Delft University of Technology

A tailless aerial robotic flapper reveals that flies use torque coupling in rapid banked turns

Karasek, Matej; Muijres, Florian T.; Wagter, Christophe De; Remes, Bart D.W.; De Croon, Guido C.H.E.

DOI
10.1126/science.aat0350

Publication date
2018

Document Version
Accepted author manuscript

Published in
Science

Citation (APA)
Karasek, M., Muijres, F. T., Wagter, C. D., Remes, B. D. W., & De Croon, G. C. H. E. (2018). A tailless aerial robotic flapper reveals that flies use torque coupling in rapid banked turns. *Science*, *361*(6407), 1089-1094. https://doi.org/10.1126/science.aat0350

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.

This work is downloaded from Delft University of Technology.
For technical reasons the number of authors shown on this cover page is limited to a maximum of 10.

A tailless aerial robotic flapper reveals that flies use torque coupling in rapid banked turns

(accepted version[§])

Authors: Matěj Karásek[1]*, Florian T. Muijres[2], Christophe De Wagter[1], Bart D.W. Remes[1], Guido C.H.E. de Croon[1]

Affiliations

[1]Micro Air Vehicle Laboratory, Control and Simulation, Delft University of Technology, Delft, The Netherlands.

[2]Experimental Zoology Group, Wageningen University and Research, Wageningen, The Netherlands.

*Correspondence to: m.karasek@tudelft.nl, microuav@gmail.com.

Abstract

Insects are among the most agile natural flyers. Hypotheses on their flight control cannot always be validated by experiments with animals or tethered robots. To this end, we developed a programmable and agile autonomous free-flying robot, controlled through bio-inspired motion changes of its flapping wings. Despite being 55 times larger than a fruit fly, the robot can accurately mimic the rapid escape maneuvers of flies, including a correcting yaw rotation towards the escape heading. Because the robot's yaw control was turned off, we showed that these yaw rotations result from passive, translation-induced aerodynamic coupling between yaw torque and the roll/pitch torques produced throughout the maneuver. The robot enables new methods for studying animal flight, while its flight characteristics allow for real-world flight missions.s

Main Text

Flying insects demonstrate extraordinary agility when they reject wind gusts (*1*), catch prey (*2*), or evade a human hand trying to swat them (*3*). Such aerobatic feats, enabled by unsteady aerodynamics (*4*, *5*), require both quick and precise reactions of their neural sensory-motoric control system (*6*). Research of the underlying mechanisms typically involves in-vivo observations of flight maneuvers, usually via high-speed cameras, and has proposed hypotheses on the possible control strategies during maneuvers such as saccades (*7*), evasive maneuvers (*3*) or aerial tumble recovery (*8*). These hypotheses are currently being tested using theoretical modelling (*8*), or using tethered dynamically-scaled robots (*9*). However, such tests are incomplete, as existing theoretical models lack sufficient fidelity and tethered robots cannot model the full, coupled and non-linear, multiple degree-of-freedom movement dynamics experienced during free flight.

[§] This is the author's version of the work. It is posted here by permission of the AAAS for personal use, not for redistribution. The definitive version was published in *Science*, (2018-09-14), doi: 10.1126/science.aat0350

Lately, many bio-inspired robotic platforms were developed trying to mimic the flapping flight of hummingbirds (*10*, *11*), bats (*12*), beetles (*13*) and even flies (*14*). Unlike traditional fixed wing and rotary wing robots that differ greatly from flying animals, and thus experience different flight dynamics, flapping-wing vehicles could be used to test existing hypotheses on animal flight control, and bring insights, through systematic and programmable experiments with known internal processes (*15*, *16*). Nevertheless, due to technological challenges coming from the stringent weight and size restrictions, most existing designs cannot match the flight performance of their biological counterparts, especially lacking the agility, and even sufficient power to take off or sufficient energy to fly for more than a minute. The state-of-the art so far, the Nano Hummingbird (*10*) and the Robobee (*14*) come close in terms of performance and autonomous flight control, respectively. However, the Nano Hummingbird can only be manually operated by a trained human pilot and thus cannot fly autonomously (*10*), while the fly-sized Robobee is tethered to an off-board power supply, which limits its maneuvering capabilities (*14*).

Here, we present a fully-programmable autonomous and freely flying insect-inspired robot that is agile both around hover and in fast forward flight (Fig. 1). The robot is tailless, and thus, like in flies (*5*, *8*), position and orientation in space are controlled solely through wing motion adjustments. When designing the robot, we build upon the reliable flapping mechanisms with flexible wings found in (*17*). Two wings on each side of the robot flap in counter-phase and clap and peel with each other to enhance the produced thrust (Movie S1), inspired by the clap-and-fling mechanism observed in nature during thrust-demanding tasks (*4*). The size, wing morphology and wing kinematics of our robot do not mimic any specific natural flyer but were instead optimized for maximal power efficiency when driven by miniature brushless DC motors, an engineering alternative to the powerful animal flight muscles. The resulting 28.2 g robot has a wingspan of 33 cm (Fig. 1A); its 14-cm-long wings have a flapping frequency of approximately 17 Hz in hover. The power efficiency of our robot enables a flight endurance of 5 minutes in hover, or a flight range of more than 1 km, on a single battery charge.

To control its orientation, the robot produces torques around the three orthogonal body axes via bio-inspired adjustments of the wingbeat pattern (Fig. 1H–J). Like in fruit flies (*5*, *8*), yaw torque is produced by changing the wing root angle such that the (wingbeat-average) thrust vectors of the left and right wing(s) are tilted in opposite directions (Fig. 1E,H). Pitch torques are generated by adjusting the dihedral angle (center line of the flapping wings), which as in fruit flies (*5*), shifts the wing thrust vectors relative to the center of mass (CoM) (Fig. 1F,I). Roll torques are produced by generating thrust difference of the left and right wing (pair). Flies, with a coupled wingbeat actuation system, achieve this using asymmetric stroke amplitude adjustments (*5*), whereas our robot adjusts stroke frequencies (Fig. 1G,J). Finally, symmetric flapping frequency variation is used for modulation of thrust magnitude. The resulting control torques and thrust have high magnitude (Fig. S1,S2), bandwidth (Fig. S3), very little coupling (Figs. S3-S5), and are minimally sensitive to the vertical location of the CoM (Fig. S5E, F) (*18*).

Tailless flapping flight is unstable near hover (*19*), and thus the robot needs to actively stabilize its attitude (orientation in space). To this end, it carries a miniature 2.8 g autopilot (*20*) equipped with a programmable microcomputer and sensors for attitude estimation (*18*), i.e., a 3-axis gyroscope and accelerometer (Fig. S6). Attitude control

(Fig. S7) uses the estimated attitude as well as the body rotation rate signals from the gyroscopes (*18*), and is on the short-term in many respects comparable to the haltere-based PI controller suggested to be used by fruit-flies (*5*, *8*). Apart from remotely-piloted operation, the autopilot allows to execute pre-programmed control sequences, triggered by the pilot.

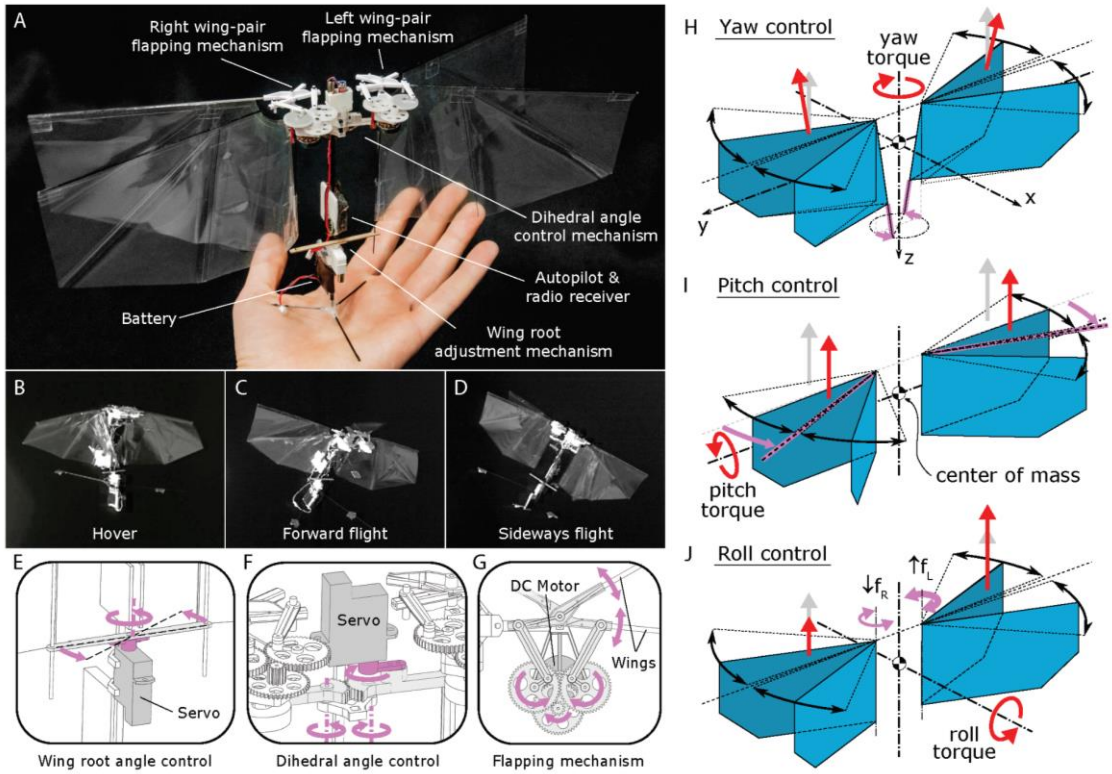

Fig. 1. **An insect-inspired free-flying robotic platform is controlled through its two pairs of independently flapping wings.**
(**A**) description of the robot's components, (**B-D**) high speed camera frames capturing the robot in hover (**B**), forward flight (**C**), and sideways flight (**D**), from Movies S1-S3, respectively. (**E-G**) Details on the robot design: (**E**) the wing root adjustment mechanism for yaw torque control, (**F**) the dihedral control mechanism for pitch torque control, and (**G**) the flapping mechanism (of the left wing pair), used for thrust and roll torque control. (**H-J**) Wing actuation and aerodynamic forces and torques during yaw control (**H**), pitch control (**I**), and roll control (**J**). Magenta arrows represent actuation action, gray arrows the nominal wingbeat-average aerodynamic thrust vectors, and red arrows show wingbeat-average thrust and torques after control actuation.

Similar to many flying insects (*3*, *5*), the robot can actively control 4 degrees-of-freedom (DOFs): roll, pitch, yaw and thrust. To move in a 6 DOF space, translation control follows the helicopter model, whereby forward/backward flight speed is controlled via body pitch (Fig. 1C, Movie S2), and sideways flight is achieved by rolling the body left or right (Fig. 1D, Movie S3). The robot can perform stationary hovering flight and move in any direction with rapid, yet smooth and reproducible transitions (Figs. S9-S12, Movie S4). In forward flight, the highest power efficiency is achieved around 3 m/s; applying full power results in the maximal speed of 7 m/s. The maximal sideways speed is 4 m/s.

The key property of interest here is the robot's agility, which is exceptional for a flapping wing robot and approaches that of rotorcraft robots (*21*). It can perform 360° roll and

pitch flips, during which it reaches angular accelerations of up to about 5000° s^{-2} (Figs. S13, S14, Movie S5, S6). Its maximal thrust to weight ratio of 1.3 allows for quick climbs, although it remains inferior to the best natural fliers with thrust-to-weight ratios ≥2 (*3*).

Despite the large variation in size, mass, wing kinematics and wing morphology among flapping wing fliers, they all experience the same type of passive aerodynamic damping, termed flapping-counter-torque and flapping-counter-force (*22*), and thus their flight dynamics share similar characteristics (*19*). We can thus use the robot to study maneuver dynamics and control in a wide range of flying animals, from similarly-sized hummingbirds to tiny two-winged fruit flies. To demonstrate this potential for animal flight research, we have programmed the robot to mimic the rapid banked turns observed in fruit flies when evading predators (Movie S7) (*3*).

It has been hypothesized that these evasive maneuvers consist of two distinct phases (*3*, *5*). The first phase is simply a feedforward program triggered by the visually-detected threat, whereby the fly produces a combined roll and pitch rotation, with the ratio of pitch-rate to roll-rate (q/p) defining the turn angle. The second recovery phase is controlled using sensory feedback provided by the halteres. It has also been hypothesized that fruit flies do not control body yaw throughout the evasive maneuver, but instead control for the sideslip caused by the maneuver after the turn, possibly by using much slower visual feedback (*3*, *5*, *9*).

We tested these two hypotheses by reproducing a range of evasive maneuvers described in fruit flies, which were preprogrammed and fully autonomous to assure the control inputs were always the same. The initial phase of the maneuver was controlled using an open-loop (OL) combination of pitch and roll torque commands (*18*). For the recovery phase, we used feedback control of roll and pitch, whereas yaw control was turned off.

In our first set of experiments, we performed two evasive maneuvers initiated with different pitch-to-roll rate ratios: the first dominated by rolling (q/p =0.52, Figs. 2A, S15), and the second dominated by pitching (q/p = 1.67, Figs. 2C, S16). Despite significant differences between our four-winged robot and the much smaller two-winged fruit fly, the robotic maneuvers resembled those observed in fruit flies remarkably well (Fig. 2A–D, Movies S8, S9). To be able to quantitatively compare the fruit fly and robot maneuvers, we normalized distances by wing length and time by the wingbeat-cycle period (flies) and half-the-wingbeat-cycle period (robot) (Figs. S17, S18) (*18*), and determined the Pearson's correlation coefficient c_r and normalized root-mean-squared error *rmse* between the normalized robot and fruit fly data (Table S5). For both maneuvers (Fig. 2A-D), we found high correlations and low *rmse* (Table S5), further highlighting the similarity.

Analyzing a 90-degree turn initiated with 94% of the maximal pitch torque and 74% of the maximal roll torque (q/p = 0.54) in detail (Figs. 2E–I, S19, Movie S10), showed that at the apex of the turn, an extreme body attitude of ~100° in roll is reached, which is comparable to the equivalent attitudes observed in fruit flies (Fig. 2B) (*3*). In the subsequent feedback phase the transition between the positive and negative acceleration peaks (difference of 9000° s^{-2}) took less than 3 wingbeats (≈0.18 s), while closely following the wing actuation that generates the body-accelerating torques (Fig. 2G,I).

