The Foldable Drone: A Morphing Quadrotor That Can Squeeze and Fly

Davide Falanga ⓘD, Kevin Kleber, Stefano Mintchev ⓘD, Dario Floreano ⓘD, and Davide Scaramuzza ⓘD

Abstract—The recent advances in state estimation, perception, and navigation algorithms have significantly contributed to the ubiquitous use of quadrotors for inspection, mapping, and aerial imaging. To further increase the versatility of quadrotors, recent works investigated the use of an adaptive morphology, which consists of modifying the shape of the vehicle during flight to suit a specific task or environment. However, these works either increase the complexity of the platform or decrease its controllability. In this letter, we propose a novel, simpler, yet effective morphing design for quadrotors consisting of a frame with four independently rotating arms that fold around the main frame. To guarantee stable flight at all times, we exploit an optimal control strategy that adapts on the fly to the drone morphology. We demonstrate the versatility of the proposed adaptive morphology in different tasks, such as negotiation of narrow gaps, close inspection of vertical surfaces, and object grasping and transportation. The experiments are performed on an actual, fully autonomous quadrotor relying solely on onboard visual-inertial sensors and compute. No external motion tracking systems and computers are used. This is the first work showing stable flight without requiring any symmetry of the morphology.

Index Terms—Aerial systems: Applications, aerial systems: mechanics and control, motion control, robust/adaptive control of robotic systems.

I. INTRODUCTION

QUADROTORS are disrupting industries ranging from agriculture to transport, security, infrastructure, entertainment, and search and rescue [1]. Their maneuverability and hovering capabilities allow them to navigate through complex structures, inspect damaged buildings, and even explore underground tunnels and caves. Yet, current quadrotors still lack the

(a) H morphology. (b) O morphology. (c) T morphology.

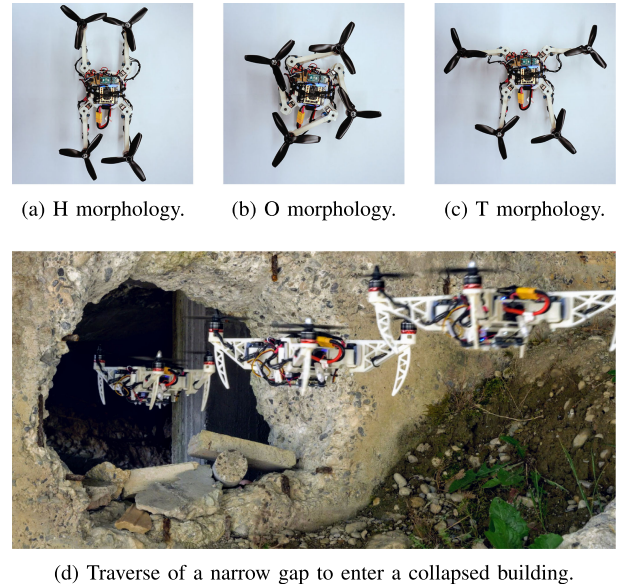(d) Traverse of a narrow gap to enter a collapsed building.

Fig. 1. Quadrotor with morphofunctional folding capabilities. The drone can transition from the standard X configuration to task-specific morphologies. (a) H configuration to fly through narrow vertical gaps. (b) O configuration, where the drone is fully folded to fly through horizontal gaps. (c) T configuration for proximity inspection of vertical surfaces. (d) Traverse of a gap narrower than the vehicle size using the H morphology. From right to left: the quadrotor approaches the gap with the X configuration; the vehicle initiates the folding maneuver to reach the H configuration; the gap is traversed using an elongated morphology to avoid collisions.

ability to adapt to different flight conditions and tasks, which is commonly observed in birds [2]. This would provide useful in complex scenarios, such as rescue and rescue missions or inspection of complex structures. For example, pigeons [3] and swifts [4] adapt their wing surface by folding in order to optimize gliding efficiency over a broad range of speeds. Pigeons have also been shown to choose different morphologies of their wings to negotiate gaps of different sizes: they fold the wings upward to negotiate relatively large vertical gaps, and fold them tight and close to their body in order to traverse narrower gaps [5]. In a similar way, a large drone could fold only when it has to fly in very cluttered environments [6]. In this way negotiation of narrow gaps can be achieved without miniaturizing the drone with consequent trade-offs in terms of flight time and payload. However, morphing quadrotors where the relative position or orientation of propellers can be modified during flight in order to extend the flight envelope remains a largely unexplored topic. The optimization of the relative orientation of the propellers [7] or the use of tiltable rotors have been investigated to increase the

(a) Riviere et al. [6].   (b) Zhao et al. [11].   (c) Zhao et al. [12].

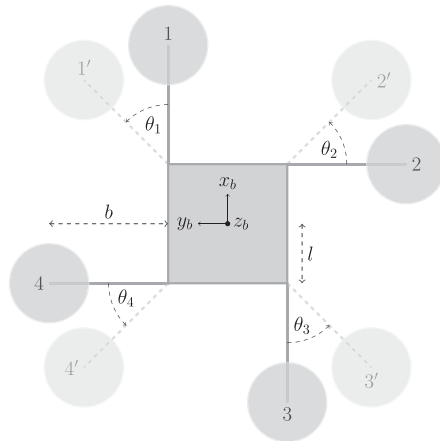Fig. 2.   Examples of other morphing aerial vehicles.

