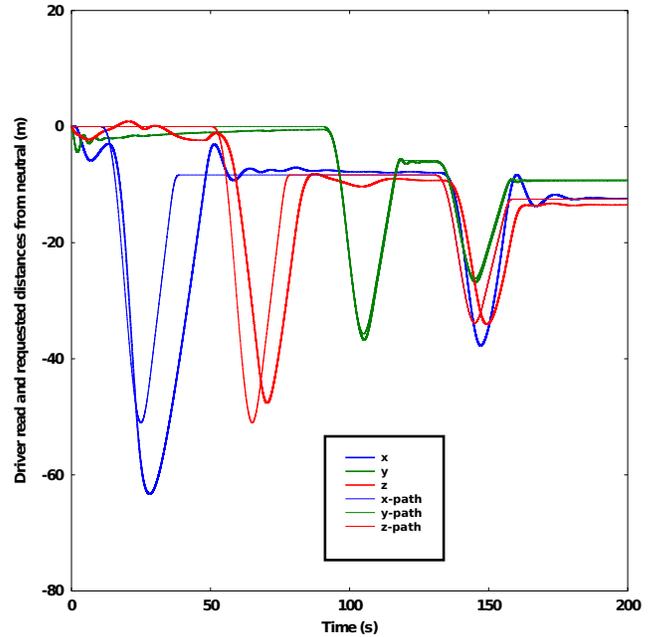


FIGURE 5. HELICOPTER GENERATED PATH AND RESPONSE.

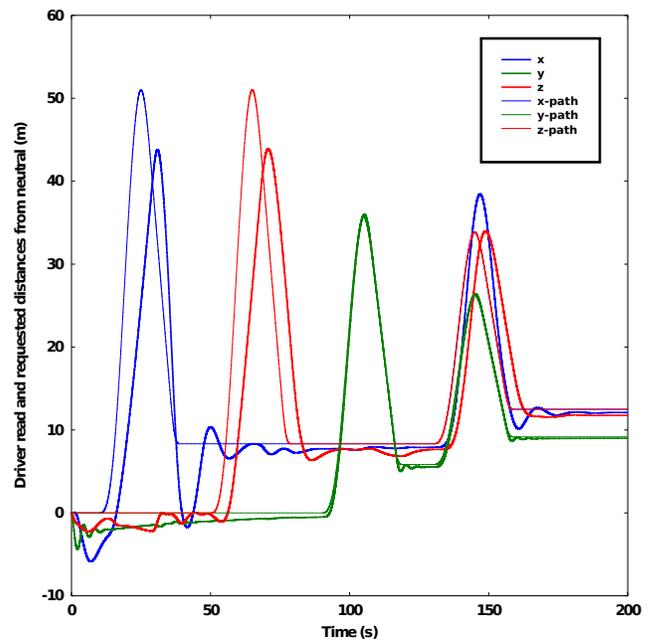


FIGURE 6. HELICOPTER MIRROR PATH AND RESPONSE.

require any information about the gradient of the objective function $F(\mathbf{y})$, and the parameters \mathbf{y} can be bounded. One limitation of the MTC is that a fraction of the allocated processors remains unused because the SOGA algorithm requires that a population evaluation be complete before conducting a subsequent popula-

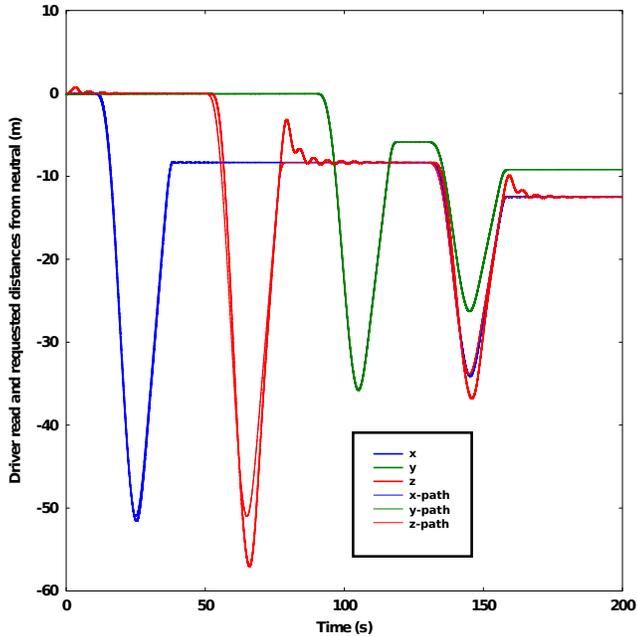


FIGURE 7. HELIGYRO GENERATED PATH AND RESPONSE.