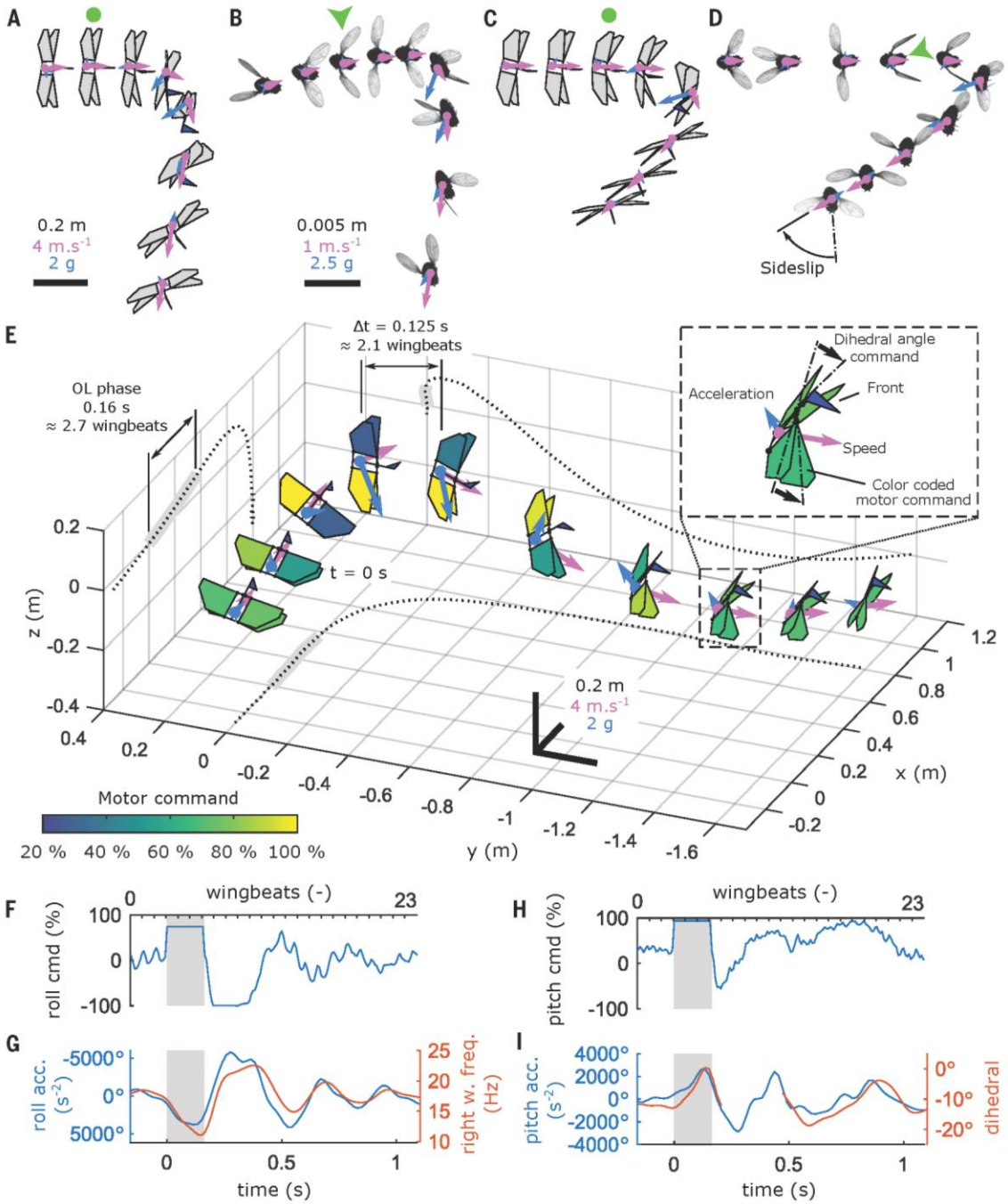

Fig. 2. **The robot mimics rapid banked turns observed in escaping fruit flies.**
(**A**,**C**) Time sequences (top view) of a roll-dominated maneuver (pitch-to-roll rate q/p = 0.52) and a pitch-dominated maneuver (q/p = 1.67), respectively. The start of the OL phase is marked with a green circle. (**B**,**D**) Time sequences (top view) of the equivalent fruit fly evasive maneuvers with the stimulus (green arrow) coming from the left and the front, respectively. (**E–I**) Detailed analysis of a roll dominated maneuver (q/p = 0.54). (**E**) Time sequence with constant time interval of 0.125 s, and trajectory projections (black dotted line). Wings are color-coded with thrust command magnitude, see color bar. (A–E) Blue and magenta arrows represent velocity and acceleration vectors, respectively. Vectors for the robot and flies are scaled by the black scale bars in (A,E) and (B), respectively. (**F–I**) Time histories of roll command (F), roll acceleration and flapping frequency of the right wing pair (G), pitch command (H), and pitch acceleration and dihedral angle (I). The OL phase is highlighted by gray background.

These results confirm that a rapid banked turn can be successfully achieved with an open-loop maneuver initiation and closed-loop recovery, supporting the hypothesis that the recovery phase of the evasive maneuver in fruit flies is also controlled using a PI-like roll and pitch control system, whereby halteres might provide the sensory input (*3*, *5*).

To systematically test the relation between *q/p* and turn angle, we performed evasive maneuvers at 5 combinations of roll and pitch torque commands, varying *q/p* from ~0.3 to ~2.1 (Figs. 3, S20) (*18*). *q/p* at the end of the OL phase had a clear effect on the flight trajectory (Fig. 3A), and in agreement with fruit flies, *q/p* was positively correlated with turn angle (c_r = 0.95, Fig. 3F), whereby turns with dominant pitch rates resulted in larger turn angles.

In agreement with the fruit fly experiments (*3*), we also observed high yaw rates in the direction of the turn during the recovery phase (Fig. 3D). Because no yaw command was provided during the robot maneuvers, these yaw movements were thus passively induced (Fig. S21).

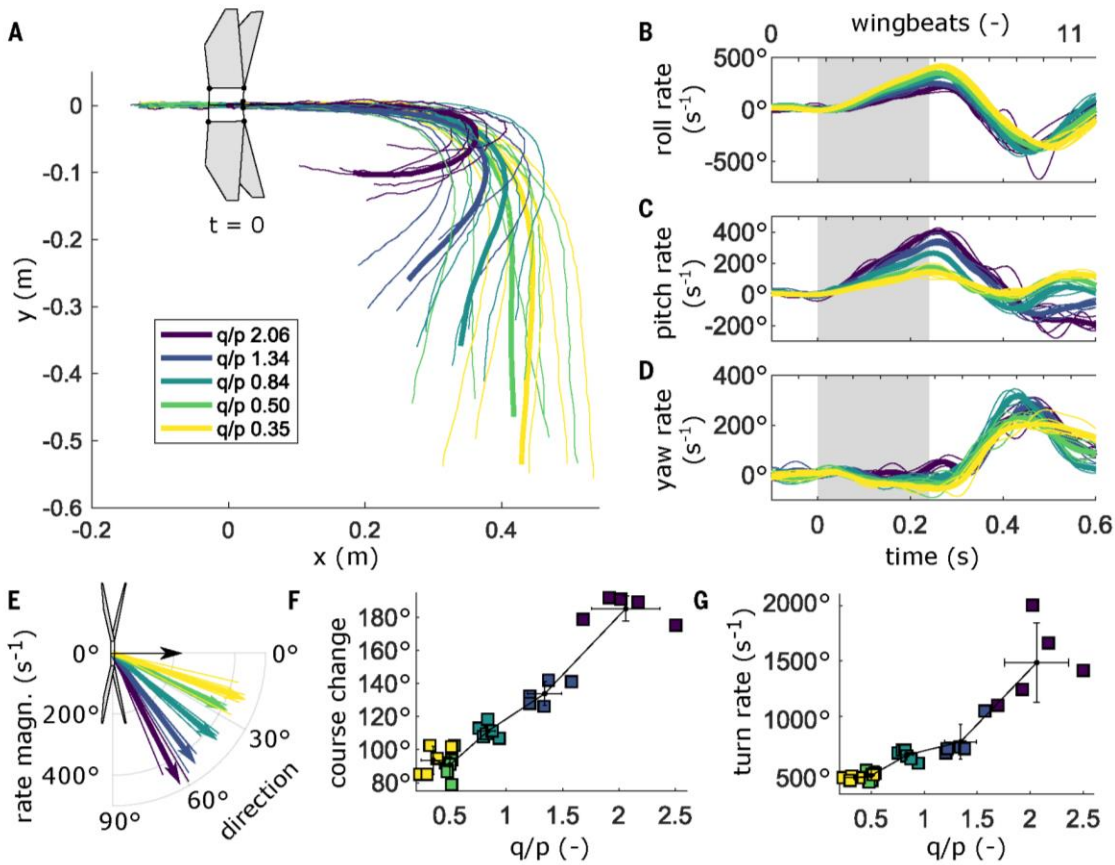

Fig. 3. **The turn angle during the banked turn can be controlled by varying the ratio of pitch to roll torque commands.**

Results are color-coded according to pitch-to-roll rate of the maneuver (*q/p*), see legend in (A); thin lines are individual trials, and thick lines series averages. (**A**) Top-view of trajectories, aligned at the start of the OL phase (*t* = 0 s) where the robot is shown. (**B–D**) Time histories of angular rates during the maneuvers. The OL phase is highlighted by gray background. (**E**) angular rate vector in the horizontal body plane, relative to the forward-directed black arrow. (**F,G**) Turn angle and turn rate versus *q/p*, respectively, for individual tests (squares) and mean±standard deviation per condition (error bar).

For the robot as well as the fruit flies, the observed yaw accelerations $\dot{r}$ correlate strongly with roll accelerations $\dot{p}$, and particularly for the sharper turns (high q/p), also with the pitch accelerations $\dot{q}$ (Fig. 4A,B). Moreover, the largest yaw accelerations were observed in the recovery phase, where both the robot and the flies move at relatively high forward and sideways flight velocities (Fig. S22).

Based on these observations, and thanks to the simple wing kinematics of the robot where roll and pitch torque are each modulated by only one parameter, we were able to develop a functional aerodynamic yaw torque model that explains the observed yaw accelerations in the robot maneuvers (Figs. 4D-H, S23) (*18*):

$$N = \frac{\dot{r}}{I_{zz}} = -b\Phi R\left[(f_L + f_R)Rr + (f_L - f_R)u + (f_L + f_R)\Gamma v\right]. \tag{1}$$

Here, N is yaw torque, $\dot{r}$ yaw acceleration, I_{zz} the moment of inertia around the vertical body axis, b is the linear damping coefficient due to flapping-counter-force (*22*), Φ is flapping amplitude, R is wing length, f_L and f_R are flapping frequency of the left and the right wing, respectively, r is yaw rate, u and v are forward and sideways velocity of the robot, respectively, and Γ is the wing dihedral angle.

The first term $-b\Phi R^2(f_L + f_R)r$ represents the flapping-counter-torque (N_{FCT}) that is, per definition, opposite to the direction of yaw rate (Fig. 4F) (*22*), and can thus not explain the observed yaw accelerations. The two remaining terms are our new addition to the passive yaw-torque system for flapping flight, which from here we will call translation-induced coupled yaw torque (N_{TCT}), as it models the coupling effect of roll and pitch torque production on yaw torque in the presence of translating body motions ($N_{\mathrm{TCT}} = N_{u\&\mathrm{roll}} + N_{v\&\mathrm{pitch}}$, Fig. 4G-H). Thus, in the presence of a forward velocity, u, the differential flapping frequency $(f_L - f_R)$ used for roll torque production results in also a yaw torque ($N_{u\&\mathrm{roll}}$, Fig. 4G). Similarly, in the presence of a sideways velocity, v, the dihedral angle Γ used for pitch torque production equally generates a yaw torque ($N_{v\&\mathrm{pitch}}$, Fig. 4H). In steady hover conditions, such coupling is negligible (Fig. S4).

Despite its low complexity, this model accurately estimates the observed yaw acceleration in both roll-dominated and pitch-dominated banked turn maneuvers (Fig. 4D–E, c_r =0.73±0.30 (mean±standard deviation), for the 5 sets of trials in Fig. S24 and Table S6). Notably, the model predicts the same coupling effects also when differential flapping amplitudes are used for roll torque production (Eqs. S10–S14, Fig. S23), which is what many biological fliers as well as other aerial robotic flappers employ.

Due to the high complexity of fruit fly wing motion patterns, we were unable to adapt the model for fruit flies where multiple degrees of freedom are involved in generating roll and pitch torques. For the fruit fly maneuvers, we estimated yaw accelerations resulting from translation-induced coupled yaw torque as the measured acceleration minus the accelerations caused by flapping-counter-torque (Eq. S19). These flapping-counter-torque corrected yaw accelerations correlate even better with both the product of roll acceleration and forward velocity, and the product of pitch acceleration and sideways velocity (Fig. 4C) than our initial yaw-pitch and yaw-roll correlations (Fig. 4B),

suggesting that translation-induced coupled yaw torque is also present in the fruit fly maneuvers.

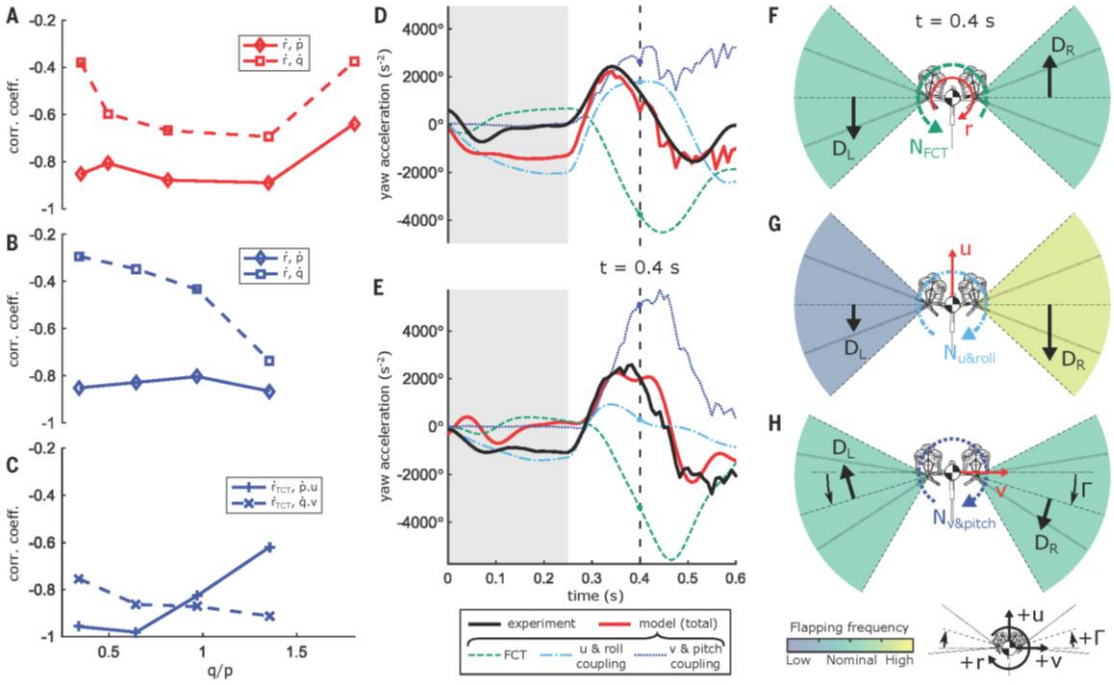

Fig. 4. **The passive yaw accelerations during the recovery phase of banked turns originate from the coupling between the roll and pitch torque generation mechanisms and translational body motions.**
(**A,B**) Correlation coefficient between the yaw acceleration and the roll (solid) and pitch (dashed) accelerations at various pitch-to-roll rate ratios *q*/*p*, for rapid banked turns produced by the robot (**A**), and for evasive maneuvers produced by fruit flies (**B**). (**C**) Correlation coefficients for the same evasive maneuvers of fruit flies, after correcting for the flapping-counter-torque and including the translational body velocities. (**D**–**E**) Measured (black) and modeled (red) yaw accelerations during the banked turn of the robot: (**D**) roll dominated banked turn (*q*/*p* = 0.50), (**E**) pitch dominated banked turn (*q*/*p* = 1.34). Line style of individual model components follows the legend at the bottom of panel (E). (**F**–**H**) The three passive yaw torque producing mechanisms at *t* = 0.4 s in panels (**D**–**E**) are flapping-counter-torque (**F**), torque due to forward motion with uneven left and right flapping frequencies (**G**), and torque due to sideways motion with non-zero dihedral angle (**H**). The legend for the color-coding of the flapping frequency and the positive directions of all the coordinates are shown at the bottom of panel (**H**).

Together with the high similarity between the turn dynamics of the robot and the fruit flies (Fig. S22), these results provide strong support for the hypothesis that fruit flies do not actively control yaw throughout evasive maneuvers (3, 5), but instead use the translation-induced coupled yaw torque (Eq. 1) to rotate their body in the direction of the banked turn. Given that many maneuvers of flying animals occur at non-zero translational velocities (*2–4*, *7*, *22–24*), the here-identified passive torque coupling might be more widespread among natural flyers.

Despite the observed yaw accelerations, high sideslips remained at the end of the turns, for both the robot and the flies (Figs. 2A–D, Fig. S17, S18, S22). In a separate set of robot experiments, we were able to completely remove this sideslip by producing feed-forward maximum-yaw torque throughout the recovery phase, but this body alignment did not increase the speed of the turn (Figs. S25, S26). Hence, producing such precisely-

timed and thus complex feedforward yaw command throughout the turn might not enable flies to increase evasive performance. This might be one of the reasons why flies prefer to rely on passive alignment throughout the turn, and fully align their body actively only after the turn, possibly using a simpler feedback controller based on visual information (*3*, *5*).