Fig. 3.   Schematics of our quadrotor, able to change its morphology while flying. Each propeller is connected to the main body through an arm, which can rotate with respect to the body thanks to a servo-motor. Each arm moves independently of the others, allowing asymmetric configurations.

controllability of hovering platforms [8]–[10]. Although these approaches facilitate the execution of complex trajectories and manipulation tasks, they do not entail significant shape change of the frame. Quadrotors with frames that morph during flight have been investigated by Zhao *et al.* [11], [12], Desbiez *et al.* [13], Riviere *et al.* [6] and Zhao *et al.* [14] in order to negotiate narrow gaps or grasp objects, each with their own advantages and trade-offs (cf. Fig. 2). For example, the robots in [6] and [13] can only fold into a narrow and elongated configuration (Fig. 2a), which allows flying through narrow vertical gaps, but hampers the negotiation of tight horizontal gaps. Once folded, the quadrotor is not able to guarantee a continuous stable flight and resorts to a ballistic motion to traverse the gaps. Therefore the drone needs a significant speed at the moment it negotiates the aperture, requiring a large space before and after the gap, which might not be available in cluttered environments. Another example is the morphing aerial vehicle composed of four serially connected links equipped with propellers proposed in [11]: this robot (Fig. 2b) is specifically conceived to wrap around objects and grasp them without the need of additional gripping device. In [12] the authors improved the morphing versatility of the drone to achieve 3D folding by departing from the standard quadrotor structure in favor of a multilink platform (Fig. 2c). In that work a basic assumption is that each joint is actuated very slowly. The aerial transformation is time consuming, hence hampering the prompt execution of complex maneuvers. Also, the mechanical design adopted by the authors requires a large number of components (i.e., four servo motors and two rotors for each actuation unit), increasing the complexity and weight of the robot. In [15] a quadrotor able to rotate and shrink its arms was presented. However, the approach proposed in that work is not able to handle non-symmetrical configurations, and only simulation results are presented. Finally, in [16] a control strategy for a flying robot with multiple degrees of freedom was proposed, and its application to a flying humanoid robot was shown.

A. Contributions

In this manuscript, we show how adaptive morphology can address the challenge of increasing quadrotors' versatility by tailoring their shape to different tasks, while limiting trade-offs such as degradation of flight time and maneuverability. The morphing approach consists of two elements working in synergy: a frame with four independently rotating arms that fold around the main frame (Fig. 3 and 1) and a control scheme able to take into account the current morphology of the vehicle to guarantee stable flight at all times. Each arm is connected to the main body through a servo motor and, to prevent the propellers from colliding with each other, adjacent motors have a vertical

offset. This simple morphing technique allows our vehicle to preserve the structural simplicity of quadrotors without requiring complex folding mechanisms [12] or tailoring it to specific applications [11].

Differently from [6], our quadrotor is able to guarantee stable flight independently of the morphology. The key challenge to do so is the need for an adaptive control scheme able to cope in real-time with the dynamic morphology of the vehicle. Any time a new morphology is adopted, our adaptive control strategy is updated in real-time to take into account the new geometry of the robot by i) computing the inertia matrix of the platform and ii) solving online an Algebraic Riccati Equation (ARE) to optimize the gains of a Linear Quadratic Regulator (LQR) responsible for controlling the body rates. Also, a morphology-dependent control allocation scheme is used to compute the required propellers speeds.

We validate the effectiveness of our approach on a small-scale, autonomous, vision-based quadrotor. We show that our adaptive control strategy is able to guarantee stable in-flight morphology transition during hovering and dynamic trajectories (up to 2 ms^{-1}), without requiring any symmetry of the robot geometry. We demonstrate that the proposed morphing strategy allows a quadrotor to adapt to different tasks: i) negotiation of narrow vertical gaps (Fig. 1a and 1c), ii) negotiation of narrow horizontal gaps (Fig. 1b), and iii) close proximity inspection structures (Fig. 1c). Finally, we show that the variable geometry of our quadrotor allows it to grasp and transport an object by wrapping the arms around it. Because our control and perception algorithms run directly onboard and do not need external tracking systems, we could demonstrate our drone outdoor to traverse a narrow gap and enter a partially collapsed building (see Fig. 1c).

B. Structure of the Paper

The remainder of this letter is organized as follows. In Sec. II we present our foldable quadrotor. In Sec. III we introduce the adaptive control scheme used to guarantee stable flight with any morphology. In Sec. IV we validate our approach on
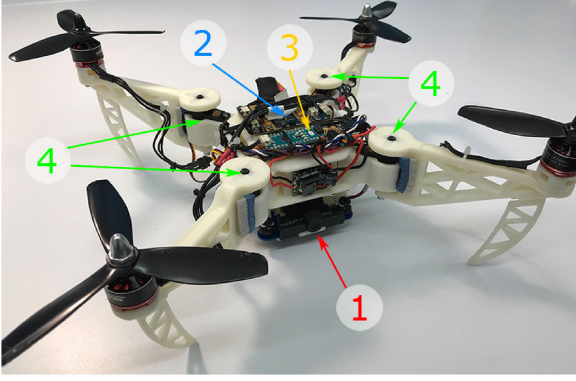
Fig. 4. A close-up picture of our foldable drone reporting the main component used. (a) The Qualcomm Snapdragon Flight onboard computer, provided with a quad-core ARM processor, 2 GB of RAM, an IMU and two cameras. (b) The Qualcomm Snapdragon Flight ESCs. (c) The Arduino Nano microcontroller. (d) The servo motors used to fold the arms.

a real platform and show real-world experiments. In Sec. V we draw the conclusions.

II. MECHANICAL DESIGN

Morphing systems require compromising between design complexity and shape shifting versatility. For instance, while 3D morphing frames can transition between varied and different shapes, the associated mechanical complexity could lead to cumbersome and heavy drones with limited flight time and payload [12]. 2D morphing strategies based on rotating links proved to be a reasonable compromise between feasibility and versatility [6], [11]. Avoiding singularities during morphing is another important aspect to consider in the selection of the morphing strategy to prevent complete control losses during flight [6]. We therefore decided to adopt the simple yet robust and versatile planar folding strategy composed of four folding arms as illustrated in Fig. 3.

The mechanical design of our foldable quadrotor is composed of two main parts: i) a central rigid body hosting the battery and the perception and control systems required for flight, and ii) four foldable arms with rotors. Each arm has an adjustable angle $\theta_i, i = 1, \ldots, 4$, around the body z_b axis, which is controlled by a servomotor hosted in the central body of the drone (see Fig. 4). The quadrotor can transition during flight form a standard X configuration (Fig. 4, $\theta_i = \pi/4, i = 1, \ldots, 4$) to task-specific morphologies while trading-off flight time and maneuverability. Once the task is concluded, the quadrotor re-assumes the X configuration recovering nominal flight efficiency and maneuverability. For example, by folding the front and rear arms forward and backward respectively, the quadrotor assumes a narrow H-configuration suited to fly through narrow vertical gaps (Fig. 1a, $\theta_1 = \theta_3 = 0, \theta_2 = \theta_4 = \pi/2$). However, this configuration has lower maneuverability along the roll axis than the standard X morphology. By folding all the four arms around the central body, the quadrotor undergoes a significant size reduction along both the x and y axis (Fig. 1b, $\theta_i = \pi, i = 1, \ldots, 4$). This fully folded morphology (O configuration) enables to fly through narrow horizontal gaps at the expense of major efficiency and

maneuverability reductions. By folding all the arms backward, the quadrotor assumes a T configuration with the frontal part of the drone clear from propellers (Fig. 1c, $\theta_1 = \theta_3 = \pi/2$, $\theta_2 = \theta_4 = 0$). This configuration exposes the sensorized central body of the drone, for example for the inspections of vertical surfaces.