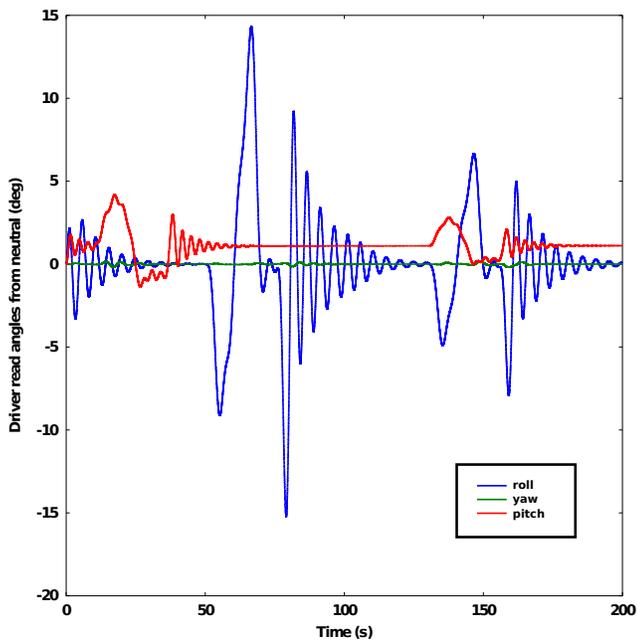


FIGURE 8. HELIGYRO GENERATED PATH ANGLES.

tion evaluation. In the extreme case, a series of evaluation could finish early and leave a single trajectory evaluation in execution on a single processor while 108 of them are actually allocated. These situations are, however, unlikely as the single multibody evaluation of a trajectory does not require a large amount of time

to complete.

Amongst the attempts that were made with the Heligyro, were optimizations with the *generated* trajectory. That approach was, however, abandoned because, although some vague reproduction of the required path was seen, the correlation was poor and the residual value would cease improving for the last 10,000 function evaluations, leaving little hope for improvement with the current controller model. Another attempt at improving the Heligyro response was to embed an inner optimization for the cycloidal rotors blades' geometry. It considered the geometry of the rotor and operating velocity to estimate the minimum possible weight that plastic printed blades reinforced with aluminum spars could take. This approach was, however, inconclusive because the plastic blade design was never strong enough to sustain the required thrust forces and the inner optimization was thus removed.

The optimized Quadricyclogyro rotors turn at 267 rad/s and the full aircraft simulation runs at 54.5 times the realtime using a single processor. The optimized Heligyro cycloidal rotors turn at 49 rad/s and the full aircraft solution takes 6.7 times the realtime using a single processor. The optimized helicopter, which has a 44.4 rad/s main rotor velocity, takes 3.8 times the realtime to solve using a single processor. These times are calculated while using a single core of a 12-core computer made of 6 Intel Core i7-3930K CPUs running at 3.20GHz. They compare quite well to the helicopter times of roughly 350 times realtime previously reported [5]. Also, the time required by the optimization algorithm is negligible with respect to that required by the multibody simulations.

FLIGHT PATH TESTS

The first evaluation of the automatically piloted aircraft is made by requesting a pure hover. The response and requested positions are shown for the helicopter, Heligyro, and Quadricyclogyro in Figs. 9 to 11.

In light of the testing done by Aldawoodi [5], some complex but not-extreme flight paths are requested to the three aircraft models and their ability to follow them accurately is observed. Aldawoodi [5] does also note that the autopilot develops inaccuracies which can increase in time. Also, the author relied on path following at very slow airspeed, in the order of 2 m/s, and thus this paper also uses slow speed verification of the developed autopilot. The helicopter used by Aldawoodi is a small unmanned helicopter which is presented in the thesis as a Matlab model developed by a group of students and of which the details are unknown. It can be assumed the model is thus a fairly simple helicopter representation. Nonetheless, the variable height figure eight path is used as a benchmark for the aircraft autopilot models. The *trickier* path [5] and the associated responses are plotted in Figs. 12 to 14.

Another other chosen path, which is more challenging due to

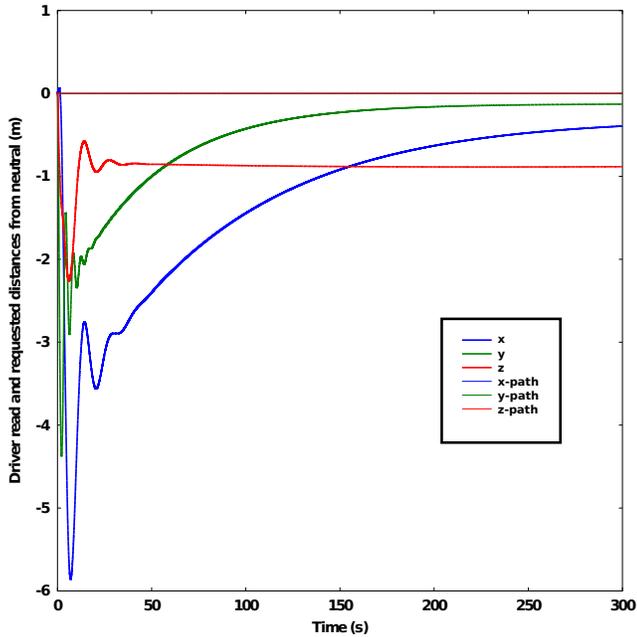


FIGURE 9. HELICOPTER PATH AND RESPONSE IN HOVER.