References and Notes

1. J. T. Vance, I. A. Faruque, J. S. Humbert, Kinematic strategies for mitigating gust perturbations in insects. *Bioinspir. Biomim.* **8**, 16004 (2013).
2. S. A. Combes, D. E. Rundle, J. M. Iwasaki, J. D. Crall, Linking biomechanics and ecology through predator-prey interactions: flight performance of dragonflies and their prey. *J. Exp. Biol.* **215**, 903–13 (2012).
3. F. T. Muijres *et al.*, Flies Evade Looming Targets by Executing Rapid Visually Directed Banked Turns. *Science* **344**, 172–177 (2014).
4. D. D. Chin, D. Lentink, Flapping wing aerodynamics: from insects to vertebrates. *J. Exp. Biol.* **219**, 920–932 (2016).
5. M. H. Dickinson, F. T. Muijres, The aerodynamics and control of free flight maneuvers in Drosophila. *Philos. Trans. R. Soc. B.* **371**, 20150388 (2016).
6. W. Shyy, C. Kang, P. Chirarattananon, S. Ravi, H. Liu, Aerodynamics, sensing and control of insect-scale flapping-wing flight. *Proc. R. Soc. A.* **472**, 20150712 (2016).
7. F. T. Muijres, M. J. Elzinga, N. a. Iwasaki, M. H. Dickinson, Body saccades of Drosophila consist of stereotyped banked turns. *J. Exp. Biol.* **218**, 864–875 (2015).
8. L. Ristroph *et al.*, in *Natural Locomotion in Fluids and on Surfaces*, S. Childress, A. Hosoi, W. W. Schultz, J. Wang, Eds. (Springer, New York, 2012), vol. 155 of *The IMA Volumes in Mathematics and its Applications*, pp. 83–99.
9. M. J. Elzinga, W. B. Dickson, M. H. Dickinson, The influence of sensory delay on the yaw dynamics of a flapping insect. *J. R. Soc. Interface*. **9**, 1685–1696 (2011).
10. M. Keennon, K. Klingebiel, H. Won, A. Andriukov, Development of the Nano Hummingbird: A Tailless Flapping Wing Micro Air Vehicle. *AIAA Pap.*, 1–24 (2012).
11. A. Roshanbin, H. Altartouri, M. Karásek, A. Preumont, COLIBRI: A hovering flapping twin-wing robot. *Int. J. Micro Air Veh.* **9** (2017).
12. A. Ramezani, S.-J. Chung, S. Hutchinson, A biomimetic robotic platform to study flight specializations of bats. *Sci. Robot.* **2**, 1–13 (2017).
13. H. V. Phan, T. Kang, H. C. Park, Design and stable flight of a 21g insect-like tailless flapping wing micro air vehicle with angular rates feedback control. *Bioinspir. Biomim.* **12**, 36006 (2017).
14. K. Y. Ma, P. Chirarattananon, S. B. Fuller, R. J. Wood, Controlled flight of a biologically inspired, insect-scale robot. *Science* **340**, 603–7 (2013).

15. N. Franceschini, F. Ruffier, J. Serres, A Bio-Inspired Flying Robot Sheds Light on Insect Piloting Abilities. *Curr. Biol.* **17**, 329–335 (2007).

16. M. Kovač, The Bioinspiration Design Paradigm: A Perspective for Soft Robotics. *Soft Robot.* **1**, 28–37 (2014).

17. G. C. H. E. de Croon, M. Percin, B. D. W. Remes, R. Ruijsink, C. De Wagter, *The DelFly - Design, Aerodynamics, and Artificial Intelligence of a Flapping Wing Robot* (Springer Netherlands, 2016).

18. Materials and methods are available as supplementary materials at the Science website.

19. M. Sun, J. Wang, Y. Xiong, Dynamic flight stability of hovering insects. *Acta Mech. Sin.* **23**, 231–246 (2007).

20. B. D. W. Remes *et al.*, in *IMAV* (2014), pp. 280–285.

21. S. Lupashin, A. Schöllig, M. Sherback, R. D'Andrea, in *International Conference on Robotics and Automation* (IEEE, 2010), pp. 1642–1648.

22. T. L. Hedrick, B. B. Cheng, X. Deng, Wingbeat time and the scaling of passive rotational damping in flapping flight. *Science* **324**, 252–255 (2009).

23. I. G. Ros, L. C. Bassman, M. a Badger, A. N. Pierson, A. A. Biewener, Pigeons steer like helicopters and generate down- and upstroke lift during low speed turns. *Proc. Natl. Acad. Sci.* (2011), doi:10.1073/pnas.1107519108.

24. K. Ghose, T. K. Horiuchi, P. S. Krishnaprasad, C. F. Moss, Echolocating bats use a nearly time-optimal strategy to intercept prey. *PLoS Biol.* **4**, 865–873 (2006).

25. B. Bruggeman, thesis, Delft University of Technology (2010).

26. BLHeli assembly code for sensorless brushless motor electronic speed control boards, available at https://github.com/bitdump/BLHeli.

27. http://wiki.paparazziuav.org/wiki/Main_Page.

28. M. Karásek, A. J. Koopmans, S. F. Armanini, B. D. W. Remes, G. C. H. E. de Croon, in *RSJ IROS* (IEEE, 2016).

29. Z. E. Teoh et al., in RSJ International Conference on Intelligent Robots and Systems (IEEE, 2012), pp. 3209–3216.

Acknowledgements: We greatly acknowledge the helpful comments on the manuscript from M.H. Dickinson, M. Kovac, J.L. van Leeuwen, H. Goosen and B. van Oudheusden. We also thank everyone involved in the earlier stages of the DelFly project, from which we greatly benefited, and especially S. Tijmons for the useful discussions on the robot design and K.M. Kajak for his work on flight dynamics modelling. **Funding:** FTM was supported by a grant from the Netherlands Organization for Scientific Research, NWO-VENI-863-14-007. **Author contributions:** All authors contributed to the conception of the comparative study and to the analysis and interpretation of the results; M.K. designed and built the robot, performed the robot experiments, and processed the robotic data; F.T.M. provided the fruit fly data; The manuscript was primarily written by M.K.,

F.T.M., and G.d.C. All authors contributed critically to the drafts and gave final approval for publication. **Competing interests:** MK is the inventor on patent application PCT/NL2018/050317 submitted by Delft University of Technology that covers the concept of the flapping wing robot. **Data and materials availability:** All the data relevant to the manuscript are deposited as open access data in Dataverse NL: https://hdl.handle.net/10411/ROXE7F.

Supplementary Materials

Materials and Methods

Supplementary Experimental Results

Figures S1–S26

Tables S1–S6

Movies S1–S10

References and notes (*25–29*)

Science
AAAS

Supplementary Materials for

A tailless aerial robotic flapper reveals that flies use torque coupling in rapid banked turns

Matěj Karásek, Florian T. Muijres, Christophe De Wagter, Bart D.W. Remes, Guido C.H.E. de Croon.

correspondence to: m.karasek@tudelft.nl, microuav@gmail.com

This PDF file includes:

Materials and Methods
Supplementary Experimental Results
Figs. S1 to S26
Tables S1 to S6
Captions for Movies S1 to S10

Other Supplementary Materials for this manuscript include the following:

Movies S1 to S10

Materials and Methods

Robot prototype

The robot (Fig. 1) is primarily built from off-the-shelf components, in order to facilitate production and maintenance at low costs. It reuses two custom-made injection-molded flapping mechanisms of the DelFly II MAV (*25*). Each flapping mechanism features a four-bar linkage mechanism, a 2-stage reduction gearbox with a gear ratio of 21.33 and a custom-made brushless DC motor (DC Enterprises, Bangalore, India). The motor is driven by a SuperMicro MX-3A electronic speed controller (ESC). The ESC was flashed with BLheli firmware v13.0 (*26*), customized to enable rpm sensing by sending a TTL pulse with every switch of the motor phases.

The wings are built of a 15 μm thick transparent boPET foil (Mylar), 0.28 mm round carbon fiber reinforced polymer (CFRP) rod stiffeners and 1.4 mm D shape CFRP rod leading edges. Their geometry was adopted from the design 'wing8436' in (*25*). The manufacturing process is similar to the 'advanced cut & glue method' in (*25*). The complete 28 cm wing is folded in the center to form the left/right wing pair of 14 cm length. The root of the wing pair is taped to the root bar made of 0.7 mm CFRP rod, which extends from the flapping mechanism frame.

The dihedral control mechanism parts are 3D printed using a UV curable resin 'Frosted Ultra Detail' from Shapeways, Inc. The hinge axes are made of 1 mm CFRP tubes. The mechanism is driven by a HK5330 rotary servo actuator, with its casing partially removed to save weight. 2 mm square CFRP tube is used for the connection between the control mechanism arms and the flapping mechanisms as well as the central fuselage.

The root bar deflection mechanism consists of a second HK5330 rotary servo actuator, attached to the central fuselage, and an arm made of 1 mm plywood rectangular plate. The wing root bars are inserted into the holes located on the sides of the plywood arm; the hole tolerances allow axial displacement of the bars when the arm is actuated.

The Lisa/S autopilot board features a 72MHz ARM Cortex-M3 microcomputer and an MPU6000 6-axis MEMS IMU (consisting of a 3 axis gyroscope and accelerometer) among other sensors (*20*). The autopilot is seated on a vibration isolator to prevent sensor saturation due to the high frequency vibration of the robot structure, excited by the flapping wings. The isolator is made of two 7 mm x 7 mm x 20 mm PU foam blocks and a 2mm Depron sheet base (Fig. S8C). The robot is further equipped with a Deltang Rx31 receiver (connected to the autopilot over a single wire PPM) and a 2 GB micro SD card for on board data logging (connected via SPI). Due to limited number of interfaces only one rpm signal from the ESC was connected to the Lisa/S autopilot.

The power is provided by a 180 mAh LiPo single cell battery (Hyperion CX G^3 25C). 28 AWG stranded copper hook-up wires bring power from the battery to the autopilot. The rest of the wiring (power, signals) is made of 0.18 mm enameled copper wires. The connection diagram of the robot's avionics is in Fig. S6.

Table S1 lists all the robot's components. The mass and inertial properties of the robot, estimated from the measured location and weight of individual components, are summarized in Table S2. At zero dihedral angle Γ (leading edges of left and right wing pairs being collinear), the center-of-mass (CoM) of the robot, lies within 1 mm from the

ideal yaw axis in Fig. 1J. The vertical position of the center-of-mass, z_{CoM}, is positioned approximately 50 mm below the plane of wing leading edges. The CoM shift at the maximal and minimal dihedral angle (±17°) remains within 5 mm of the nominal case and has a marginal effect (5% or less) on the body inertia (Table S2).

For all the presented experiments, the vehicle was equipped with reflective markers such that its position and orientation could be tracked by an optical motion tracking system, see Fig. S8. To assure good tracking under any conditions (even when the body is upside down), markers of approximately 12 mm in diameter were made using an IR reflective tape (OptiTrack, Inc.), pre-cut and folded into an approximate, hollow sphere, each weighting approximately 0.26 g. Three of these markers were placed on a structure added to the top of the vehicle, one was placed on the structure supporting the autopilot on the fuselage. Two additional flat circular markers were placed on top of the outside gears of the two flapping mechanisms such that also the dihedral angle Γ could be tracked. The tracking markers represented an extra payload of 1.61 g and increased the moments of inertia by about 12% (roll and yaw axis) and 17% (pitch axis). Despite an inevitable effect on the vehicle dynamics (lower agility), the same set of gains as for the 'unloaded' robot could be used without a noticeable decrease of performance.

On-board state estimation and control

The autopilot is running the open-source Paparazzi UAV software v5.11 (*27*), which includes modules for attitude estimation as well as for attitude stabilization.

For the attitude estimation, we employed the fixed-point implementation of attitude and heading reference system (AHRS) 'int_cmpl_quat'. This algorithm fuses the accelerometer and gyroscope data using complementary filters and estimates the attitude (in quaternion form) as well as biases of the gyroscopes. The attitude quaternion is used to avoid the gimbal lock of Euler angles. Nevertheless, the estimated attitude is also converted to Euler angles, which are used in the attitude stabilization. For the attitude stabilization, we employed the algorithm 'attitude_euler_int' of Paparazzi. In this fixed-point implementation, rotations around individual body axes are treated independently.

For roll and pitch stabilization we employed attitude feedback control with proportional (P) and derivative (D) terms. Even though natural fliers are believed to primarily stabilize their angular rates with a PI controller (on rates) (*5*, *8*), in the short term these two control types will result in comparable control inputs. The references were generated from the attitude setpoint using a second order system according to Fig. S7A. For the majority of the experiments presented here, the attitude setpoint was given by a preprogrammed sequence within the autopilot to assure consistency among multiple trials. This means that all the fruit-fly-inspired maneuvers were performed fully autonomously by the robot. Of course, the set points can also be commanded by the human pilot, which was the case in the hover to forward/sideways flight transitions (Figs. S9, S10). The angular position error is computed as the difference between the estimated attitude and the reference. The rate error is computed by subtracting directly the gyroscope reading from the rate reference. The controller schematics is in Fig. S7B. On top of the standard implementation of attitude stabilization in the Paparazzi UAV software, we added a low pass filter (2nd order Butterworth) to the controller output to reduce the amount of noise in the servo actuator commands. During the flight testing, we

found the 15 Hz cut-off to be a good compromise between noise reduction and the delay introduced to the control loop.

Yaw control was turned off during the fly-inspired rapid banked turns, but remained active otherwise to be able to steer the robot's heading in between the individual trials and to maintain straight flight just before the turn was triggered. In this axis, we were unable to use the bio-inspired D gain (i.e. P gain on rates) because of a combination of low load on the yaw actuator and noisy yaw rate measurements, due to a vibration around the yaw axis (torsion of the lightweight fuselage tube). Despite a stronger low pass filter used (10 Hz cut-off), the actuator would oscillate and get saturated. Thus, the (auto)pilot, giving a yaw rate setpoint, was directly commanding the deflection of the wing root mechanism through a feedforward term (Fig. S7C). Because of the inherent flapping-counter-torque (*22*), which is proportional to the yaw rate, the constant wing root deflection results in an equilibrium with a constant yaw rate (about 90° s^{-1} at maximal deflection). A P controller was employed to maintain the desired heading, computed by integrating the turn rate commanded by the pilot. All the parameters of the control loops and reference generators are summarized in Table S3.

The pitch command, cmd_{pitch}, and yaw command, cmd_{yaw}, were sent directly to the dihedral and wing root deflection mechanism servos, respectively. The theoretical dihedral deflection for ±100% pitch command was approximately ±17°. However, due to an inevitable play in the gear mechanism and an elastic deformation of the dihedral mechanism arms under the aerodynamic loads, larger dihedral angles were reached in forward flight, as evidenced by the motion tracking system data. The neutral dihedral angle was trimmed during the initial flights for moment equilibrium. The range of the wing root deflection servo arm was ±45° for ±100% yaw command.

The commands to the left and right motors, cmd_{motorL} and cmd_{motorR} were calculated from the roll and throttle commands as:

$$cmd_{motorL} = cmd_{throttle} - 0.5cmd_{roll} + 0.1\left|cmd_{roll}\right|, \quad \text{(S1)}$$

$$cmd_{motorR} = cmd_{throttle} + 0.5cmd_{roll} + 0.1\left|cmd_{roll}\right|. \quad \text{(S2)}$$

The last terms ($0.1\left|cmd_{roll}\right|$) were added to compensate for the loss of thrust due to body roll when flying sideways. Finally, the motor commands were bounded to remain between 0% and 100%.

Force balance setup and data processing

A Nano17 Titanium 6 DOF force/torque sensor (ATI-Industrial Applications, Inc., calibration SI-8-0.05) was employed to evaluate the performance of the robot in a static, hover-like (still-air) condition. An AIM-TTI Instruments PL155 power supply powered the robot instead of a battery to assure constant conditions during the measurements. Its remote sensing function was employed to compensate for the voltage drop on the supply cables; the voltage at the robot thus remained close to the preset value of 4 V.