III. CONTROL

The morphology of a quadrotor has a strong impact on its mechanical properties. Specifically, the folding of the arms has a direct impact on i) the location of the Center of Gravity (CoG) of the vehicle, ii) the inertia tensor of the platform, and iii) the mapping between the single rotor thrusts produced by the propellers and the forces and torques acting on the body. Therefore, a control strategy able to take into account these structural variations of the system to guarantee stable flight with any morphology is necessary.

A. Center of Gravity and Inertia

In standard quadrotors, the Center of Gravity is either considered to be located at the geometric center of the body or its offset with respect to this is estimated [17]. However, this assumption does not hold for our foldable quadrotor, as the arm angles $\theta_i, i = 1, \ldots, 4$, can be changed individually. The CoG, therefore, has to be recomputed when the configuration is adjusted. Similarly, the inertia matrix of the vehicle is morphology-dependent. Let $\theta_i, i = 1, \ldots, 4$, be the four angles of the servo motors actuating the arms. The offset $r_{\mathrm{CoG}} \in \mathbb{R}^3$ between the CoG and the geometric center of the vehicle is:

$$r_{\mathrm{CoG}} =$$

$$\frac{m_{body} r_{body} + \sum_{i=1}^{4} \left(m_{arm} r_{arm,i} + m_{mot} r_{mot,i} + m_{rot} r_{rot,i} \right)}{m_{body} + \sum_{i=1}^{4} \left(m_{arm,i} + m_{mot,i} + m_{rot,i} \right)},$$

(1)

where the position vectors r on the right-hand side of (1) are those of the corresponding part's own CoG. To simplify the computations, we refer the inertia tensor J of our foldable quadrotor with respect to the MAV's CoG. Specifically, J consists of the inertia tensors of the individual parts, which can be combined using the parallel axis theorem. We model the motors and rotors as cylinders. The arms are approximated as rectangular cuboids of length b, width w_{arm} and height h_{arm}. Finally, we model the central body as a box having length and width l, and height h_{body}, resulting in:

$$J_{body} = \frac{m_{body}}{12} \mathrm{diag}\left(h_{body}^2 + l^2, h_{body}^2 + l^2, l^2 + l^2 \right),$$

$$J_{arm} = \frac{m_{arm}}{12} \mathrm{diag}\left(w_{arm}^2 + h_{arm}^2, h_{arm}^2 + b^2, w_{arm}^2 + b^2 \right),$$

$$J_{mot} = \frac{m_{mot}}{12} \mathrm{diag}\left(3r_{mot}^2 + h_{mot}^2, 3r_{mot}^2 + h_{mot}^2, 6r_{mot}^2 \right),$$

$$J_{rot} = \frac{m_{rot}}{12} \mathrm{diag}\left(3r_{rot}^2 + h_{rot}^2, 3r_{rot}^2 + h_{rot}^2, 6r_{rot}^2 \right).$$

As the arms, motors, and rotors are rotated around z with respect to the body frame O_b, their inertia tensors must be rotated

as well. Since the inertia tensor of a cylinder does not change when rotated around its z-axis, this rotation can be neglected for the motors' and rotors' inertia tensors. The inertia tensor of the body does not have to be rotated, as the bodies' frame of reference is fixed to O_b. Accordingly, the inertia tensors for the arms can be represented as follows:

$$\boldsymbol{J}_{arm,i} = \boldsymbol{R}_z(\theta_i)\boldsymbol{J}_{arm}\boldsymbol{R}_z(\theta_i)^T \qquad i \in (1,2,3,4), \quad (2)$$

where $\boldsymbol{R}_z$ is the rotation matrix around z depending on θ_i. With these, we derived $\boldsymbol{J}$ as:

$$\begin{aligned}\boldsymbol{J} = \ &\boldsymbol{J}_{body} - m_{body}[\boldsymbol{r}_{body} - \boldsymbol{r}_{\text{CoG}}]^2 \\ &+ \sum_{i=1}^{4}(\boldsymbol{J}_{arm,i} - m_{arm}[\boldsymbol{r}_{arm,i} - \boldsymbol{r}_{\text{CoG}}]^2 \\ &+ \boldsymbol{J}_{mot} - m_{mot}[\boldsymbol{r}_{mot,i} - \boldsymbol{r}_{\text{CoG}}]^2 \\ &+ \boldsymbol{J}_{rot} - m_{rot}[\boldsymbol{r}_{rot,i} - \boldsymbol{r}_{\text{CoG}}]^2),\end{aligned} \quad (3)$$

with $[\boldsymbol{r}]$ being the skew-symmetric matrix of the vector $\boldsymbol{r}$.

B. Morphology-Dependent Control

Once the center of gravity and the inertia matrix for the current configuration are computed, it is necessary to adapt the control scheme. The morphology-dependent controller presented in the following assumes the rotational speed of the arms around the main body to be negligible (i.e., $\dot{\theta}_i \approx 0 \ \forall i$). This assumption does not represent an issue thanks to the fact that our adaptive controller continuously updates its parameters in order to cope with changes in the robot morphology. Whenever an arm is required to reach a new position, the rotation necessary to obtain it is divided into small steps and, for each step, the controller is adapted.

Since the arms can only rotate around axes parallel to the body z_b axis, the direction of the thrust produced by each propeller does not depend on the morphology. Therefore, position control, providing the desired collective thrust t_{des}, can be achieved following the standard model derived for fixed-geometry quadrotors [18] by using state-of-the-art nonlinear controllers [19]. On the contrary, attitude control, providing the desired body torques $\boldsymbol{\tau}_{des}$, requires a morphology-dependent and adaptive approach, since the configuration has an impact on the rotational dynamics.