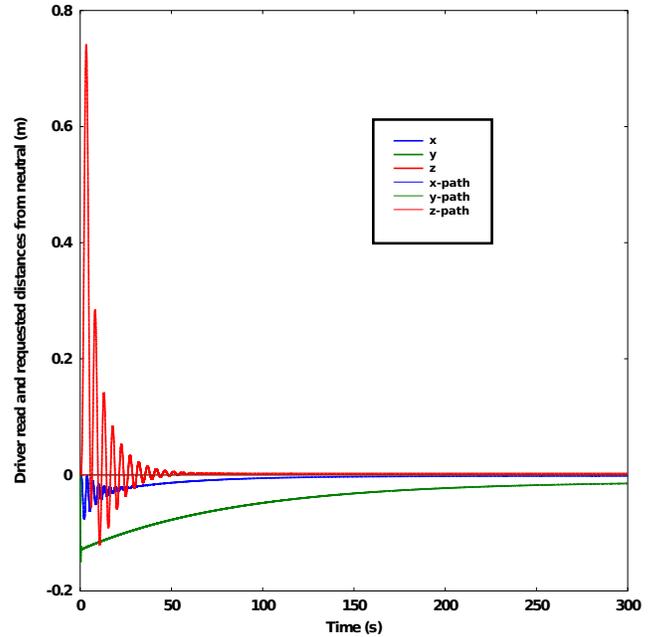


FIGURE 11. QUADRI. PATH AND RESPONSE IN HOVER.

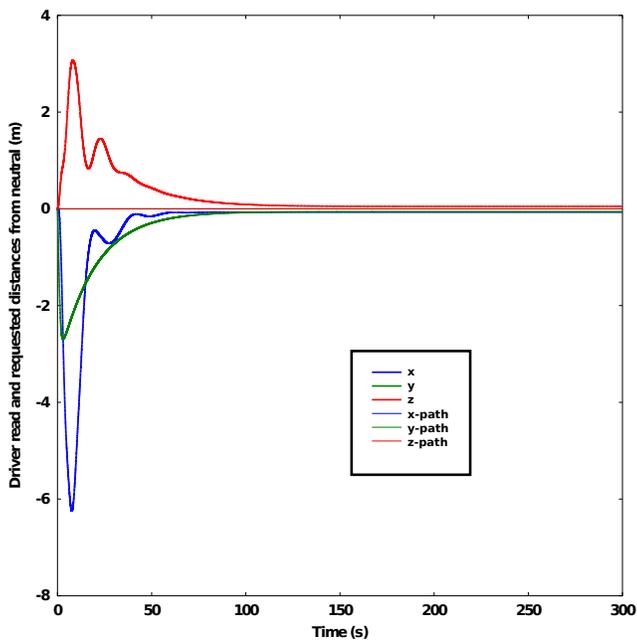


FIGURE 10. HELIGYRO PATH AND RESPONSE IN HOVER.

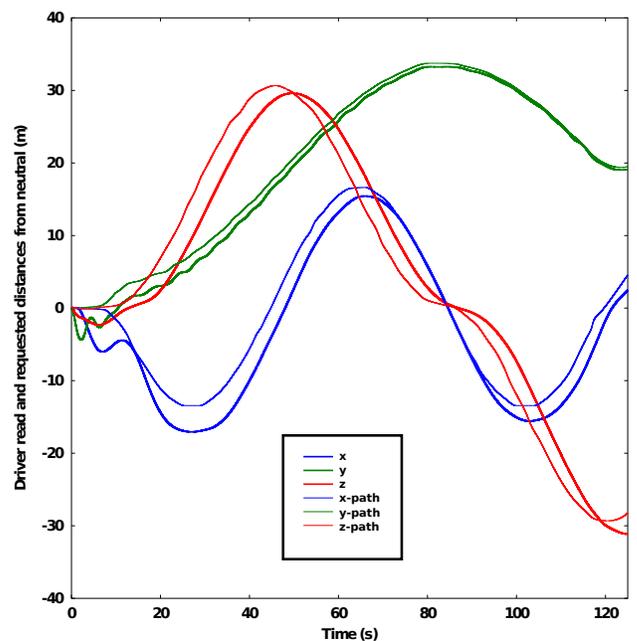


FIGURE 12. HELICOPTER TRICKIER PATH AND RESPONSE.