The data was acquired using an NI cRIO-9024 controller, featuring a real-time processor and a field-programmable gate array (FPGA) allowing high acquisition rates. The controller was equipped with modules for input and conditioning of the 6DOF sensor bridge signals (2 x NI 9237), analogue input modules for acquisition of the voltage level

at the robot input (NI 9239) and the input current (NI 9227), and a digital input/output module (NI 9403) for pulse-width modulated (PWM) actuator command signals and rpm sensing pulses from the two ESCs. The sampling frequency was set to 10kHz during the static tests (constant commands) and 5kHz during the step response tests.

The acquired data was post-processed using MATLAB 2016a software (MathWorks, Inc.). The torque measurements were transformed to the CoM of the vehicle using its position in the sensor reference axes and the measured forces. In the static measurements (constant commands, Figs. S1-S5), we present the mean value over the 2-s measurement. All the results are discussed in the Supplementary experimental results section.

In the step response measurements (Fig. S3), the command was repeatedly switched between two preset values. The switching was done manually. The force and torque signals were low pass filtered with a 4th order zero-phase Butterworth filter with a 5 Hz cut-off. This allowed to filter out most of the variation due to flapping and due to the flapping-induced vibration yet to keep the low frequency content that was within the bandwidth of the actuators.

Free flight experiments

We performed 3 types of flight experiments with the robot, being:

1. Agility demonstration maneuvers: quick transitions from hover to forward/sideways flight and back to hover, and 360° roll and pitch flips
2. Repetitive transitions to forward/sideways flight to quantify the reproducibility of the maneuvers
3. Rapid banked turns inspired by the evasive maneuvers of fruit flies

Agility demonstration maneuvers

The experiments demonstrating the agility of the robot were performed in the "Open Jet Facility" wind tunnel room of the TU Delft, which provided sufficient space for these experiments (the wind tunnel itself was not used). We equipped the room with 10 Optitrack Prime 17W motion capture cameras, covering a tracking volume of approximately 5 m x 6 m x 4 m and recording the position and attitude of the robot equipped with the reflective markers (Fig. S8) at 360 frames-per-second (fps).

In parallel to the motion tracking, data were logged on-board of the robot with a sampling rate of approximately 100 Hz. These included the attitude estimates (Euler roll-pitch-yaw angles $[\phi_{IMU}, \theta_{IMU}, \psi_{IMU}]$), angular rates $\boldsymbol{\omega}_{\mathbf{IMU}} = [p_{IMU}, q_{IMU}, r_{IMU}]$, commands ($cmd_{throttle}$, cmd_{roll}, cmd_{pitch}, cmd_{yaw}), attitude setpoints $[\phi_{set}, \theta_{set}, \psi_{set}]$ and the rpm of the right flapping mechanism motor.

A) Rapid transitions from hover to fast forward/sideways flight and back

Unlike in all the other experiments, the vehicle was piloted by a human operator during these tests. The pilot was giving the attitude set-point through the sticks of a remote controller. The test sequence consisted of hovering (zero pitch and roll commands), transition to fast forward or sideways flight (full pitch or roll command, respectively) and return to hover (zero pitch and roll commands).

B) Rapid 360° roll and pitch up flips: barrel rolls and loopings

These maneuvers consisted of a 5-phase program, triggered by the pilot when the robot is hovering:

a. Gaining altitude by commanding 94% throttle for a duration of 0.6 s while still in attitude feedback mode.
b. Initiation of the flip by commanding a full positive roll/pitch torque in open loop (OL) mode to initiate the maneuver until a limit angle of $\phi_{STOP}/\theta_{STOP}$ = 90°, estimated by integrating the gyro signal of the respective axis, is reached; the throttle command is nominal.
c. 'Free rotation' phase with 26% throttle command and zero roll and pitch torque commands until $\phi_{START}/\theta_{START}$ = 270°, estimated by the single axis gyro integration, is reached.
d. Recovery by switching back to attitude feedback mode with hover attitude setpoint, with a 94% throttle command for a duration of 0.8 s to stop the fall around the initial altitude.
e. Return to hover (attitude feedback mode) with nominal throttle.

The timing and other parameters of the program were hand tuned for reliable performance. The non-zero throttle command during the "free rotation" phase was necessary to prevent the brushless motors from stopping as their spin-up time from a full stop is much longer.

Reproducibility tests

The tests of maneuver reproducibility were performed in the "Cyberzoo" flight arena of the TU Delft, which was equipped with 12 OptiTrack Prime 17W motion capture cameras, giving a tracking volume of approximately 9 m x 9 m x 4 m. Again, this was complemented by the on-board logged data.

We performed repetitive transitions from hover to various pitch setpoints (forward flight tests) and roll setpoints (sideways flight test). A perfect step in the setpoint was achieved by linking the setpoint value to a mechanical switch of the remote control that was triggered by the pilot. For forward flight maneuvers, we performed pitch steps from 0° to 15°, 30°, 45°, 60° and 75° (Fig. S11), while the roll and yaw setpoints remained 0° and the thrust command remained constant. Similarly, roll steps to 15°, 30°, 45° and 60° were performed for the sideways maneuvers (Fig. S12). Each maneuver was repeated at least five times.

Rapid banked turns inspired by the evasive maneuvers of fruit flies

The fly-inspired experiments were also performed in the "Cyberzoo" flight arena of the TU Delft. Because the motion tracking system was undergoing an upgrade over the course of these experiments, the cameras and settings used differ for individual datasets. The test in Fig. 2E was recorded at 56 fps with 24 Optitrack Flex13 motion capture cameras. The tests in Fig. 2A,C,F–I and Fig. 3 were recorded at 120 fps with 10 Optitrack Prime 17W motion capture cameras. Finally, the tests in Fig. S25 were recorded at 360 fps with 12 Optitrack Prime 17W motion capture. All the tests and the respective camera setups are also listed in Table 4. Again, additional data was logged on-board.

The autonomous, insect-inspired, rapid banked turns were triggered by the pilot when the robot was hovering and consisted of 4 phases:

a. Acceleration to an initial velocity U_{IN} by commanding pitch θ_{IN} in attitude feedback mode.
b. Maneuver initiation by constant feedforward commands to the actuators in open loop (OL) mode for a duration of t_{OL}.
c. Recovery from the OL phase perturbation by commanding pitch θ_{OUT} in attitude feedback mode (roll & pitch) and a feedforward command $cmd_{yaw,OUT}$ in yaw.
d. Return to hover.

In the systematic tests (Figs. 3,4), 5 to 6 maneuvers were recorded for each combination of commands, resulting in different *q/p* at the end of the OL phase. Each turn was preceded by a body pitch command θ_{IN} = -30° given at t = -1 s in order to accelerate to a velocity $U_{\mathrm{IN}} \approx 1.2$ m s^{-1} at t = 0 s (start of OL phase). The duration of the OL phase (t_{OL} = 0.25 s ≈ 4 wingbeats) was tuned such that roll angles would remain below 90° even for the largest roll commands in these trials. Because the robot had a tendency to accelerate during the less sharp turns, in accordance with fruit flies (*3*), a smaller pitch angle command θ_{OUT} = -15° was used during the recovery phase to guarantee successful recovery within the bounds of the flight arena. The feedforward yaw command $cmd_{yaw,OUT}$ was set to zero in all the tests, except for the trials with feedforward yaw control included (Fig. S25C,D), where the maximal yaw actuator deflection ($cmd_{yaw,OUT}$ = 100%) was commanded.

Processing and analysis of the flight data

All the processing was carried out using MATLAB 2016a software (MathWorks, Inc.). The motion tracking data consists of position vector ${}^{\mathbf{G}}\mathbf{x} = [x, y, z]$ and orientation quaternion $\mathbf{q}$ of the body within the world reference frame, as well as the position of all the individually tracked markers ${}^{\mathbf{G}}\mathbf{x}_n$, with n being the marker number. The orientation quaternion was transformed to the Euler Roll-Pitch-Yaw angles $[\phi, \theta, \psi]$ following the standard aerospace formulas. The tracking quality was monitored by the marker error parameter of OptiTrack Motive 1.10.3 software (NaturalPoint, Inc.); the mean values and standard deviations of the marker error of the recordings used in the analysis are listed in Table S4. Individual frames with a mean marker error larger than 10 mm were considered to be outliers. Short segments (up to 10 frames) of outliers or gaps in the data due to an occasional loss of tracking were filled with linear interpolation (*interp1(… ,… ,'linear')* function in MATLAB). Experiments with longer outlier segments or gaps, were discarded.

The data was corrected for the misalignment between the body axes (with an origin at the CoM) and OptiTrack-defined axes like in (*28*). The body CoM velocity ${}^{\mathbf{G}}\mathbf{v} = [v_x, v_y, v_z]$ and accelerations ${}^{\mathbf{G}}\mathbf{a} = [a_x, a_y, a_z]$ were computed by numerical differentiation of the position vector (in ground fixed axes). By transforming these data using the Euler angles, we determined the velocity and acceleration of the robot in the body reference frame (${}^{\mathbf{B}}\mathbf{v} = [u, v, w]$ and ${}^{\mathbf{B}}\mathbf{a}$, respectively**)**. Similarly, angular rates $\boldsymbol{\omega} = [p, q, r]$ and angular accelerations $\boldsymbol{\alpha} = [\dot{p}, \dot{q}, \dot{r}]$ were computed by numerical differentiation, starting from the Euler angles and doing the necessary kinematic transformations.

The flight speed of the robot, U, was taken as the norm of the body velocity vector ${}^{\mathbf{B}}\mathbf{v}$, and the course is the direction of its horizontal components of ${}^{\mathbf{B}}\mathbf{v}$ (filtered, as

explained further). Turn rate (rate of change of course) was found by numerical differentiation of course over time. Sideslip angle β was computed as

$$\beta = \arcsin(v / U)\,, \tag{S3}$$

whereby v is the lateral component of the body velocity vector ${}^{\mathbf{B}}\mathbf{v}$.

The dihedral angle of the wings Γ was computed from the position of the individual markers, transformed into the body frame, using basic trigonometry, similar to the deflection of the control surfaces in (*28*).

Finally, a zero-phase low pass filter (Butterworth, 8th order, 10 Hz cut-off) was applied to the linear and angular velocities and accelerations to reduce the amount of noise, amplified by the numerical differentiation. When generating the animation sequences as in Fig. 2, the same filter was also applied to the dihedral angle and the commands to avoid aliasing effects due to low 'sampling rates' of the sequence.

Cross-correlation was used to find the time difference Δt between the motion tracking and on-board datasets. We used the (estimates of) body pitch and roll, the two signals contained in both datasets, which were pre-filtered using the filter described above (identical for both datasets) and interpolated to 1kHz using spline interpolation (*interp1(…, …, 'spline')* function in MATLAB). The time difference was computed from the relative signal displacement giving a maximum value of cross-correlation (*xcorr(…)* function in MATLAB). An average of the two values found for roll and pitch was used to synchronize the datasets.

In the rapid banked turns, the data was aligned by zeroing the position, course and body yaw when the turning maneuver was triggered (t = 0 s). The data presented in all the flight experiment figures is based on the motion tracking data, unless stated otherwise, except for the commands and the right-wing flapping frequency. Apart from time measured in seconds, a secondary abscissa is shown at the top of some figures representing the number of wingbeats. Because the flapping frequency is not constant and differs significantly between the left and right wing-pairs when roll torque is commanded, the mean flapping frequency at hover (f = 17 Hz) was used.

Comparison with fruit fly data

To be able to perform a quantitative comparison of the robot and fruit fly flight data, scaling was used to account for the large differences in size and wing kinematics. We used root-to-tip wing length L_W as a scaling factor of length (L_{Wrobot} = 140 mm, L_{Wfly} = 3.0 mm). For time scaling, a good match was achieved when using wingbeat cycle time as a scaling factor τ for fruit flies ($\tau_{fly} = 1/f_{fly} = 5.3\ ms$) and half-wingbeat cycle time for the robot ($\tau_{robot} = 1/(2f_{robot}) = 29\ ms$). While this could be due to the fact that the robot has twice the number of wings than the flies do, it is more likely that it is just a coincidence, and the factor of 2 captures also the differences in flapping amplitudes, wing aspect ratios, etc. Time alignment was achieved by searching for maximum of the cross-correlation function between the roll and pitch rates in the two datasets. The procedure was equivalent to what was described in the previous section for synchronization of the motion tracking and on-board data.

The data in both systems (robot and fruit flies) is represented in a body-fixed stroke plane reference system, where the xy plane is aligned with the wing stroke plane. The

robot needs to pitch forward more to reach the same scaled speed, which may be due to different relative position of the center of mass and the center of pressure of the wings , but also because the robots wingbeat kinematics is nearly symmetric, unlike in flies where differences between downstroke and upstroke are present. Thus, the body pitch of the robot was offset from that of the fly throughout the turn. Nevertheless, the angular rates along the same, stroke-plane-fixed, axes as well as the velocities in the stroke plane direction, *u* and *v*, followed the same trends.

In Fig. S17 and Fig. S18 we compared the scaled data for individual turns shown in Fig. 2A,B and Fig. 2C,D. To quantify the similarity, we used Pearson's correlation coefficient c_r and root-mean-squared error *rmse*. The *rmse* was normalized by the maximal absolute value of each variable in the robot data.

Qualitative comparison was carried out for the average rapid banked turns for the various *q*/*p* in Fig. 3, which were compared to the fruit fly maneuvers from (*3*), binned according to the direction of the looming target into four groups and averaged.

Yaw dynamics model

High yaw rates (and accelerations) were present in the recovery phase of the maneuver both in the robot (Fig. 3D) as well as in the fruit-fly experiment (*3*). Because the yaw command was set to zero throughout the robotic experiment, these yaw accelerations must be passively induced. In an effort to explain these, we constructed a simple aerodynamic model, which was built around the flapping-counter-torque and flapping-counter-force concepts (*22*).

Assuming a 2D case, making the quasi-steady assumption, and assuming only the translational force component (*4*, *5*), the flap averaged wing drag force can be expressed as

$$\bar{D} = \frac{1}{T}\int_0^T \frac{1}{2}\rho C_D(\alpha(t)) S U(t)^2 \mathrm{dt}, \qquad \text{(S5)}$$

where $T = 1/f$ is the wingbeat period, ρ the air density, C_D the drag coefficient at the angle of attack α, S the wing area, U the wing air speed and t time.

We assume further that the body is moving with a constant forward speed U_{body} and that the wing is flapped back and forth with a constant speed and a constant angle of attack throughout each half stroke (i.e. both the speed due to flapping as well as the angle of attack follow a rectangle function). The speed due to flapping can be expressed as

$$U_{flap} = 2\Phi R f, \qquad \text{(S6)}$$

where Φ is the flapping amplitude, R is the distance from the flapping hinge point to the wing's center of pressure and f the flapping frequency. If the flapping speed is higher than the body speed U_{body}, i.e. $U_{flap} > U_{body}$ (Fig. S23A), the integration in (S5) can be under the above assumptions rewritten as

$$
\begin{aligned}
\bar{D} &= \frac{1}{2}\bar{D}_{down} + \frac{1}{2}\bar{D}_{up} = \\
&= \frac{1}{4}\rho C_D(\alpha)S(U_{flap}+U_{body})^2 - \frac{1}{4}\rho C_D(\alpha)S(U_{flap}-U_{body})^2 = \\
&= \rho C_D(\alpha)SU_{flap}U_{body} = \\
&= 2\rho C_D(\alpha)S\Phi RfU_{body} = \\
&= b\Phi fU_{body}.
\end{aligned} \tag{S7}
$$

Thus, the drag due to body motion can be represented by a linear damping force, proportional to the body speed U_{body}, but also to the flapping amplitude Φ and flapping frequency f, where the parameter b depends on the wing geometry and kinematics. While many assumptions that were made do not hold for the real robot with a pair of highly flexible wings following rather a harmonic motion, this model was shown to represent well the forces of an insect-scale robot measured in the wind tunnel in the presence of wind (*29*), and could even be used to develop a longitudinal dynamics model that could represent the dominant oscillating motion of a hovering hummingbird-inspired robot (*11*).