The body rate controller used in this work is inspired by [20]. The dynamics of the quadrotor's body rates $\boldsymbol{\omega}$ are:

$$\dot{\boldsymbol{\omega}} = \boldsymbol{J}^{-1}\left(\boldsymbol{\tau} - \boldsymbol{\omega} \times \boldsymbol{J}\,\boldsymbol{\omega}\right). \quad (4)$$

We model the rotor thrusts f_i as first order systems:

$$\dot{f}_i = \frac{1}{\alpha}\left(f_{des,i} - f_i\right) \qquad i \in (1,2,3,4). \quad (5)$$

Assuming the coefficient relating the drag torque and the thrust of a single propeller k to be constant, for slowly changing geometry (5) leads to a first-order dynamics for the body torques:

$$\dot{\boldsymbol{\tau}} = \frac{1}{\alpha}\left(\boldsymbol{\tau}_{des} - \boldsymbol{\tau}\right). \quad (6)$$

Combining (4) and (6), we can establish a dynamic system with state $\boldsymbol{s} = [\boldsymbol{\omega}^T\ \boldsymbol{\tau}^T]^T$ and input $\boldsymbol{u} = \boldsymbol{\tau}_{des}$, which we lin-

earize around $\boldsymbol{\omega} = \boldsymbol{0}$ and $\boldsymbol{\tau} = \boldsymbol{0}$ obtaining:

$$\begin{bmatrix}\dot{\boldsymbol{\omega}} \\ \dot{\boldsymbol{\tau}}\end{bmatrix} = \underbrace{\begin{bmatrix}\boldsymbol{0} & \boldsymbol{J}^{-1} \\ \boldsymbol{0} & -\frac{1}{\alpha}\boldsymbol{I}_3\end{bmatrix}}_{A}\begin{bmatrix}\boldsymbol{\omega} \\ \boldsymbol{\tau}\end{bmatrix} + \underbrace{\begin{bmatrix}\boldsymbol{0} \\ \frac{1}{\alpha}\boldsymbol{I}_3\end{bmatrix}}_{B}\boldsymbol{\tau}_{des}. \quad (7)$$

We designed a continuous-time infinite-horizon linear-quadratic regulator (LQR) control law $\boldsymbol{u} = \boldsymbol{u}_0 + \mathbf{K}_{\text{LQR}}\left(\boldsymbol{s} - \boldsymbol{s}_{ref}\right)$ based on (7) in order to minimize the cost function:

$$\mathcal{L}(\boldsymbol{s}, \boldsymbol{u}) = \int \tilde{\boldsymbol{s}}^T\boldsymbol{Q}\tilde{\boldsymbol{s}} + \tilde{\boldsymbol{u}}^T\boldsymbol{R}\tilde{\boldsymbol{u}}\,dt, \quad (8)$$

where $\tilde{\boldsymbol{s}} = \boldsymbol{s} - \boldsymbol{s}_{ref}$, $\tilde{\boldsymbol{u}} = \boldsymbol{u} - \boldsymbol{u}_{ref}$, and $\boldsymbol{Q}$ and $\boldsymbol{R}$ are diagonal weight matrices. Furthermore, we added two terms to the resulting control law: i) a feedback-linearizing term $\hat{\boldsymbol{\omega}} \times \boldsymbol{J}\hat{\boldsymbol{\omega}}$, which compensates the coupling terms in the bodyrates dynamics (6); ii) a feed-forward term $\boldsymbol{J}\dot{\boldsymbol{\omega}}_{des}$ to guarantee that $\boldsymbol{\omega}_{des}$ is reached with $\dot{\boldsymbol{\omega}} = \dot{\boldsymbol{\omega}}_{des}$. This results in the following control policy:

$$\boldsymbol{\tau}_{des} = \mathbf{K}_{\text{LQR}}\begin{bmatrix}\boldsymbol{\omega}_{des} - \hat{\boldsymbol{\omega}} \\ \boldsymbol{\tau}_{ref} - \hat{\boldsymbol{\tau}}\end{bmatrix} + \hat{\boldsymbol{\omega}} \times \boldsymbol{J}\hat{\boldsymbol{\omega}} + \boldsymbol{J}\dot{\boldsymbol{\omega}}_{des}, \quad (9)$$

where $\hat{\boldsymbol{\omega}}$ and $\hat{\boldsymbol{\tau}}$ are the estimates of $\boldsymbol{\omega}$ and $\boldsymbol{\tau}$.

Since a stable controller is needed for changing system dynamics, we recompute the LQR gains online whenever the momentary configurations deviates significantly from the linearization point. This guarantees that the system can be stabilized in all possible configurations as long as this is feasible within the motor saturation limits. These solutions could also be precomputed and applied from a lookup-table (LUT), but our online computation has three main advantages: i) it can adapt to the systems exact momentary state without quantization error as in a LUT; ii) it does not require extensive re-computation on cost adjustment or other tuning; iii) it can handle online cost changes, which might be needed to adapt to many different task scenarios.

To minimize (8), the following Algebraic Riccati Equation must be solved:

$$\boldsymbol{A}^T\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} = 0, \quad (10)$$

Leading to the optimal gain matrix $\mathbf{K}_{\text{LQR}} = -\boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P}$. Since the arm configuration of the MAV substantially changes the inertial tensor, it has a significant influence on the body dynamics and therefore in the resulting LQR gain matrix $\mathbf{K}_{\text{LQR}}$. To guarantee stable flight, the LQR gains must be adapted in real-time. This can be achieved using value iteration known from dynamic programming. Specifically, we use the approach presented in [21] for the case of a linear system resulting in an iterative algorithm to solve the discrete Algebraic Riccati Equation. The iteration process can be summarized as an iteration over the matrix $\boldsymbol{P}$ as $\boldsymbol{P}_{i+1} = \boldsymbol{A}^T\boldsymbol{P}_i + \boldsymbol{Q} - \boldsymbol{A}^T\boldsymbol{P}_i\boldsymbol{B}\left(\boldsymbol{R} - \boldsymbol{B}^T\boldsymbol{P}_i\boldsymbol{B}\right)^{-1}\boldsymbol{B}^T\boldsymbol{P}_i\boldsymbol{A}$. Termination is done upon reaching a threshold in the relative norm of the matrix $\boldsymbol{P}$ between consecutive iterations. Further details are available in [21]. To solve the problem fast enough to guarantee real-time performances, we can start from the last known value for $\boldsymbol{P}$ and therefore initialize the iterative algorithm already close to the new solution. To ensure a robust control strategy over all execute configurations, we update the

dynamic model, linearization and LQR gains online based on the work in [22].