its higher velocity comes from a NASA report [1] treating a helicopter obstacle avoidance algorithm. The more complex path from the report has been chosen and is referred to as the *nasa* path. The original velocity is 10.29 m/s (20 knots) over and covers roughly 500 m. This high velocity or distance traveled cause

instability in the model and is thus also evaluated at half-speed. An initial hover period and gradual speedup are also provided to allow the aircraft to start the path at the correct speed. The responses of the Helicopter for both full and half speeds *nasa* trajectories are given in Figs. 15 and 16, respectively. The principal

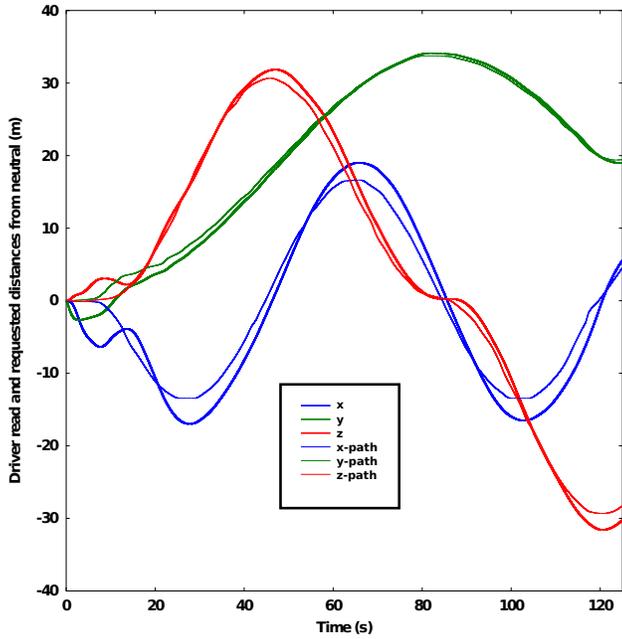


FIGURE 13. HELIGYRO TRICKIER PATH AND RESPONSE.

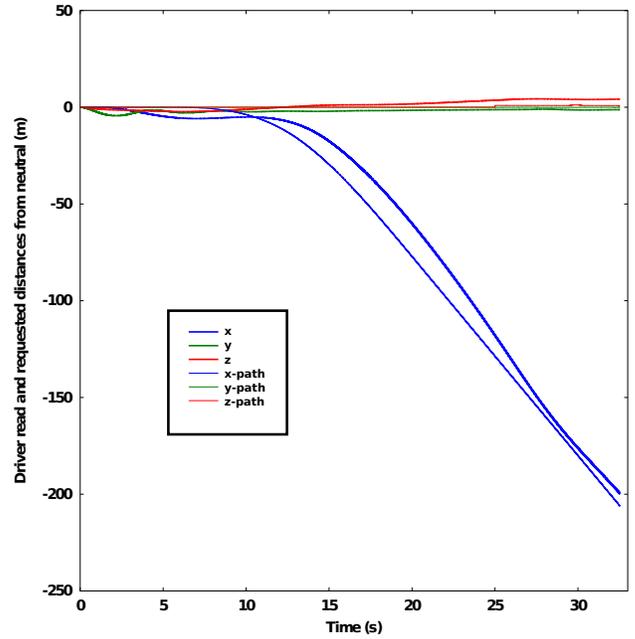


FIGURE 15. HELICOPTER NASA PATH AND RESPONSE.

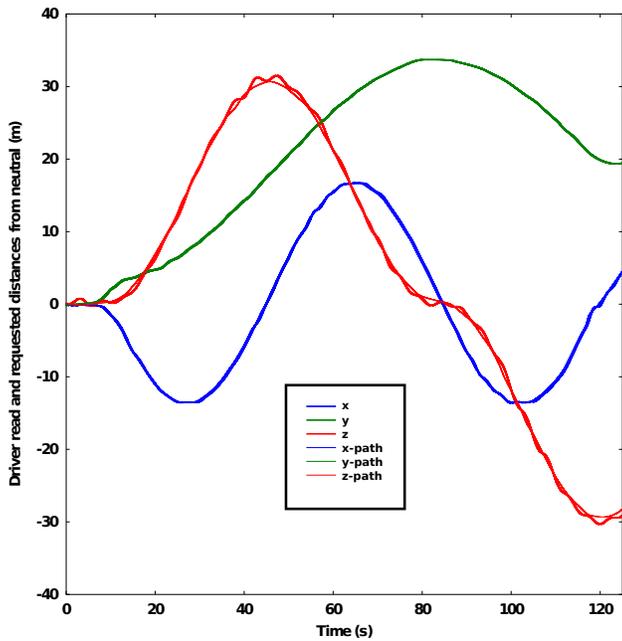


FIGURE 14. QUADRI. TRICKIER PATH AND RESPONSE.