Here we generalize this concept to the problem of body yaw dynamics coupled with motion in the horizontal body plane. Each wing pair is represented, in the wings-closed position, by a point at the distance R from the wing hinges (Fig. S23B). The normal component of the speed due to body motion can be expressed for the left (subscript $_L$) and right (subscript $_R$) wing pair as

$$
\begin{aligned}
U_L &= u + v\Gamma + rR, \\
U_R &= u - v\Gamma - rR.
\end{aligned} \tag{S8}
$$

where Γ is the dihedral angle, u and v the body velocity in the longitudinal and lateral body direction and r is the angular rate around the yaw axis of the body. Small angle approximation of the trigonometric functions was used since the dihedral angle Γ typically remains below 25°. Substituting the speed in (S8) for U_{body} in (S7), the drag forces of the individual wing pairs become

$$
\begin{aligned}
D_L &= b\Phi_L f_L U_L = b\Phi_L f_L(u + v\Gamma + rR), \\
D_R &= b\Phi_R f_R U_R = b\Phi_R f_R(u - v\Gamma - rR).
\end{aligned} \tag{S9}
$$

Finally, the yaw torque acting on the body can be written as

$$
\begin{aligned}
N &= R(D_R - D_L) = \\
&= R\left[b\Phi_R f_R(u - v\Gamma - rR) - b\Phi_L f_L(u + v\Gamma + rR)\right] \\
&= -bR\left[(f_R\Phi_R + f_L\Phi_L)(Rr + \Gamma v) + (f_L\Phi_L - f_R\Phi_R)u\right].
\end{aligned} \tag{S10}
$$

The torque can be decomposed into three components

$$N = N_{FCT} + N_{u\&roll} + N_{v\&roll}, \tag{S11}$$

where N_{FCT} is the well-known flapping-counter-torque (*22*)

$$N_{FCT} = -bR^2(f_R\Phi_R + f_L\Phi_L)r, \tag{S12}$$

$N_{u\&roll}$ is a torque due to coupling of the roll torque generation mechanism (via flapping frequency and/or flapping amplitude difference) with the forward body motion u

$$N_{u\&roll} = -bR(f_L\Phi_L - f_R\Phi_R)u, \tag{S13}$$

and $N_{v\&pitch}$ is a torque due to coupling of the pitch torque generation mechanism (via dihedral angle, sometimes also called the mean stroke angle) with the sideways body motion v

$$N_{v\&pitch} = -bR(f_R\Phi_R + f_L\Phi_L)\Gamma v. \tag{S14}$$

We term the newly discovered torque due to coupling of roll and pitch torque generation mechanisms with the body translation the "translation-induced coupled yaw torque"

$$N_{TCT} = N - N_{FCT} = N_{u\&roll} + N_{v\&pitch}. \tag{S15}$$

Due to autopilot limitations, we could only log the flapping frequency of the right wing pair. However, we were able to estimate the flapping frequency of both wings from the motor commands. The relationship between the motor command and the (steady state) flapping frequency f_{SS} was estimated from the linear fits of force balance measurements in Fig. S1 as

$$f_{SS} = 0.20cmd_{motor} + 3.7c_{batt} - c_v v - c_w w, \tag{S16}$$

where c_{batt} is a correction factor added to account for the battery voltage drop as the on-board battery discharges, and the c_v and c_w correction factors were added to account for the drop of flapping frequency observed in the presence of lateral and vertical body velocity, u and w, respectively. To further account for the actuator dynamics, a first order transfer function was identified from the force balance data in Fig. S3M with a help of MATLAB 2016a System Identification toolbox (MathWorks, Inc.)

$$\frac{f(s)}{f_{SS}(s)} = \frac{12.56}{s + 12.56}. \tag{S17}$$

The values of the correction factors giving the closest predictions were $c_v = 0.5\ \mathrm{m}^{-1}$ and $c_w = 0.5\ \mathrm{m}^{-1}$. The battery charge correction factor, with values between 0 and 1, was tuned for each trial individually. A comparison of the measured and model predicted flapping frequencies for the experiments reported in Fig. 2 is in Fig. S24A–E.

Using the predicted flapping frequencies, we could estimate the yaw accelerations in the series of experiments in Fig. 2. Since the flapping amplitude of the robot is always constant ($\Phi = \Phi_L = \Phi_R = 44°$), its yaw acceleration can be predicted as

$$\dot{r} = \frac{N}{I_{zz}} = -\frac{b\Phi R}{I_{zz}}\left[(f_R + f_L)Rr + (f_L - f_R)u + (f_R + f_L)\Gamma v\right]. \tag{S18}$$

where I_{zz} is the moment of inertia around the yaw axis. We assumed the drag force to be acting in the middle of the wing in the spanwise direction. Including half of the body width, we used a value of R = 98.4 mm The damping coefficient b was tuned to minimize the error between the predicted and measured yaw accelerations, the value of 2.7e-3 $N.s^2.m^{-1}$ was then used for all the trials. Comparison of the measured and predicted yaw accelerations (Fig. S24F–J), shows that the model captures the dominant effects well, which is also documented by the (in most cases) high correlation coefficients (Table S6).

The model was not directly applicable to the fruit-fly maneuvers due to the high complexity of the observed wing motion patterns. However, we could estimate the yaw acceleration due to the translation-induced coupled yaw torque as

$$\dot{r}_{TCT} = \dot{r} - \frac{N_{FCT}}{I_{zz}} \tag{S19}$$

to evaluate the correlation of this flapping-counter-torque corrected yaw acceleration to the product of roll acceleration $\dot{p}$ (effect of roll torque production) with forward velocity u and the product of pitch acceleration $\dot{q}$ (effect of pitch torque production) with sideways velocity v (Fig. 4C).

Supplementary Experimental Results

Force and torque generation performance

The force balance tests showed that the robot's thrust scales approximately linearly with the throttle command as well as with flapping frequency (Fig. S1A,B). The robot's operating point is close to the maximal thrust to power ratio, showing a good match of the brushless motor and the load characteristic of the flapping wings (Fig. S1C).

The generated roll and pitch torques are almost completely decoupled and remain approximately linear functions of the torque commands (Fig. S2, S4, S5), per design. The standard deviations of the yaw torque are higher due to noise coming from a marginal structural vibration, which was present in this axis when the robot was clamped. In all the measurements, the thrust remained within ±5% from the nominal case value (zero attitude commands).

One of the advantages of the presented design is that torque production is independent of the vertical center-of-mass (CoM) location. We specifically designed for this independence, as in lightweight drones, small variations of heavier components (such as the replaceable battery) can alter the CoM location. Our control torque generation concept allows the CoM location, a parameter having a significant influence on the robot's flight dynamics (*11*), to be chosen almost completely freely. This insensitivity is shown by means of experimental results in Fig. S5E,F, transformed to three (significantly) different CoM locations: the nominal CoM, 50 mm above, and 50 mm below the nominal CoM. The slight slope variation observed is caused by an imperfect alignment of the flapping mechanisms and the wings, which resulted in non-zero, command varying, horizontal forces, which in turn influenced the roll and pitch torques after the transformation.

The bandwidth of the control actuation is characterized in Fig. S3, where step inputs are commanded, individually, in the three torque axes. The rise times were all below 0.2 s, which is equivalent to ~3.4 wingbeats at the nominal flapping frequency.

Free flight performance

Our set of free flight experiments demonstrates the agility and control of our robot (Figs. S9-S19).

Rapid transition from hover to fast forward and sideways flight and back

The rapid transitions from hover to forward / sideways flight (and vice-versa), initiated by pitching / rolling are shown in Fig. S9 and Fig. S10, respectively. During the forward flight transition (Fig. S9), angular acceleration of up to 3000° s^{-1} is reached, allowing the vehicle to change its attitude from hover (0° pitch) to fast forward orientation (60° pitch) or vice versa in less than 5 wingbeats. Despite no changes in the throttle command, the vehicle quickly accelerates (up to 0.8 g in the early phase, where g is gravitational acceleration) reaching a levelled flight speed of 4 m/s in approximately 1 second. The high deceleration during the transition back to hover (-1.3 g in x direction) is due to a particularly attractive property of flapping wing propulsion, the high aerodynamic damping (*22*). When commanded a zero attitude, the robot will coast to a stop, while stopping a conventional rotary-wing robot requires well-timed tilting in the opposite

direction to the robot's motion. Comparable performance was observed also in the lateral direction (Fig. S10).

For these maneuvers, we also present the on-board attitude estimates, angular rates and accelerations (Figs. S9G–I, S10D–F), which are in a good agreement with the off-board motion tracking data.

Reproducibility tests

The repetitive tests of steps in pitch (Fig. S11) and roll (Fig. S12) demonstrate that the robot maneuvers are highly reproducible, not only in the commanded attitudes, but also in the resultant body rates, velocities and accelerations, as is demonstrated by the low standard deviations of the captured data, represented by the shaded error bars. The recording length was limited by the size of the tracking volume, therefore steady state could not be reached for the larger setpoints.

Since the attitude control does not use an integrator term, steady state errors are present, especially in the sideways flight test. Flying sideways is limited to approximately 4 m/s, when the vehicle does not manage to maintain a straight flight anymore and will start yawing due to coupling with the pitch stabilization that uses the adjustable dihedral angle. Forward flight speed is only limited by the power of the vehicle. The roll and yaw "error" appearing around $t = 0.4$ s for the highest pitch set point is caused by the singularity in the Euler angle description after transformation from the attitude quaternions. The roll and yaw angular rates show that the vehicle does not rotate along these axes in reality.

Rapid 360° roll and pitch up flips: barrel rolls and loopings

High agility as well as ability to recover from large disturbances is demonstrated by 360° roll and pitch flip maneuvers (Figs. S13, S14). In these maneuvers, the robot reaches angular rates of 1000° s^{-1} and 800° s^{-1} in roll and pitch, respectively, and angular accelerations of approximately 5000° s^{-2}, both in pitch and in roll. The 360° maneuvers are completed within 0.7s, which is about 11 wingbeats.

Rapid banked turn inspired by the evasive maneuvers of fruit flies

The complete flight data for all the maneuvers in Figs. 2–4, S25 are presented in Figs. S15, S16, S19, S21, S26. Considering the short duration of the OL phase (~4 wingbeats) and random wingbeat cycle phases of the individual wing-pairs at its start, the variance of time histories of individual trials in the systematic tests (Figs. 3B–D, 4E–H, S21, S26) is low and shows a high level of repeatability of these experiments, especially in the rotational movements (attitude, angular rates, accelerations). Moreover, despite the extreme conditions reached while transitioning from forward flight to the high-bank turn, the angular rates at the end of the OL phase remained approximately proportional to the commands (Fig. S20). The slightly higher variance of the flight trajectories (Fig. 3A) could partly be caused by the different initial conditions due to parameters that could not be controlled directly (wingbeat phases of the left and right wing pair, velocity) and partly by the nonlinearities experienced during the recovery phase.

Supplementary Figures

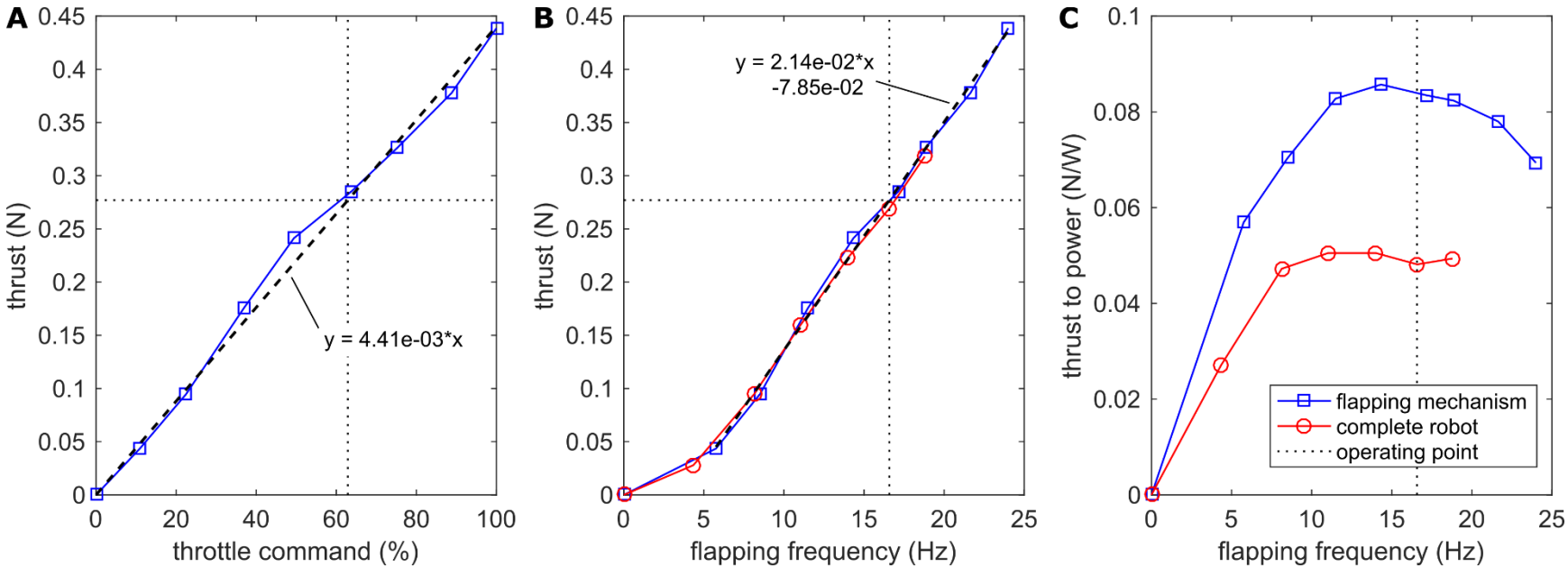

Fig. S1.

Thrust measurement: (**A**) Thrust versus throttle command and (**B**) versus flapping frequency. Both relationships follow a linear trend (black dashed line) with respective R-squared values of 0.999 and 0.997. (**C**) Lift per electrical power versus flapping frequency. The measurements were carried out with a single flapping mechanism driven by the ESC (blue, multiplied by 2) and with the complete robot with two such mechanisms, the autopilot and the servo-actuators (red line). The dotted line shows the operating point (robot without any payload).

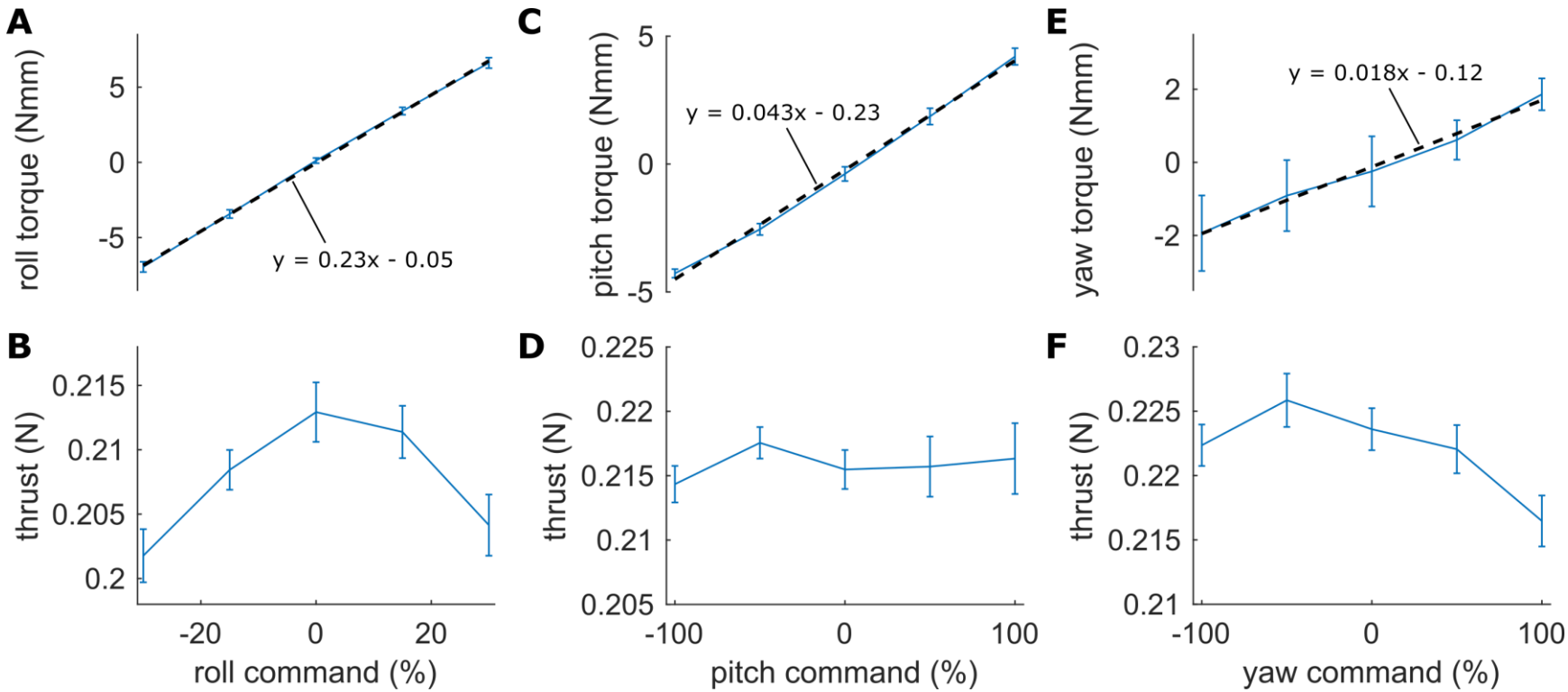

Fig. S2.