C. Control Allocation

Given the desired collective thrust t_{des} and torques $\boldsymbol{\tau}_{des}$, it is necessary to convert those into the thrust each propeller has to produce. Since our folding scheme does not modify the direction of the thrust produced by each propeller, the collective thrust t and the torque around the body z_b axis do not depend on the configuration, and their expression follows the standard quadrotor control allocation scheme [18].

The roll and pitch torques, τ_x and τ_y respectively, can be calculated as the first two components of the cross product η between the individual rotor's distance to the CoG and the rotor's thrust vector as:

$$\boldsymbol{\eta} = \sum_{i=1}^{4} (\boldsymbol{r}_{rotor,i} - \boldsymbol{r}_{\text{CoG}}) \times f_i \, e_z. \tag{11}$$

This results in the following mapping between the rotor thrusts $\boldsymbol{f}$ and the roll and pitch torques:

$$\begin{bmatrix} \tau_x \\ \tau_y \end{bmatrix} = M_{x,y} \boldsymbol{f}, \tag{12}$$

where $\boldsymbol{f} = \begin{bmatrix} f_1 & f_2 & f_3 & f_4 \end{bmatrix}^T$ and:

$$M_{x,y} = \begin{bmatrix} l+b\sin(\theta_1)-r_{\text{CoG},y} & -l-b\cos(\theta_1)+r_{\text{CoG},x} \\ -l-b\cos(\theta_2)-r_{\text{CoG},y} & -l-b\sin(\theta_2)+r_{\text{CoG},x} \\ -l-b\sin(\theta_3)-r_{\text{CoG},y} & l+b\cos(\theta_3)+r_{\text{CoG},x} \\ l+b\cos(\theta_4)-r_{\text{CoG},y} & l+b\sin(\theta_4)+r_{\text{CoG},x} \end{bmatrix}^T.$$

Replacing (12) in the control allocation matrix for a fixed-morphology quadrotor [18], we can compute the full thrust mapping equation and, by solving it with respect to $\boldsymbol{f}$, we can compute the desired single rotor thrusts.

IV. EXPERIMENTS

The supplementary video attached to this letter provides a summary of the experiments reported in the following. For an extended version of the videos reporting the experimental results we refer the reader to the project webpage:

http://rpg.ifi.uzh.ch/foldable_drone.

A. Experimental Platform

Our quadrotor is made from a 3D-printed frame accommodating the electronics necessary to guarantee autonomous flight, and the servomotors to fold the arms (cf. Fig. 4). At the end of each arm a 3 blades, 5 inch propeller is mounted on top of a Gemfan M1806L 2300 KV brushless motor. The motors are controlled by a Qualcomm Snapdragon Flight Electronic Speed Controller, which receives the desired rotor speed commands from a Qualcomm Snapdragon Flight board having a quad-core 2.26 GHz ARM processor and 2 GB of RAM. The Snapdragon Flight board also provides two cameras, one looking forward (used in our experiments to detect the vertical gap) and one looking down, tilted at 45° (used for state estimation and to

detect the horizontal gap), and an Inertial Measurement Unit (IMU). The vehicle has a take-off weight of 580 g and a tip-to-tip diagonal of 47 cm.

The folding mechanism is based on the use of a servomotor directly connected to each arm. We used HiTech HS-5070MH servo motors, which provide a range of about 170°. The servomotors are commanded through an Arduino Nano microcontroller, which generates the PWM signal based on the desired angle command received by the flight controller over a USB connection. The mechanics and electronics required for morphing have an overall weight of 65 g, which correspond to approximately 11% of the total weight of the platform. The combination of planar folding technique and non-backdrivable servomotors confers structural stiffness to the drone as proven by the lack of deformations and oscillations of the arms during flight. However, the current design is not crash resilient. Collisions force the arms to fold producing a torque overload on the servomotors. This limitation can be overcome with the integration of lightweight dual-stiffness mechanisms [23], [24] to decouple the arms from the servomotors during collisions.

All the computations necessary for autonomous flight are performed onboard. The state of the quadrotor (i.e., its position, orientation, linear and angular velocities) is estimated using the Visual-Inertial Odometry pipeline provided by the Qualcomm mvSDK. Such state estimate is fed to the flight stack described in Sec. III, which runs onboard using ROS.

B. Morphing Tradeoffs

For each configuration presented in this work (X, T, H, O) we run in-flight experiments and performed offline evaluations in order to assess their respective advantages and trade-offs. More specifically, we are interested in:

- Flight time: the time the quadrotor can fly, which is affected by the arm configuration due to the overlap between different propellers, as well as between propellers and the main body, and due to an asymmetric usage of the motors leading to over power consumption, for example in the T configuration;
- Maximum angular acceleration as controllability index: defined as the maximum angular acceleration the robot can produce in hover around the body x_b-y_b axes;
- Size: defined as the propeller tip-to-tip distance, for both the x_b and the y_b axes.