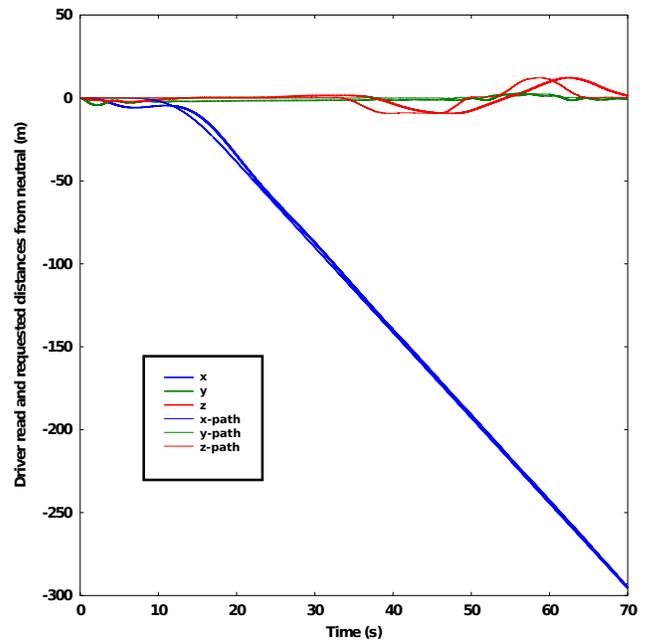


FIGURE 16. HELICOPTER NASA/2 PATH AND RESPONSE.

characteristics of this path are that the aircraft is brought up to a constant velocity and then is asked to reproduce the vertical and lateral displacements that were executed as obstacle avoidance maneuvers by NASA's algorithm.

The Heligyro has a harder time with the *nasa* paths. It is

initially able to follow the requested velocity, but nevertheless diverges before the end of the mission. Similarly, it initially follows the *generated* path, but the simulation diverges early, as seen in Fig. 17

A particular path, the *rodeo*, is created to test the capabilities

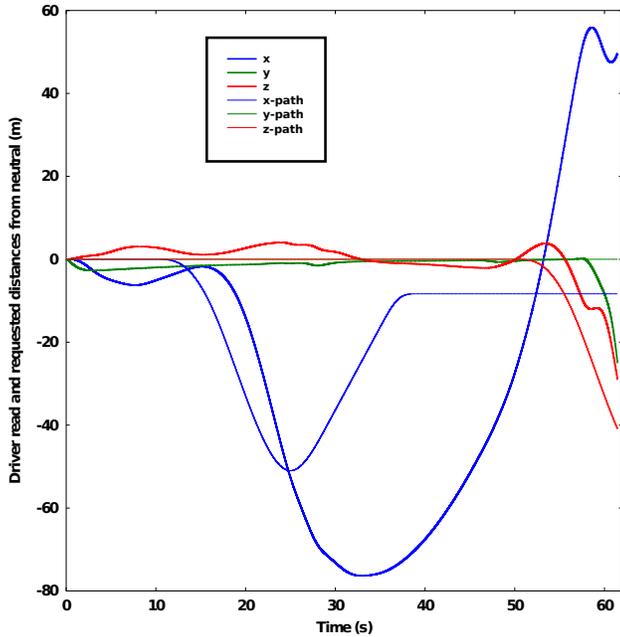


FIGURE 17. HELIGYRO GENERATED PATH AND RESPONSE.

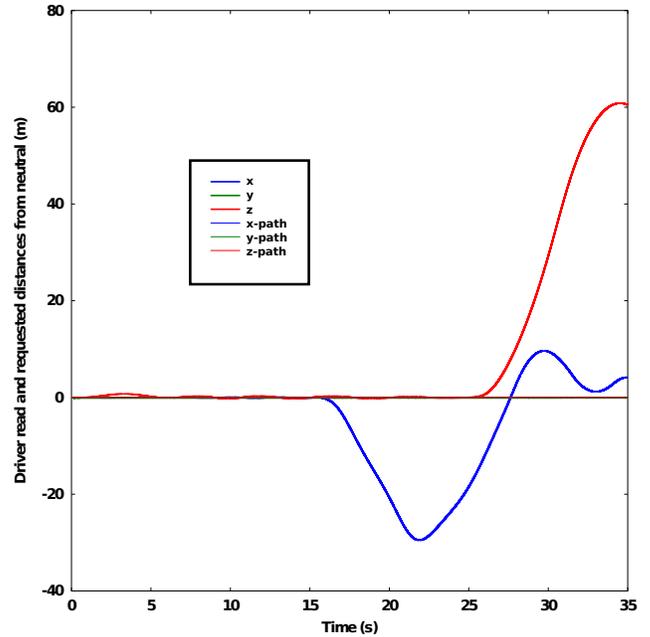


FIGURE 18. QUADRICYCLOGYRO RODEO PATH AND RESPONSE.