Torque measurements: (**A**) Roll torque versus roll command; (**C**) pitch torque versus pitch command; (**E**) yaw torque versus yaw command. (**B**, **D**, **F**) Thrust measurements corresponding to the torque measurements in (**A**, **C**, **E**), respectively. The blue lines are averages of 4 measurements, the error bars indicate the standard deviations, each black dashed line is a linear fit according to the shown formula.

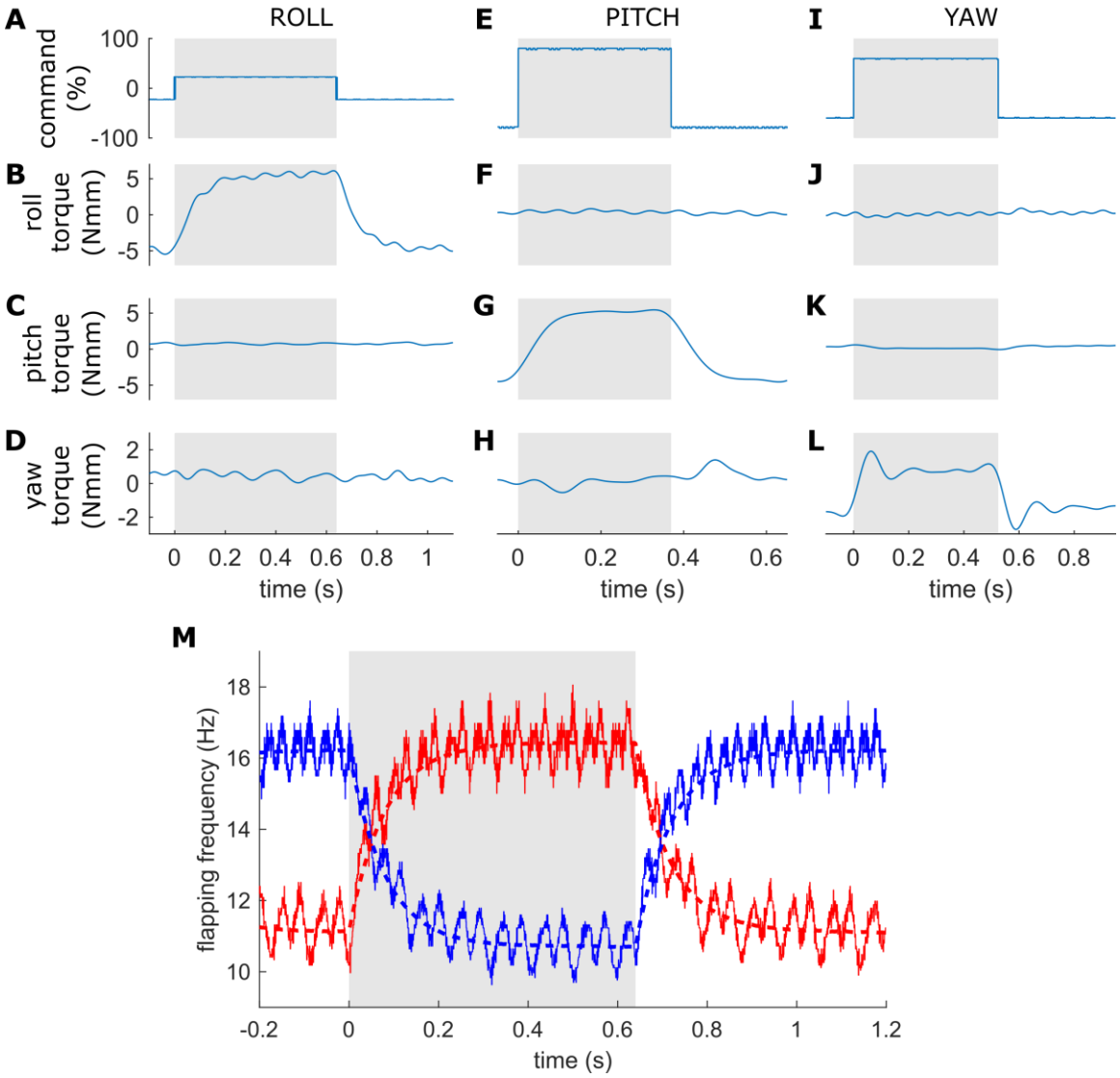

Fig. S3.

Measurements of torque command steps performed at 50% throttle command. (**A–D**) Roll torque steps measurements; (**E–H**) Pitch torque steps measurements; (**I–L**) Yaw torque steps measurements. (**A**) Roll command; (**E**) Pitch command; (**I**) Yaw command; (**B**, **F**, **J**) Roll torque; (**C**, **G**, **K**) Pitch torque; (**D**, **H**, **L**) Yaw torque. The torque data were low-pass filtered with a 4th order Butterworth filter, 5Hz cut-off. (**M**) Measurements (solid) of flapping frequency and their prediction (dashed) by the identified actuator dynamics during the roll torque steps in (B–J); Left and right wing pairs are plotted in red and blue, respectively.

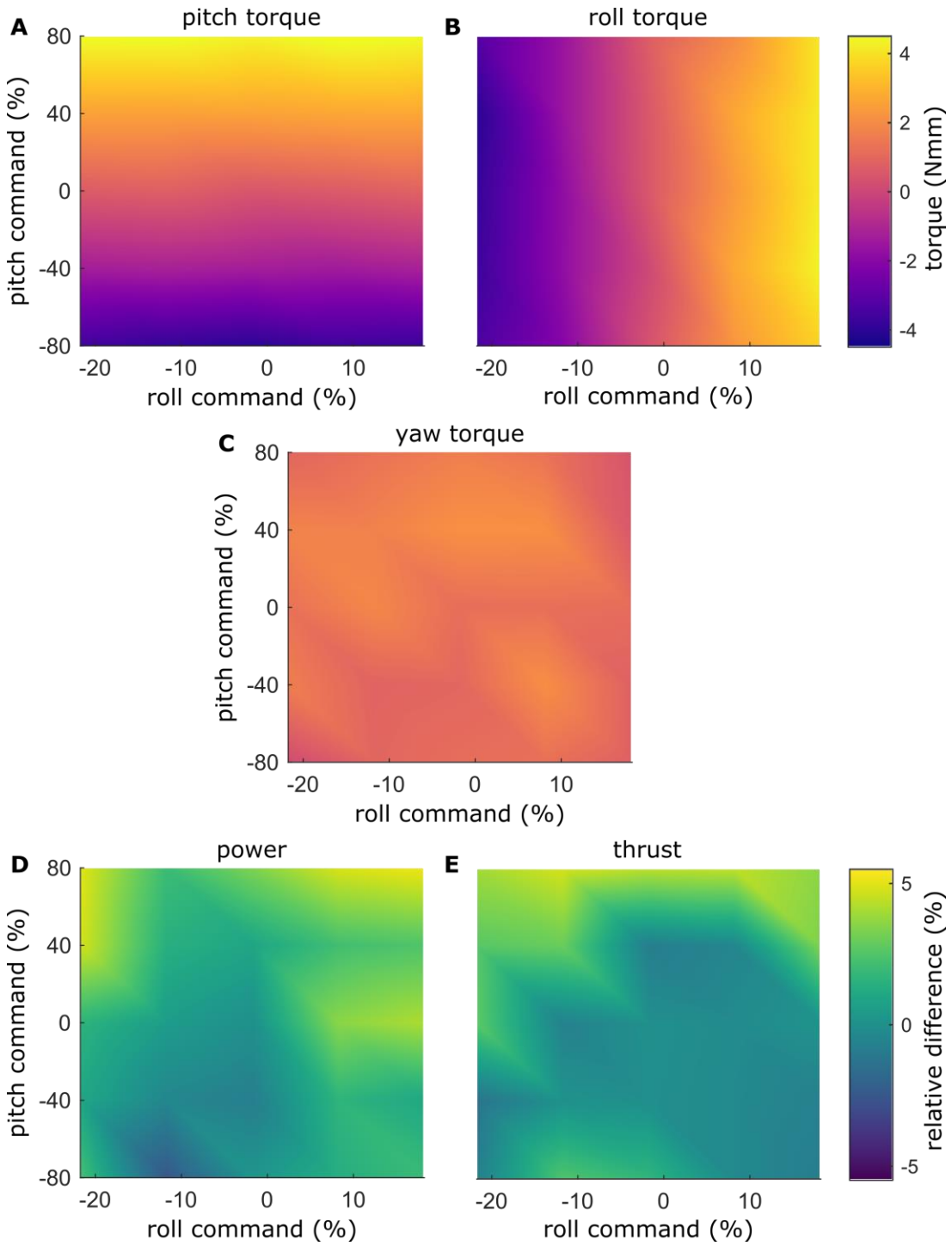

Fig. S4.

Measurements of combined pitch and roll commands performed at 50% throttle command and 0% yaw command. (**A**) Roll torque; (**B**) Pitch torque; (**C**) Yaw torque; (**D**) Relative electrical input power difference from the nominal value (3.91 W); (**E**) Relative thrust difference from the nominal value (0.219 N).

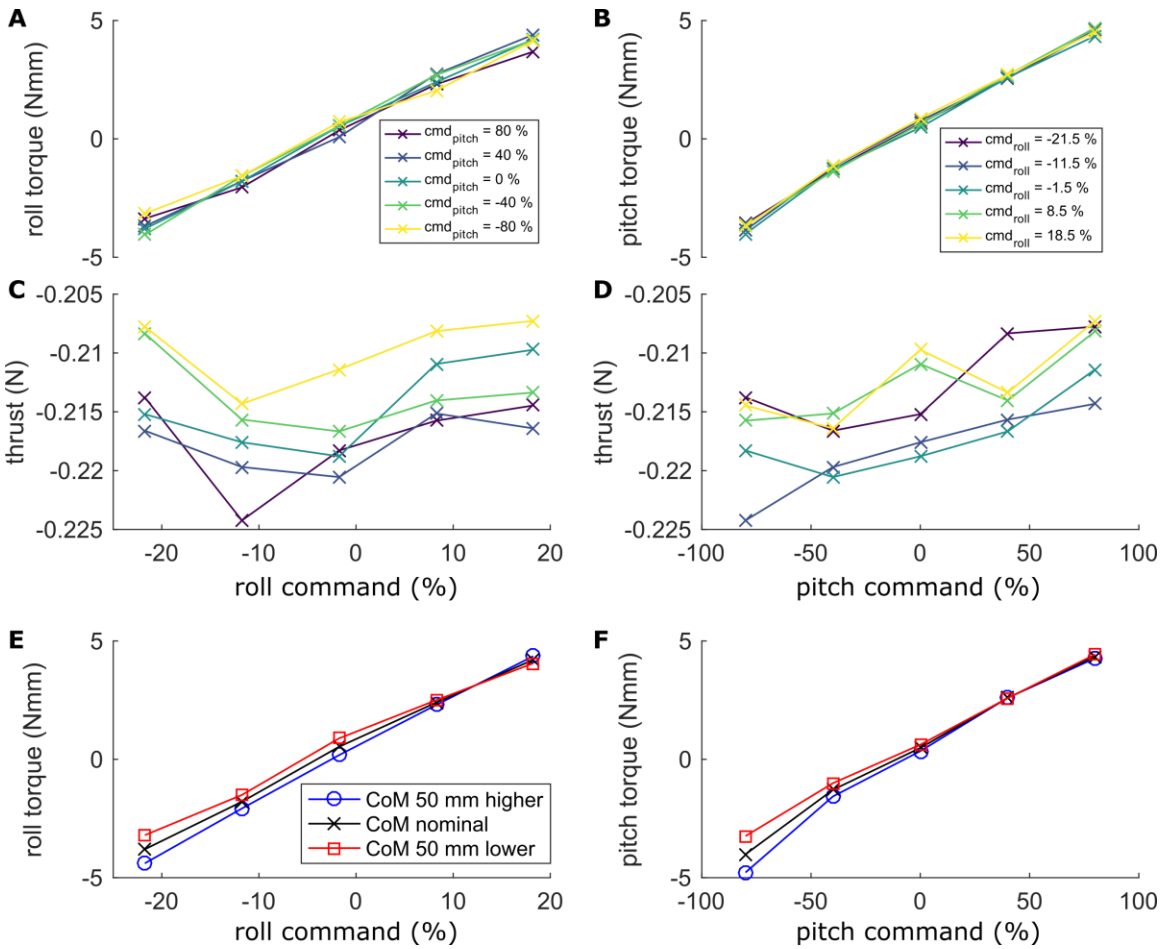

Fig. S5.

Measurements of pitch and roll torques performed at 50% throttle command. (**A**) Roll torque against roll command for various values of pitch command, color coded according to the legend. (**B**) Pitch torque against pitch command for various values of roll command, color coded according to the legend. (**C**, **D**) Thrust measurements corresponding to (**A, B**), respectively. (**E**) Roll torques for three different CoM locations along the main vertical fuselage (nominal, 50 mm higher, and 50 mm lower), color coded according to the legend. (**F**) Pitch torque for three different CoM locations along the main vertical fuselage, the same color coding as in (**E**).

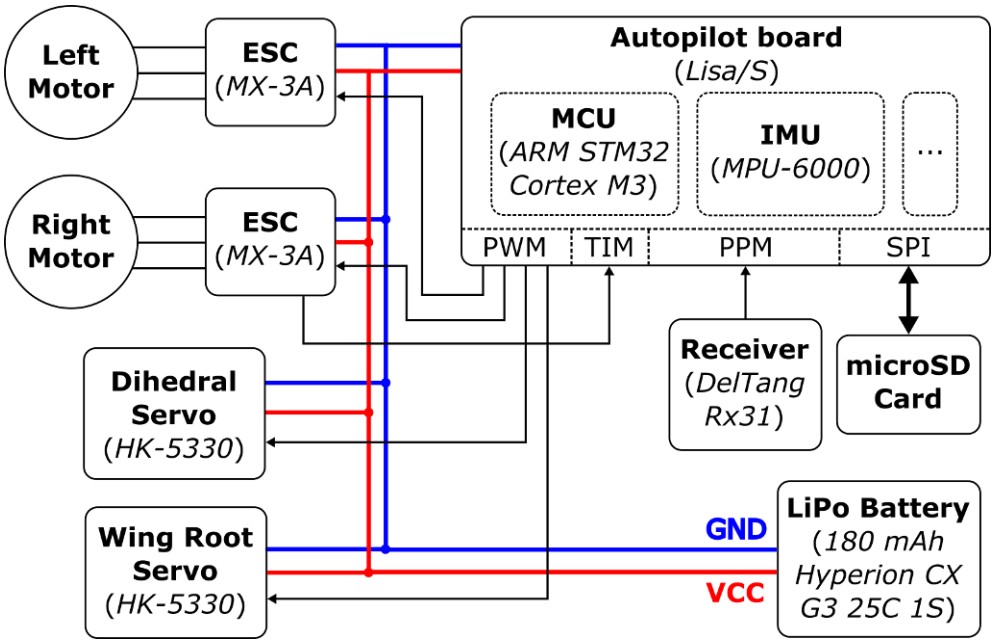

Fig. S6.