Fig. 5 provides a comparison among the different morphologies in terms of the aforementioned parameters, which are explained in the following. It is important to notice that the values reported in Fig. 5 are normalized by those obtained in the X configuration. In other words, for each parameter p_i in a configuration i, Fig. 5 reports the ratio $\frac{p_i}{p_X}$ (or its inverse, as for the size), where p_X is the same parameter evaluated in the X configuration. This is due to the fact that such a configuration is the most commonly used morphology for quadrotors, and, therefore, we took it as the reference model to evaluate advantages and disadvantages of the other configurations. Also, normalizing each value by the one obtained in the X configuration has the additional advantage of providing results that are less dependent on the specific hardware used to build our platform
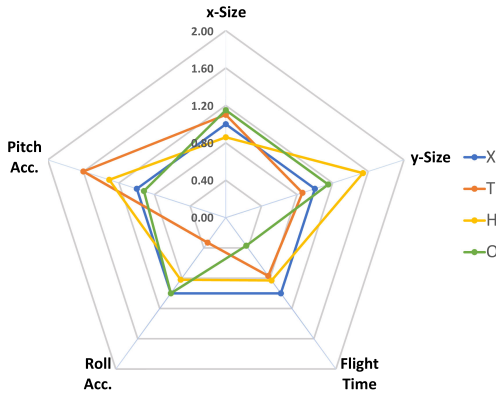
Fig. 5. Radar chart summarizing the comparison among the morphologies. We normalized each parameter to the one obtained for the X configuration, in order to provide an immediate overview about the advantages and disadvantages of each configuration compared to the classical X morphology.

and allow a more fair and general comparison among different morphologies.

1) Flight Time: The first parameter we are interested in is the flight time each configuration is capable of providing. Since flight in dynamic conditions is highly influenced by the kind of trajectory the vehicle flies, we performed our tests in hover conditions. In this regard, we let the vehicle autonomously hover while logging the battery voltage. We performed 10 trials for each configuration using a fully charged, 3-cells, Li-Po battery. It is well known that the discharge curve for LiPo batteries is linear only within a certain region [25]; therefore, we only considered such a region to compute the flight time. As expected, the X configuration is able to provide the best results and allows the vehicle to hover on average for 253 s. Changing the morphology of the drone causes a drop in the hover time of around 17%, 23%, and 63% for the H, T and O configurations, respectively. In the H configuration, this loss of endurance is partially due to the overlap between propellers. As shown in [26], when two propellers overlap, the thrust produced by the lower one depends on the vertical offset with respect to the upper one and the percentage of overlap. Our foldable quadrotor has a vertical offset between propellers of 2 cm. In the T and H configurations, the overlap is around 30% of the propeller radius, resulting in a loss of thrust for the lower propeller of around 5% [26]. The reduced flight time of the T configuration does not depend on propeller overlap, but rather on the robot geometry. In hover, rotors 1 and 2 need to rotate faster than rotors 3 and 4 due to their smaller distance to the CoG along the x_b axis (see Fig. 3). This leads to a higher power consumption in hover with the T configuration, since in near-hover conditions the power required by each motor scales with the cube of its rotational speed [27]. Finally, in the O configuration, the flight time is reduced even more because each propeller has a 30% overlap with the main frame. Our results confirm the intuition that morphologies different from the X are less efficient, which is especially emphasized with the O configuration where the vehicle is fully folded.

2) Angular Acceleration: The second parameter we used to compare the different morphologies is the maximum angular acceleration the vehicle can produce around its body x_b (roll)

and y_b (pitch) axes when hovering. This parameter is related to the agility and maneuverability of the platform, since it is an indicator of how fast the robot can rotate to accelerate laterally or forward. To calculate such acceleration, we first computed the maximum torque the vehicle can produce around each axis while simultaneously guaranteeing the hover thrust and satisfying the single motor thrust saturations. Then, we divided such torque by the inertia around the same rotation axis, obtaining the maximum instantaneous angular acceleration the quadrotor can produce. It is important to notice that the morphology of the robot plays a key role for this parameter and its contribution is twofold. On the one hand, folding or unfolding each arm around the main body changes the arm of the force produced by each propeller. This means that, for a propeller producing the same thrust, it can generate different torques depending on its position with respect to the fixed body. On the other hand, the inertia of the platform depends on how the arms are distributed around the main body and, the farther each propeller is with respect to the geometric center of the vehicle along one axis, the more it contributes to the inertia around the other two.

3) Size: Finally, we considered as last parameter of our analysis the size of the vehicle. More specifically, for each configuration we computed the tip-to-tip distance along the body's x_b and y_b. Fig. 5 reports the results of this analysis. It is important to note that, only for the size, we considered the inverse of the ratio $\frac{p_i}{p_X}$ to guarantee consistency with the other parameters, whose normalized values larger than one indicate an improvement with respect to the X morphology.

4) Conclusions: The ability of switching morphology allows a quadrotor to change its shape to optimize the execution of tasks that are difficult or impossible with the X configuration, such as passing through narrow gaps, as shown in the next section. However, this comes at a cost: the standard X morphology is the most efficient and therefore should be used as long as a different morphology is not strictly required by the task at hand. Additionally, as shown by the results in Fig. 5, reducing the size a drone by morphing does not always increase its agility. The T and H configurations, for example, are capable of providing higher angular accelerations around one of the body axes, but sacrifice their agility around the other axis. The O configuration, despite the significant size reduction, does not bring any advantage in terms of agility since the overall mass of the vehicle does not change.

C. Flight Performance

Our foldable quadrotor is able to change its morphology while flying, as shown in the attached video, where the quadrotor transitions across the four morphologies previously reported in hover conditions. Our folding scheme is able to provide stable hover flight in all such configurations, as shown in Tab. I, where the mean μ and standard deviation σ of the position error are shown for a flight of 60 s with each configuration. The position error does not show significant dependence on the configuration, except for the O morphology, where the overlap between the propellers and the mainframe causes a significant loss in the thrust produced by each rotor. Additionally, we performed experiments to show that, independently of the morphology, our

TABLE I
FLIGHT PERFORMANCE. STATISTICS FOR THE POSITION ERROR IN HOVER FOR
THE FOUR MORPHOLOGIES. MEAN μ AND STANDARD DEVIATION σ FOR THE
ABSOLUTE VALUE OF POSITION ERROR FOR 60 S OF HOVERING FLIGHT FOR
EACH CONFIGURATION, EXPRESSED IN METERS. DATA RECORDED FROM AN
OPTITRACK MOTION-CAPTURE SYSTEM

Morphology	μ [m]			σ [m]		
	x	y	z	x	y	z
X	0.022	0.014	0.024	0.009	0.008	0.006
T	0.029	0.026	0.031	0.011	0.023	0.007
H	0.035	0.022	0.029	0.008	0.015	0.024
O	0.084	0.127	0.119	0.022	0.012	0.014

Fig. 6. Traversal of an horizontal gap using the O morphology. Left: the quadrotor approaches the gap with the X configuration (time $t = 0$ s). Center: the quadrotor starts the folding maneuver to adapt its morphology to the shape of the gap ($t = 1$ s). Right: the gap has been traversed ($t = 2$ s).

quadrotor is able to reject external disturbances and return to the reference hover position when perturbed, as demonstrated by the results reported in the supplementary video. Finally, to show that our control scheme does not require any kind of geometric property of the morphology of the vehicle (e.g., symmetries), we performed an experiment where each servo motor is commanded to reach a randomly generated value within its range of motion.