of the optimized Quadricyclogyro to sustain a pitched position in hover. The *rodeo* path requires three instant changes of attitude going from 0° to 30° . They each last 5 seconds, are padded with 5 second null attitude between each of them. The order is yaw, then pitch, and then roll. The response in both position and angles is given in Figs. 18 and 19, respectively. It highlights the limitations of the optimization which was not using angle-based paths to sustain a given pitched position in hover. This capability was shown in a prior work [16] for the same multibody model. Thus, this rather shows the limitations of using an optimization strategy focused entirely on linear path following which ignores attitude requests.

Finally, the power consumption of the helicopter and the Heligyro are given for the *complex* path in Figs. 20 and 21, respectively. They show that with the current configuration and controller, the Heligyro is not yet able to reduce the power consumption. This also applies to the *hover*, *generated*, and *nasa* paths.

CONCLUSION

The multibody models of the Quadricyclogyro and the Helicopter are shown to be very stable and follow the imposed path without unwanted vibrations. This is a promising result as they are able to do so even without using predictive measures or knowing ahead what the imposed path will be at the next timestep. The Quadricyclogyro is effectively showing the full potential of the near-instant response of cycloidal rotors in thrust

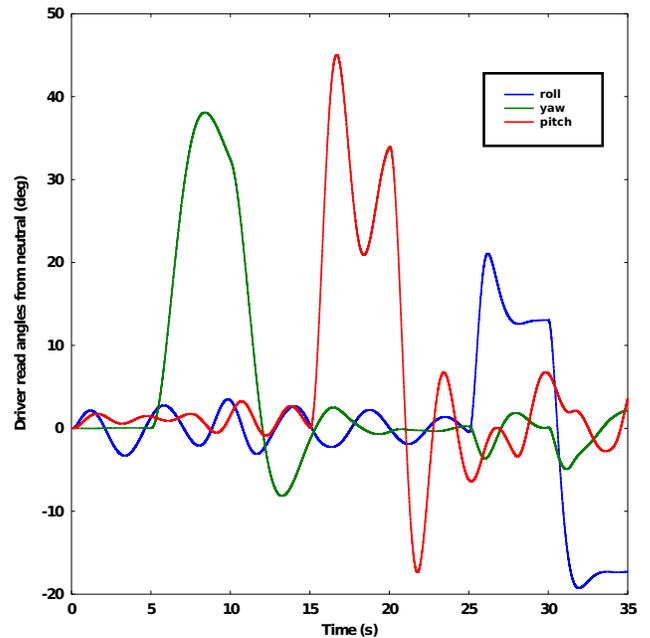


FIGURE 19. QUADRICYCLOGYRO RODEO PATH ANGLES.

directions by following the given paths almost perfectly. The Heligyro model has more difficulty in following the imposed path. Its control algorithm is somewhat more complex and more tuning must be done in order to determine the best control functions.

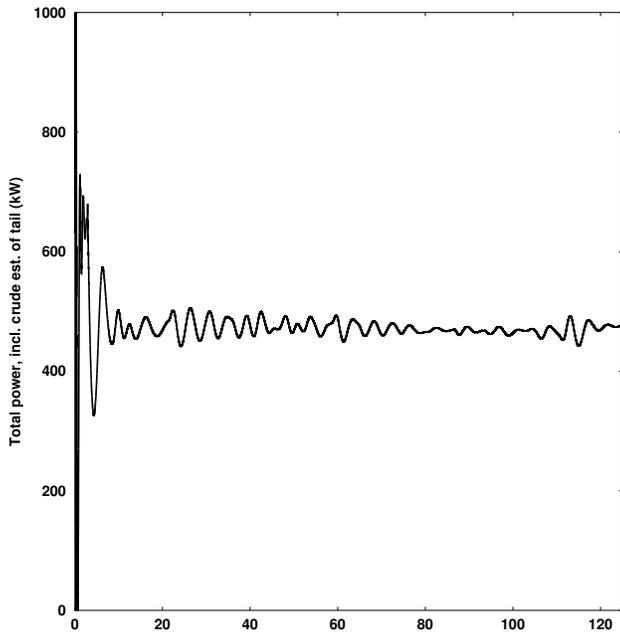


FIGURE 20. HELICOPTER TRICKIER PATH POWER.