Connection diagram of the on-board electronics (GND = ground, VCC = positive supply voltage, PWM = pulse width modulation, TIM = timer, PPM = Pulse Position Modulation, SPI = Serial Peripheral Interface bus).

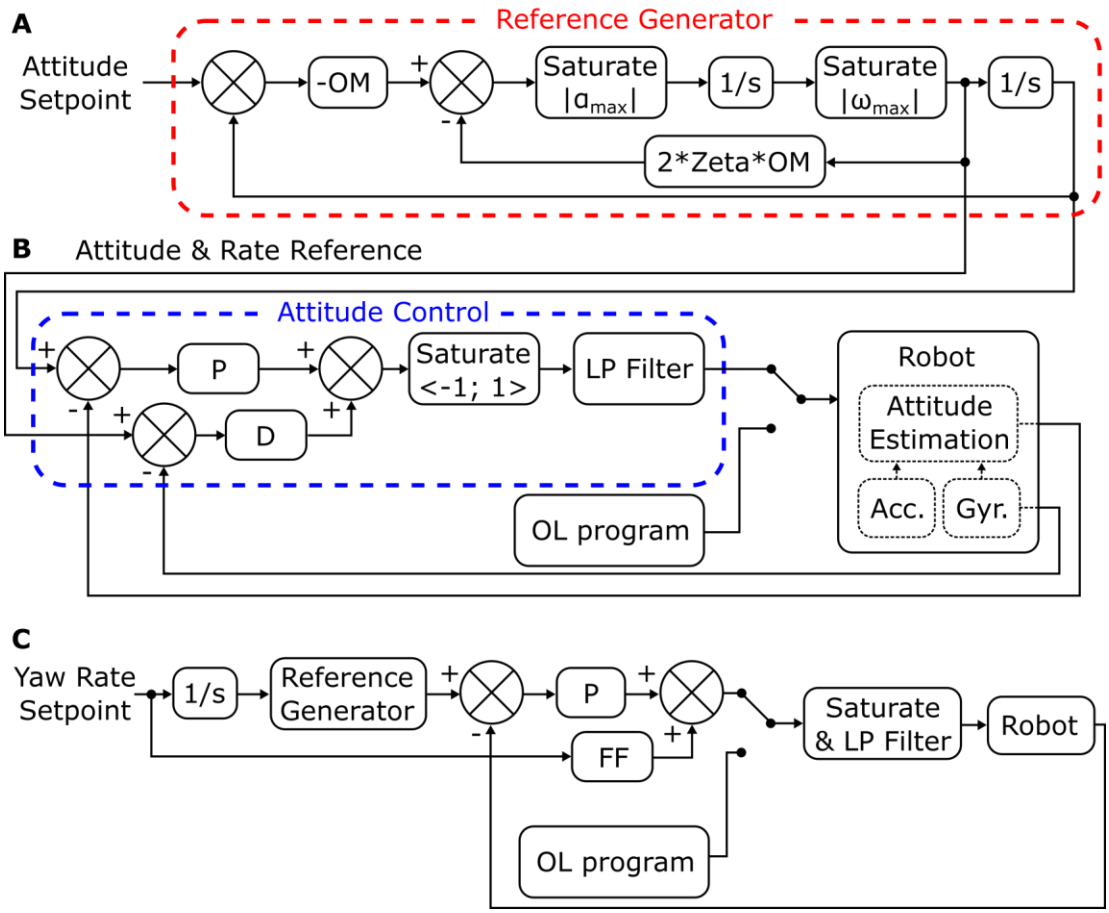

Fig. S7.

Schematics of the on-board controller: (**A**) Reference generator; (**B**) Bio-inspired attitude controller with proportional and derivative terms (equivalent to the PI controller on rates used by flies). The controller structure is identical for pitch and roll. During the rapid banked turns, the autopilot deactivates the attitude control loops and uses an open loop (OL) program instead. (**C**) The yaw loop employs only the proportional term and an extra feedforward term. The autopilot can also switch to an OL program. (**A–C**) The control parameter values are listed in Table S3.

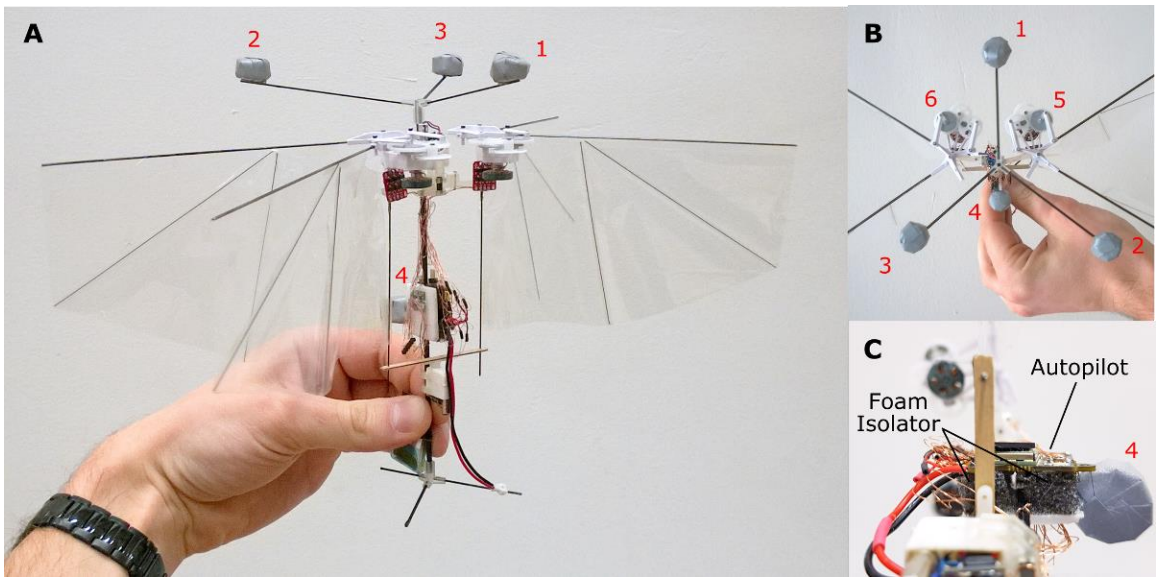

Fig. S8.

The robot with reflective markers of the motion tracking system and the autopilot soft mount. (**A**) Four approximately spherical markers defining the body. (**B**) Two additional flat circular markers (5 & 6) are used for tracking the adjustable dihedral angle (top view). (**C**) Detail on the autopilot mount made of soft PU foam, which serves as a vibration isolator.

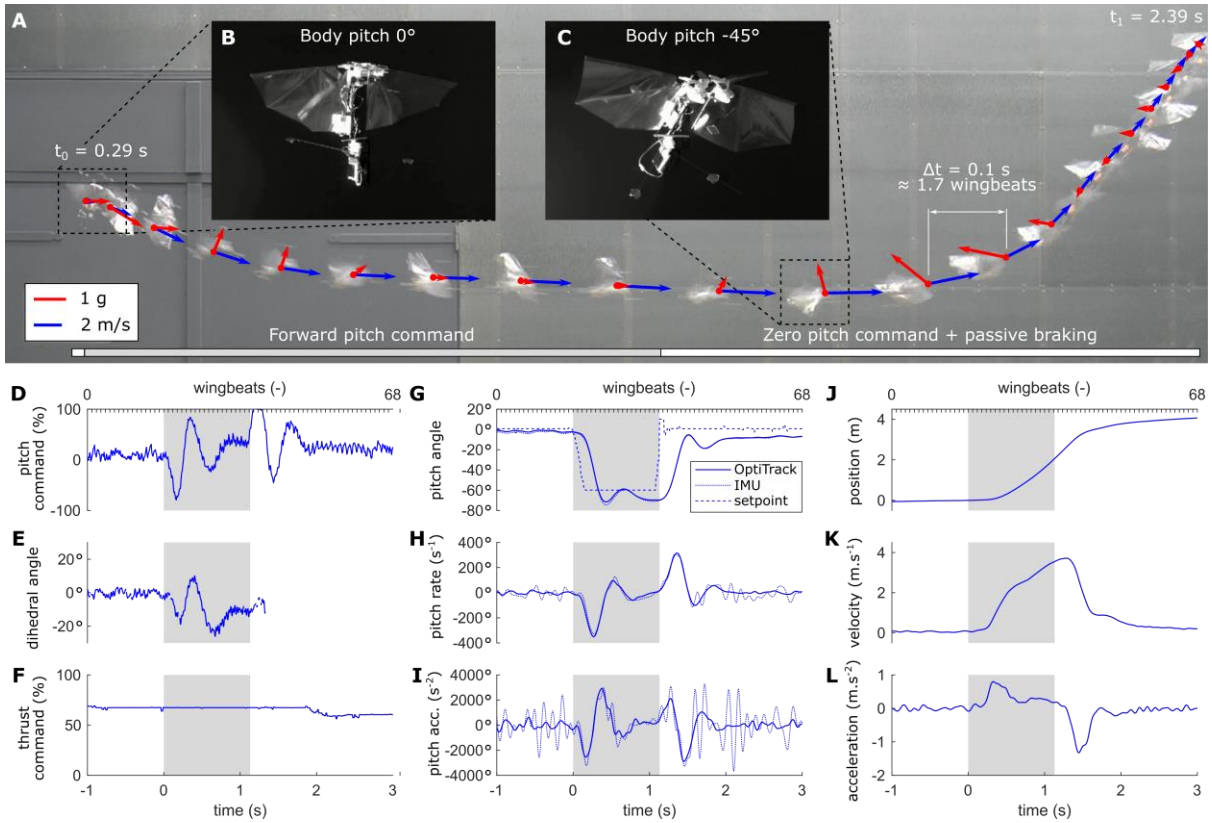

Fig. S9.

Rapid transition from hover to fast forward flight and vice versa. (**A**) A composite image of high speed camera frames, captured at constant time intervals of 0.1 s (approx. 1.7 wingbeats). In overlay, the speed and acceleration vectors are represented by blue and red arrows, respectively, scaled according to the scale bars on the bottom left. (**B**) A close-up on the flying vehicle in hover and (**C**) in forward flight. (**D–L**) The flight data captured during the maneuver. (**D**) Command to the dihedral control mechanism; (**E**) Dihedral angle; (**F**) the thrust command; (**G**) pitch angle; (**H**) pitch rate; (**I**) pitch acceleration. (**G–I**) Motion tracking data (solid line), on board estimate (dotted line) and the setpoint given by the pilot (the dashed line). (**J**) Position, (**K**) velocity and (**L**) acceleration in the ground-fixed x-axis.

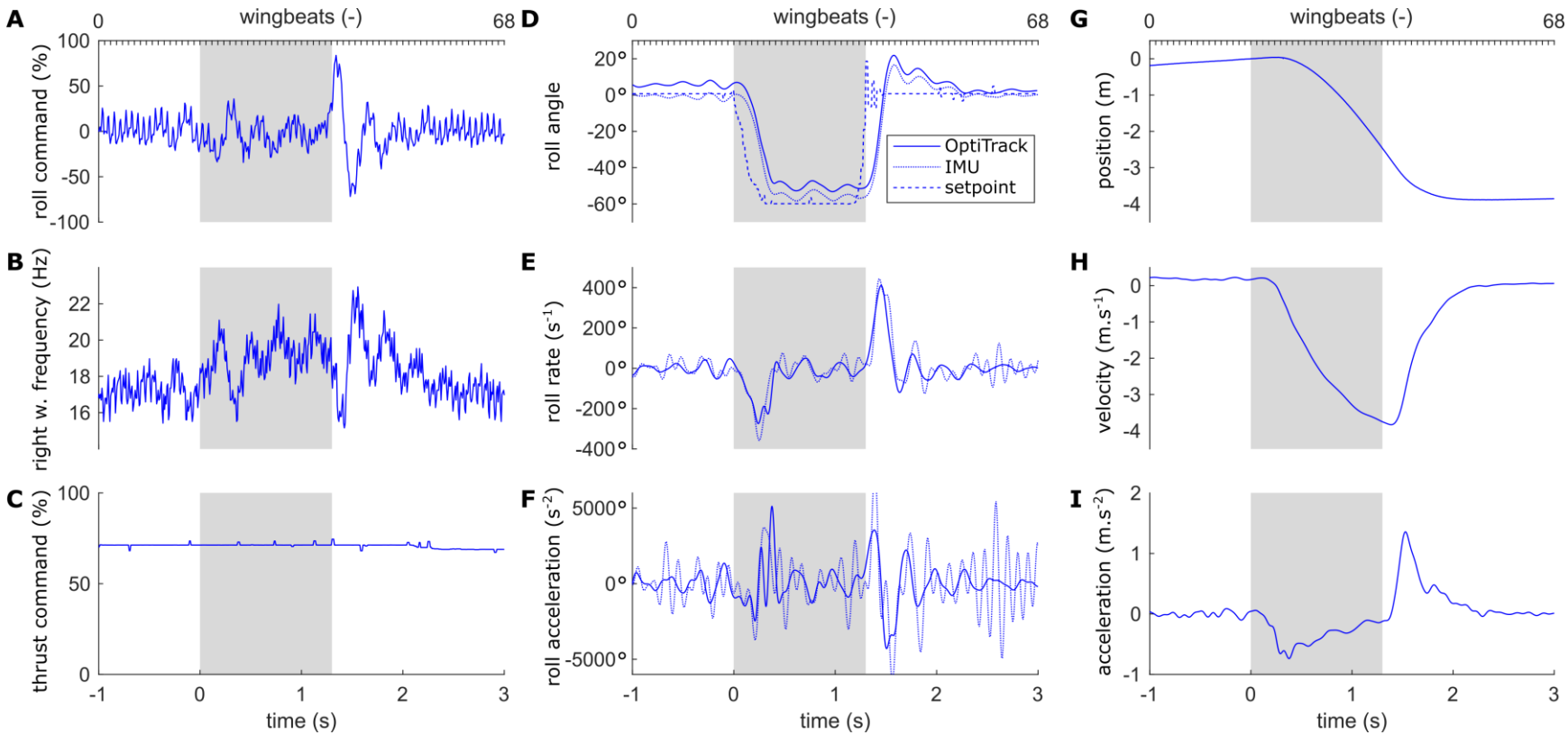

Fig. S10.

Rapid transition from hover to sideways flight and vice versa. (**A**) Roll commands; (**B**) Flapping frequency of the right wing-pair; (**C**) Thrust command; (**D**) Roll angle; (**E**) Roll rate; (**F**) Roll acceleration. (**D–F**) Motion tracking data (solid line), on board estimate (dotted line) and the setpoint given by the pilot (the dashed line). (**G**) Position, (**H**) speed and (**I**) acceleration in the ground-fixed y-axis.