Our foldable drone is capable of changing its morphology not only in hover, but also in dynamic conditions. To this regard, the supplementary material shows experiments where the vehicle is commanded to fly a circular trajectory at a given height while changing the configuration from X to T, H, and O. We chose a circular trajectory since this requires the quadrotor to rotate both around its x_b and y_b body axes. The robot flies at a speed of $2 \, \mathrm{ms}^{-1}$ on a circle of radius 1.5 m, at a height of 1.5 m, and is able to guarantee stable flight in all such configurations despite the large accelerations it is subject to.

D. Applications

Morphing allows adaptation to a broader range of tasks and, therefore, opens the door to new applications as for example, but not limited to, flight through gaps smaller than the vehicle's silhouette, proximity inspection of surfaces and object transportation. In this section we show how our foldable quadrotor can be exploited for these tasks.

1) Flight Through Narrow Gaps: Previous works addressing quadrotor flight through narrow gaps have shown that an aggressive maneuver is required to align the vehicle with the gap's orientation to avoid collisions [28], [29]. Flight through arbitrarily shaped gaps using monocular vision has also been shown in [30]. In all those works, the gap has to be large enough to let the vehicle pass through. Gaps smaller than the vehicle silhouette cannot be traversed due to the fixed morphology of the robot. This increases the risk of collisions with the gap and requires a large space for the vehicle to execute and recover from this maneuver, which might not be available in unknown environments. On the contrary, an adaptive morphology enables the vehicle to pass through gaps smaller than its size by folding the arms and flying at low speed to increase safety. This also allows the robot to require a smaller free space around the gap, since no recovery maneuver is necessary. An adaptive morphol-

ogy to allow a drone to pass through narrow gap was proposed in [6]. However, that vehicle can only change its morphology to one that lets it pass through vertical gaps and requires recovery maneuvering due to loss of controllability in the folded configuration.

An adaptive morphology, like the one we propose in this work, allows a quadrotor to safely fly through narrow gaps that are smaller than its size in the X configuration. For example, the H configuration lets our robot fly through vertical gaps as wide as $2(l + r)$, where l is the half-size of the central body and r the propeller radius. The O configuration reduces both the width and length the robot, allowing passing through small horizontal gaps with a square shape and as wide as $2(l + r)$.

The results of our experiments are reported in the supplementary material for the cases of a vertical and an horizontal gap (cf. Fig 6). In both cases, the quadrotor would collide with the frame of the gap if it would not fold the arms before traversing it. To detect the gap using on-board vision, we used the algorithm proposed in [28]. The vehicle approaches the gap with the standard X configuration, autonomously switches to a configuration that lets it traverse the gap, and finally returns to the X configuration to hover. The experiments reported in this work make use of the H configuration to traverse the vertical gap, and the O configuration to traverse the horizontal gap. The configuration to be used for each gap was decided in advance and set as a parameter in the control pipeline. The dimensions of the vertical gap are 28×26 cm, those of the horizontal gap 32×32 cm.

Additionally, we performed outdoor experiments to showcase the potential benefits of a quadrotor able to reduce its size in search-and-rescue missions by entering and exploring a collapsed building after traversing an aperture smaller than its size (cf. Fig. 1c). The video of the experiments demonstrates the feasibility of our approach in a post-disaster scenario, but nevertheless this only represents a first step towards the deployment of morphing quadrotors to the field.

2) Close Proximity Surface Inspection: The supplementary video shows the results of an experiment highlighting the benefits of the T configuration against the X morphology for surface inspection. Indeed, the T configuration allows the robot the position the onboard front-looking camera closer than it can when the arms are placed around the main body in the X configuration (cf. Fig. 7). If the inspection target is larger than the space between two propellers in the X morphology (i.e., the target cannot fit between two adjacent propellers), the shortest distance d from such a target that the front-looking camera can reach depends on the arm-length b and the propeller radius r, namely

Fig. 7. Inspection of a surface using the T configuration (left) and the X configuration (right).

$d = \frac{\sqrt{2}}{2}(b + 2r)$. Conversely, when the robot is able to fold its front arms to the side (i.e., $\theta_1 = \pi$, $\theta_2 = 0$), the camera can potentially get as close to the target as the propeller radius r. As shown in the video, the robot manages to bring the camera closer to the surface to inspect when it flies in the T configuration.

3) Object Grasping and Transportation: The drone can close its arms around objects to grasp and transport them. Although this strategy cannot replace a specialized end-effector, small and lightweight objects can be transported without the need of additional mechanisms. The supplementary video shows our foldable drone grasping an object from the hands of a human operator by changing its morphology to the H configuration. Once the object has been grasped, the vehicle flies to a delivery point, and drops the object by simply rearranging its morphology to the X configuration.

V. CONCLUSION

In this work, we presented a simple, yet effective morphing system for quadrotors that consists of four arms that can fold around the main body. Our approach does not require symmetries in the morphology to guarantee stable flight. We showed that simple morphing mechanisms combined with adaptive control strategies are a viable solution to broaden the spectrum of applications of quadrotors. This could lead to a paradigm shift in the research community towards novel morphing aerial vehicles. However, there are still a number of unsolved research questions, such as automatic morphology selection, exploitation of the morphology for improved flight at high-speed, and novel, bio-inspired mechanical designs.

REFERENCES

[1] D. Floreano and R. J. Wood, "Science, technology and the future of small autonomous drones," *Nature*, vol. 521, pp. 460–466, 2015.

[2] S. Mintchev and D. Floreano, "Adaptive morphology: A design principle for multimodal and multifunctional robots," *IEEE Robot. Autom. Mag.*, vol. 23, pp. 42–54, 2016.