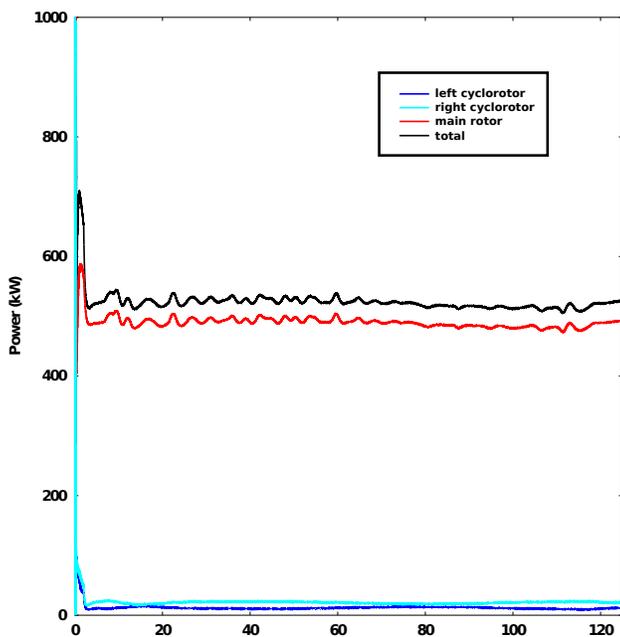


FIGURE 21. HELIGYRO TRICKIER PATH POWER.

It has the particularity of offering many different possibilities to control the aircraft and depending on a given flight configuration one may be more efficient than another. A predictor approach was not used as it was not deemed necessary since in order to test new aircraft concepts all is required is the ability to fly in

different conditions. A future step could include flexibility of the rotor blades and use larger optimization populations to better distribute the load on the high-performance computers. In its current state, the controller always keeps the default orientation of the aircraft, unless specifically requested to do so or required by the displacement drives. One improvement would be to allow orientation changes, which are currently not possible due to the controller's absolute coordinate system. Another aspect for further study is the Heligyro controller. There are many possibilities to improve it such as allowing combined contributions to roll and pitch from the main and cycloidal rotors. This would allow optimizing the Heligyro handling and energy efficiency. The currently implemented control algorithm is a simplistic proof of concept. It allowed this study to provide a basis for further investigation of cyclogyro concepts by rapidly testing many rotorcraft configurations.

ACKNOWLEDGMENT

This project was funded by the Polimi International Fellowship research grant Index no. 1378 - Ref. No. 15881 made available for the *An aeroelastic study of cycloidal rotors used in various configurations* project. High-performance computer time was funded through the *Italian SuperComputing Resource Allocation (ISCR)* project number HP10C82PIC: *AERodynamic study of cycloidal rotors used in various configurations (AERODAL)*. Finally, the work leading to this paper was made possible thanks to different open source software which principally include MBDyn for multibody dynamics, Dakota [25] for optimization, SageMath [26] for solution of analytical equations, and GNU Octave [27] for managing the optimization and automatically generating multibody models.

REFERENCES

- [1] Clement, W. F. G., 1991. Fully automatic guidance and control for rotorcraft nap-of-the-Earth flight following planned profiles. Volume 1: Real-time piloted simulation. Tech. rep., Jan.
- [2] Clement, W. F. G., 1991. Fully Automatic Guidance and Control for Rotorcraft Nap-of-the-earth Flight Following Planned Profiles. Volume 2: Mathematical Model. Tech. rep., Jan.
- [3] Masarati, P., Quaranta, G., Serafini, J., and Gennaretti, M., 2007. "Numerical investigation of aeroservoelastic rotorcraft-pilot coupling". In XIX Congresso Nazionale AIDAA.
- [4] Bottasso, C. L., and Riviello, L., 2007. "Trim of rotorcraft multibody models using neural-augmented model-predictive auto-pilot". *Multibody System Dynamics*, **18**(3), Oct, pp. 299–321.