[3] C. J. Pennycuick, "A wind-tunnel study of gliding flight in the pigeon Columba livia," *J. Exp. Biol.*, vol. 49, pp. 509–526, 1968.

[4] D. Lentink *et al.*, "How swifts control their glide performance with morphing wings," *Nature*, vol. 446, pp. 1082–1085, 2007.

[5] C. D. Williams and A. A. Biewener, "Pigeons trade efficiency for stability in response to level of challenge during confined flight," in *Proc. Nat. Acad. Sci. USA*, vol. 112, pp. 3392–3396, 2015.

[6] V. Riviere, A. Manecy, and S. Viollet, "Agile robotic fliers: A morphing-based approach," *Soft Robot.*, vol. 5, pp. 541–553, 2018.

[7] D. Brescianini and R. D'Andrea, "Design, modeling and control of an omni-directional aerial vehicle," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2016, pp. 3261–3266.

[8] M. Kamel *et al.*, "The voliro omnidirectional hexacopter: An agile and maneuverable tiltable-rotor aerial vehicle," *IEEE Robot. Autom. Mag.*, Oct. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8485627

[9] M. Ryll, H. H. Bülthoff, and P. R. Giordano, "Modeling and control of a quadrotor UAV with tilting propellers," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2012, pp. 4606–4613.

[10] M. Ryll, H. H. Bülthoff, and P. R. Giordano, "A novel overactuated quadrotor unmanned aerial vehicle: Modeling, control, and experimental validation," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 2, pp. 540–556, Mar. 2015.

[11] M. Zhao, K. Kawasaki, X. Chen, S. Noda, K. Okada, and M. Inaba, "Whole-body aerial manipulation by transformable multirotor with two-dimensional multilinks," in *Proc. IEEE Int. Conf. Robot. Automat.*, Jun. 2017, pp. 5175–5182.

[12] M. Zhao, T. Anzai, F. Shi, X. Chen, K. Okada, and M. Inaba, "Design, modeling, and control of an aerial robot dragon: A dual-rotor-embedded multilink robot with the ability of multi-degree-of-freedom aerial transformation," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, Apr. 2018.

[13] A. Desbiez, F. Expert, M. Boyron, J. Diperi, S. Viollet, and F. Ruffier, "X-Morf: A crash-separable quadrotor that morfs its X-geometry in flight," in *Proc. Workshop Res., Educ. Develop. Unmanned Aerial Syst.*, 2017, pp. 222–227.

[14] N. Zhao, Y. Luo, H. Deng, and Y. Shen, "The deformable quad-rotor: Design, kinematics and dynamics characterization, and flight performance validation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 2391–2396.

[15] D. Wallace, "Dynamics and control of a quadrotor with active geometric morphing," Master's thesis, Dept. Aeronaut. Astronaut., Univ. Washington, Seattle, WA, USA, 2016.

[16] D. Pucci, S. Traversaro, and F. Nori, "Momentum control of an underactuated flying humanoid robot," *IEEE Robot. Autom. Lett.*, vol. 3, no. 1, Jan. 2018.

[17] S. Lee, D. K. Giri, and H. Son, "Modeling and control of quadrotor UAV subject to variations in center of gravity and mass," in *Proc. Int. Conf. Ubiquitous Robots Ambient Intell.*, 2017, pp. 85–90.

[18] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 20–32, Sep. 2012.

[19] M. Faessler, A. Franchi, and D. Scaramuzza, "Differential flatness of quadrotor dynamics subject to rotor drag for accurate tracking of high-speed trajectories," *IEEE Robot. Autom. Lett.*, vol. 3, no. 2, pp. 620–626, Apr. 2018.

[20] M. Faessler, D. Falanga, and D. Scaramuzza, "Thrust mixing, saturation, and body-rate control for accurate aggressive quadrotor flight," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 476–482, Apr. 2017.

[21] A. Al-Tamimi and F. Lewis, "Discrete-time nonlinear HJB solution using approximate dynamic programming: Convergence proof," *IEEE Trans. Syst., Man, Cybern., B*, vol. 38, no. 4, pp. 943–949, Aug. 2008.

[22] P. Foehn and D. Scaramuzza, "Onboard state dependent LQR for agile quadrotors," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2018, pp. 6566–6572.

[23] S. Mintchev, S. de Rivaz, and D. Floreano, "Insect-inspired mechanical resilience for multicopters," *IEEE Robot. Autom. Lett.*, vol. 2, no. 3, pp. 1248–1255, Jul. 2017.

[24] S. Mintchev, J. Shintake, and D. Floreano, "Bioinspired dual-stiffness origami," *Sci. Robot.*, vol. 3, no. 20, Jul. 2018. [Online]. Available: http://robotics.sciencemag.org/content/3/20/eaau0275/tab-article-info

[25] L. W. Traub, "Calculation of constant power lithium battery discharge curves," *Batteries*, vol. 2, no. 2, 2016.

[26] G. Nandakumar, A. Srinivasan, and A. Thondiyath, "Theoretical and experimental investigations on the effect of overlap and offset on the design of a novel quadrotor configuration, VOOPS," *J. Intell. Robot. Syst.*, vol. 92, no. 34, pp. 615–628, Dec. 2018.

[27] M. Bangura, M. Melega, R. Naldi, and R. Mahony, "Aerodynamics of rotor blades for quadrotors," Dec. 2016. [Online]. Available: http://arxiv.org/abs/1601.00733

[28] D. Falanga, E. Mueggler, M. Faessler, and D. Scaramuzza, "Aggressive quadrotor flight through narrow gaps with onboard sensing and computing," in *Proc. IEEE Int. Conf. Robot. Automat.*, May 2017, pp. 5774–5781.

[29] G. Loianno, C. Brunner, G. McGrath, and V. Kumar, "Estimation, control, and planning for aggressive flight with a small quadrotor with a single camera and IMU," *IEEE Robot. Autom. Lett.*, vol. 2, no. 2, pp. 404–411, Apr. 2017.

[30] N. J. Sanket, C. D. Singh, K. Ganguly, C. Fermueller, and Y. Aloimonos, "Gapflyt: Active vision based minimalist structure-less gap detection for quadrotor flight," *IEEE Robot. Autom. Lett.*, vol. 3, no. 4, pp. 2799–2806, Oct. 2018.