- [5] Aldawoodi, N., 2008. “An approach to designing an unmanned helicopter autopilot using genetic algorithms and simulated annealing”. Ph.D., University of South Florida, June.
- [6] Trainelli, L., Croce, A., Riboldi, C., and Possamai, R., 2015. “Trimming a high-fidelity multibody helicopter model for performance and control analysis”. Vol. 2, pp. 1050–1061. cited By 0.
- [7] Barrass, C., 2004. “Chapter 22 - improvements in propeller performance”. In *Ship Design and Performance for Masters and Mates*, C. Barrass, ed. Butterworth-Heinemann, Oxford, pp. 218 – 227.
- [8] Yun, C. Y., Park, I. K., Lee, H. Y., Jung, J. S., and H., I. S., 2007. “Design of a New Unmanned Aerial Vehicle Cyclocopter”. *Journal of the American Helicopter Society*, **52**(1).
- [9] Benedict, M., Mattaboni, M., Chopra, I., and Masarati, P., 2011. “Aeroelastic Analysis of a Micro-Air-Vehicle-Scale Cycloidal Rotor”. *AIAA Journal*, **49**(11), pp. 2430–2443. doi:10.2514/1.J050756.
- [10] Lind, A. H., Jarugumilli, T., Benedict, M., Lakshminarayan, V. K., Jones, A. R., and Chopra, I., 2014. “Flow field studies on a micro-air-vehicle-scale cycloidal rotor in forward flight”. *Experiments in Fluids*, **55**(12).
- [11] Hu, Y., Du, F., and Zhang, H. L., 2016. “Investigation of unsteady aerodynamics effects in cycloidal rotor using RANS solver”. *The Aeronautical Journal*, **120**(1228), May, pp. 956–970.
- [12] Ilieva, G., Páscoa, J. C., Dumas, A., and Trancossi, M., 2012. “A critical review of propulsion concepts for modern airships”. *Central European Journal of Engineering*, **2**(2), pp. 189–200. doi:10.2478/s13531-011-0070-1.
- [13] Gagnon, L., Morandini, M., Quaranta, G., Muscarello, V., and Masarati, P., 2016. “Aerodynamic models for cycloidal rotor analysis”. *AEAT*, **88**(2), pp. 215 – 231. doi:10.1108/AEAT-02-2015-0047.
- [14] Gagnon, L., Morandini, M., Quaranta, G., Masarati, P., Xisto, C. M., and Páscoa, J. C., 2018. “Aeroelastic Analysis of a Cycloidal Rotor Under Various Operating Conditions”. *In press, AIAA Journal of Aircraft*.
- [15] Gagnon, L., Morandini, M., Quaranta, G., Muscarello, V., Masarati, P., Bindolino, G., Xisto, C. M., and Páscoa, J. C., 2014. “Feasibility assessment: a cycloidal rotor to replace conventional helicopter technology”. In 40th European Rotorcraft Forum.
- [16] Gagnon, L., Masarati, P., Morandini, M., and Quaranta, G., 2016. “Flight scenarios study of two cycloidal rotor aircraft concepts”. In The 4th Joint International Conference on Multibody System Dynamics.
- [17] Gagnon, L., Quaranta, G., and Schwaiger, M., 2018. “Open Source 3D CFD of a Quadrotor Cyclogyro Aircraft”. In *in press, 11th OpenFOAM Workshop Book*, J. M. Nbreaga and H. Jasak, eds. Springer International Publishing AG.
- [18] Gagnon, L., Quaranta, G., Schwaiger, M., and Wills, D., 2018. “Aerodynamic analysis of an unmanned cyclogiro aircraft”. *Submitted to SAE Tech Papers*.
- [19] Ghiringhelli, G. L., Masarati, P., Mantegazza, P., and Nixon, M. W., 1999. “Multi-body analysis of a tiltrotor configuration”. *Nonlinear Dynamics*, **19**(4), Aug, pp. 333–357.
- [20] Masarati, P., Morandini, M., Quaranta, G., and Mantegazza, P., 2003. “Open-Source Multibody Analysis Software”. In International Conference on Advances in Computational Multibody Dynamics.
- [21] Götz, J., 2006. GARTEUR HC AG16 Rotorcraft Pilot Coupling. Tech. rep., Institute of Flight Systems.
- [22] Morandini, M., and Mantegazza, P., 2007. “Using dense storage to solve small sparse linear systems”. *ACM Transactions on Mathematical Software*, **33**(1), mar, pp. 5–es.
- [23] Eddy, J., and Lewis, K., 2001. “effective generation of pareto sets using genetic programming”. In ASME Design Technical Conferences, Design Automation Conference, DETC01/DAC-21094.
- [24] Schwaiger, M., and Wills, D., 2016. “D-dalus vtol efficiency increase in forward flight”. *Aircraft Engineering and Aerospace Technology*, **88**(5), pp. 594–604.
- [25] Adams, B. M., Ebeida, M. S., Eldred, M. S., Jakeman, J. D., Swiler, L. P., Stephens, J. A., Vigil, D. M., Wildey, T. M., Bohnhoff, W. J., Eddy, J. P., Hu, K. T., Dalbey, K. R., Bauman, L. E., and Hough, P. D., 2014. Dakota, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis version 6.0 theory manual. Tech. rep., may.
- [26] The Sage Developers, 2017. *SageMath, the Sage Mathematics Software System (Version 7.5.1)*. <http://www.sagemath.org>.
- [27] John W. Eaton, David Bateman, S. H., and Wehbring, R., 2015. *GNU Octave version 4.0.0 manual: a high-level interactive language for numerical computations*.