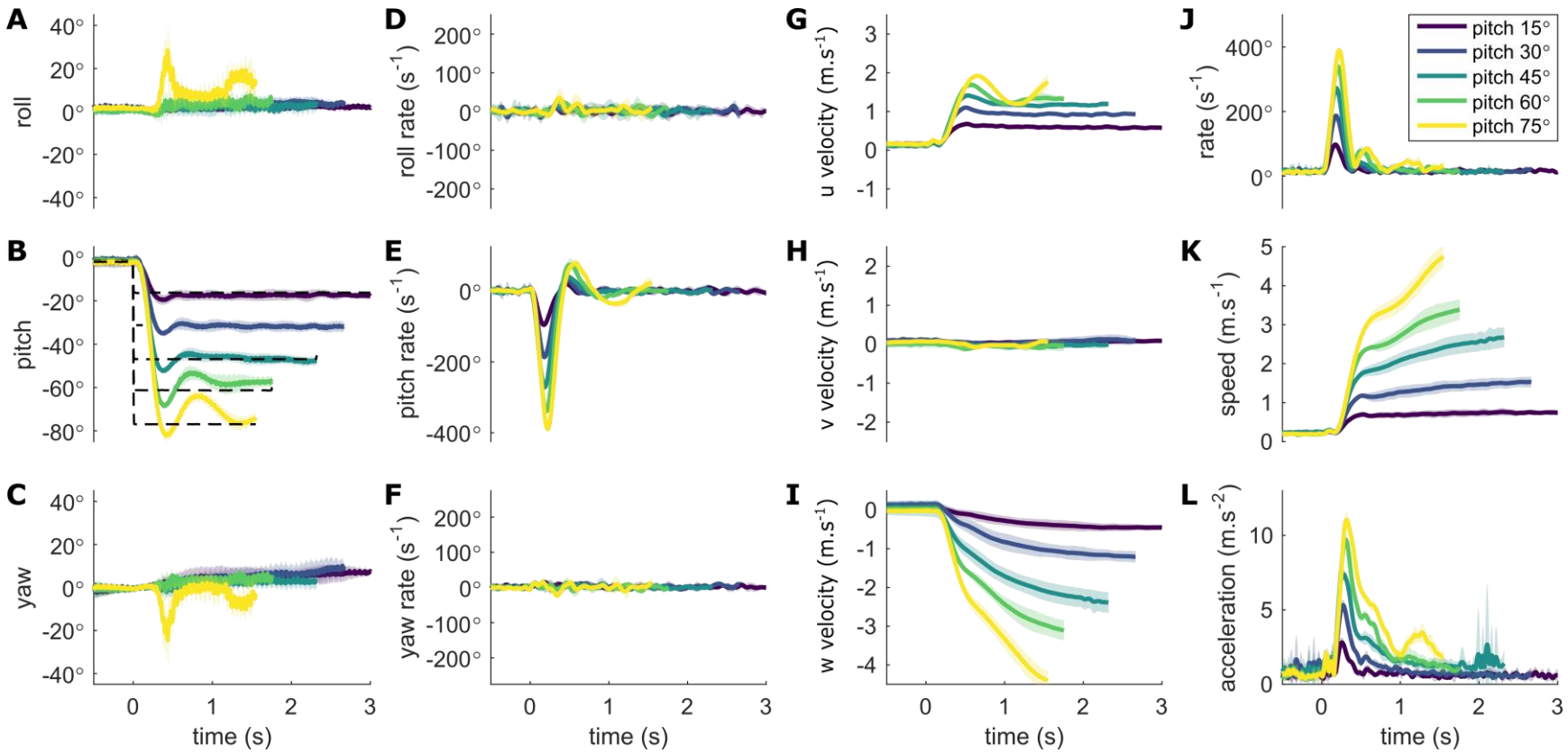

Fig. S11.

Reproducibility tests for transition from hover to forward flight with various pitch setpoints, color-coded according to the legend in panel (J). (**A–C**) Body attitude: (**A**) roll, (**B**) pitch, (**C**) yaw. (**D–F**) Body angular rates: (**D**) roll rate, (**E**) pitch rate, (**F**) yaw rate. (**G–I**) Body velocities (body frame): (**G**) x-velocity, (**H**) y-velocity, (**I**) z-velocity. (**J**) Angular rate magnitude; (**K**) Speed; (**L**) Acceleration. Thick lines represent the averages and the shaded error bars the uncertainty (± standard deviation). The commanded pitch setpoints are shown by the black dashed line in panel (B).

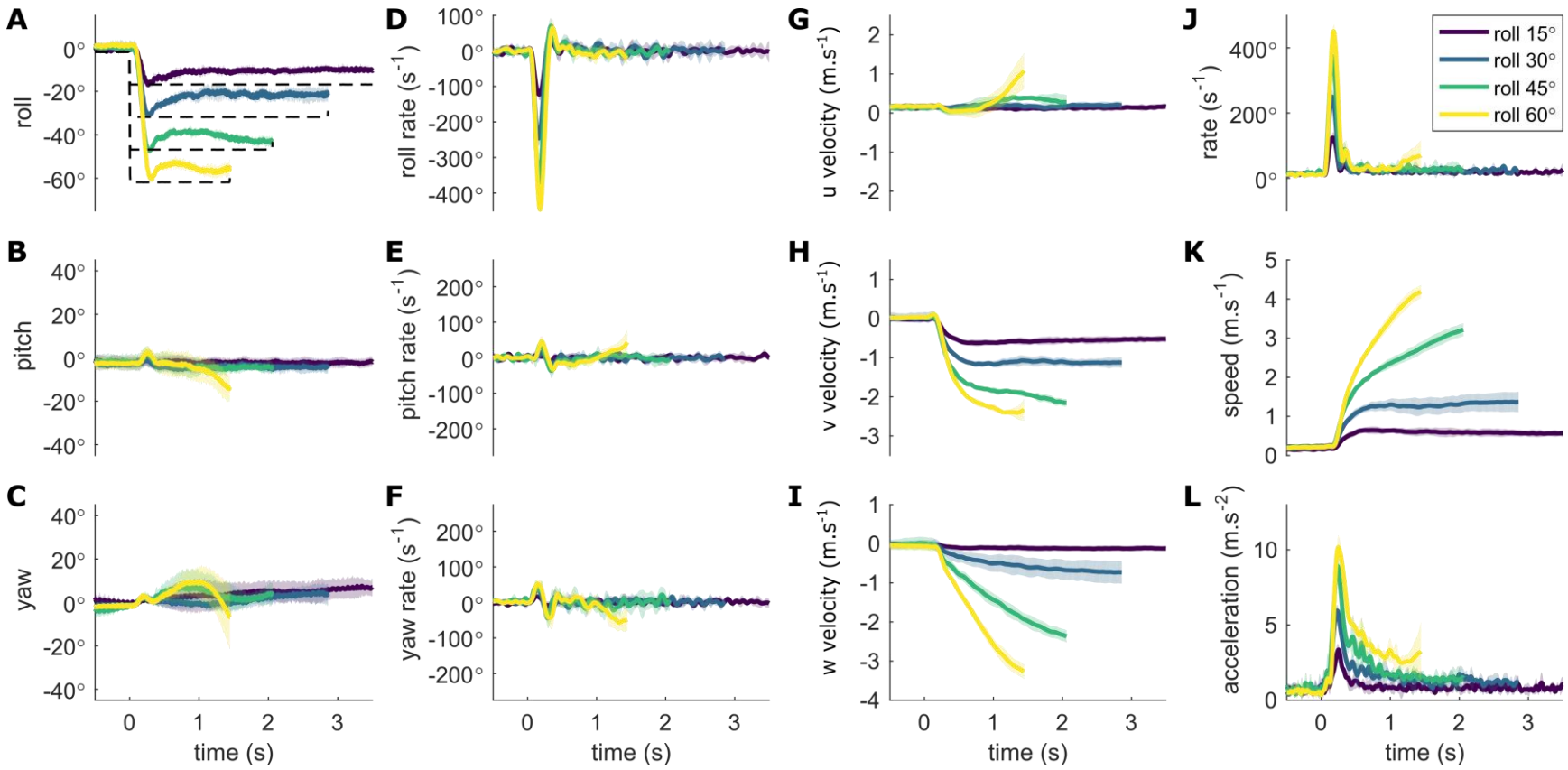

Fig. S12.

Reproducibility tests for transition from hover to sideways flight with various roll setpoints, color-coded according to the legend in panel (J). (**A–C**) Body attitude: (**A**) roll, (**B**) pitch, (**C**) yaw. (**D–F**) Body angular rates: (**D**) roll rate, (**E**) pitch rate, (**F**) yaw rate. (**G–I**) Body velocities (body frame): (**G**) x-velocity, (**H**) y-velocity, (**I**) z-velocity. (**J**) Angular rate magnitude; (**K**) Speed; (**L**) Acceleration. Thick lines represent the averages and the shaded error bars the uncertainty (± standard deviation). The commanded roll setpoints are shown by the black dashed line in panel (A).

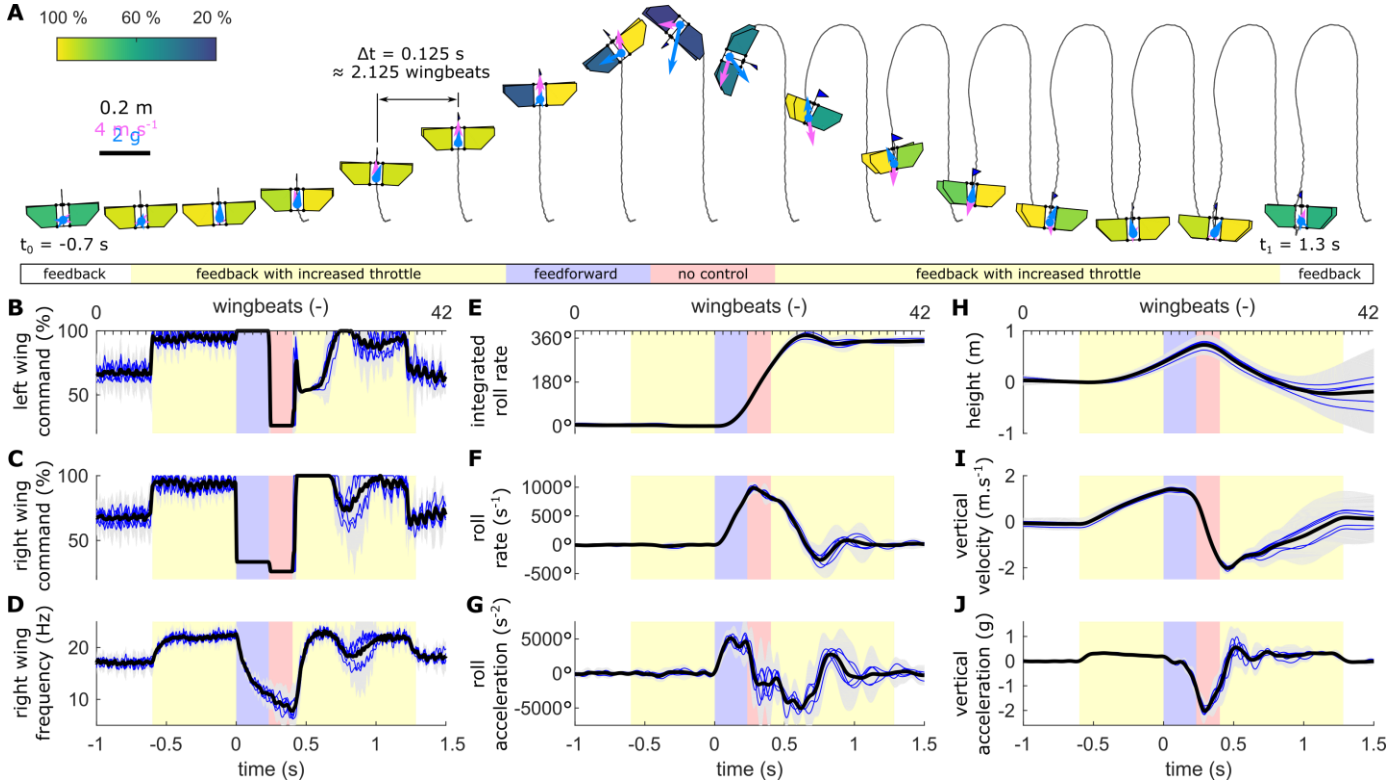

Fig. S13.

A 360° roll flip maneuver. (**A**) A sequence visualizing the robot at constant time intervals (0.125 seconds); to avoid overlap, individual time frames are shifted horizontally by a constant distance. The black line represents the flight trajectory, magenta and blue arrows the velocity and acceleration vectors, respectively, scaled according to the scale bar on the left. The thrust command to the individual wing-pairs is represented by their color, with the scale according to the color-bar on the top left. The bars at the bottom show individual phases of the maneuver; the same color coding is used in all the remaining panels. (**B–J**) The flight data recorded during 5 trials. The black thick line represents the mean, blue the individual trials, and gray shaded error bar the uncertainty (±3 standard deviations). (**B–C**) Commands to the left (**B**) and right (**C**) flapping mechanisms; (**D**) Flapping frequency of the right wing-pair; (**E**) Integrated roll rate; (**F**) Roll rate; (**G**) Roll acceleration; (**H**) Height; (**I**) Vertical velocity; (**J**) Vertical acceleration.

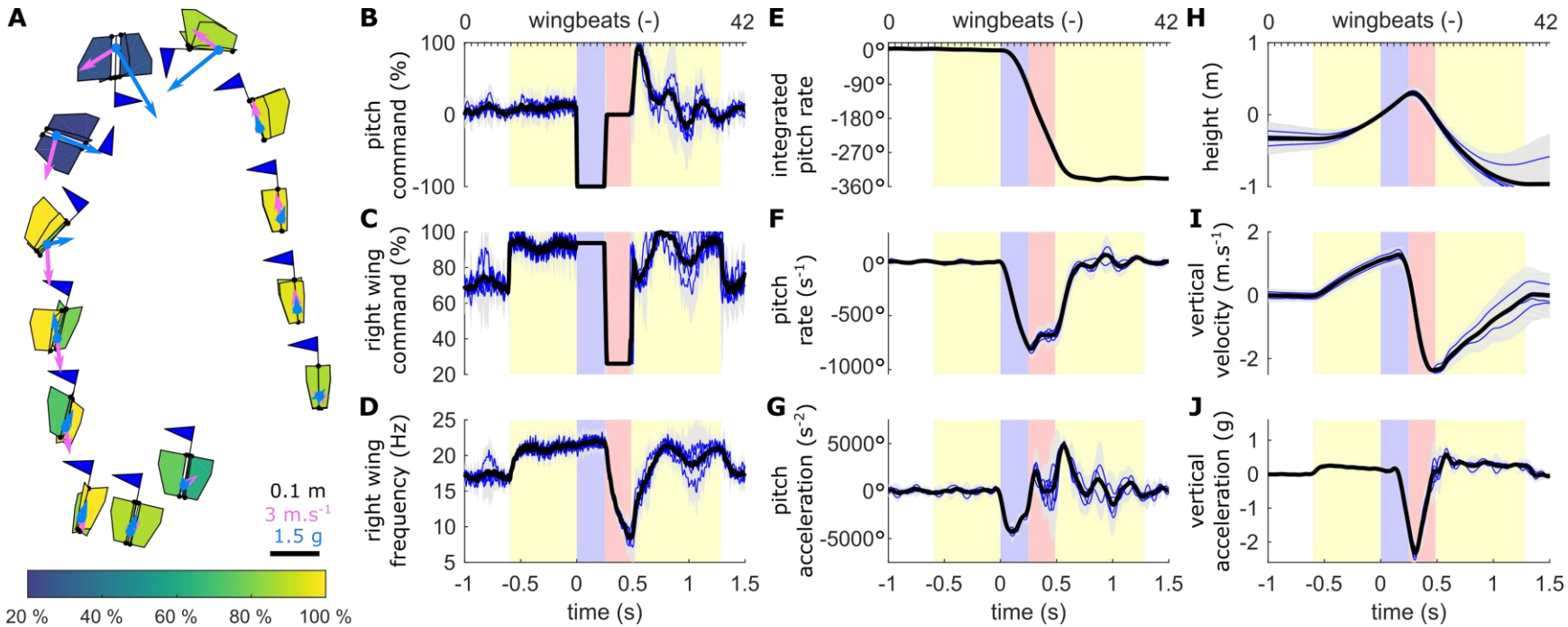

Fig. S14.

A 360° pitch flip maneuver. (**A**) A sequence visualizing the robot at various phases of the maneuver, with variable time intervals at correct spatial locations. The magenta and blue arrows represent the velocity and acceleration vectors, respectively. The thrust command to the individual wing-pairs is represented by their varying color, with the scale according to the color-bar at the bottom of the panel. (**B–J**) The flight data recorded during 5 trials. The black thick line represents the mean, blue the individual trials, and grey shaded error bar the uncertainty (±3 standard deviations). The color shading corresponds to the respective maneuver phases as in Fig. S13. (**B**) Pitch command; (**C**) Command to the right flapping mechanism; (**D**) Flapping frequency of the right wing-pair; (**E**) Integrated pitch rate; (**F**) Pitch rate; (**G**) Pitch acceleration; (**H**) Height; (**I**) Vertical velocity; (**J**) Vertical acceleration.

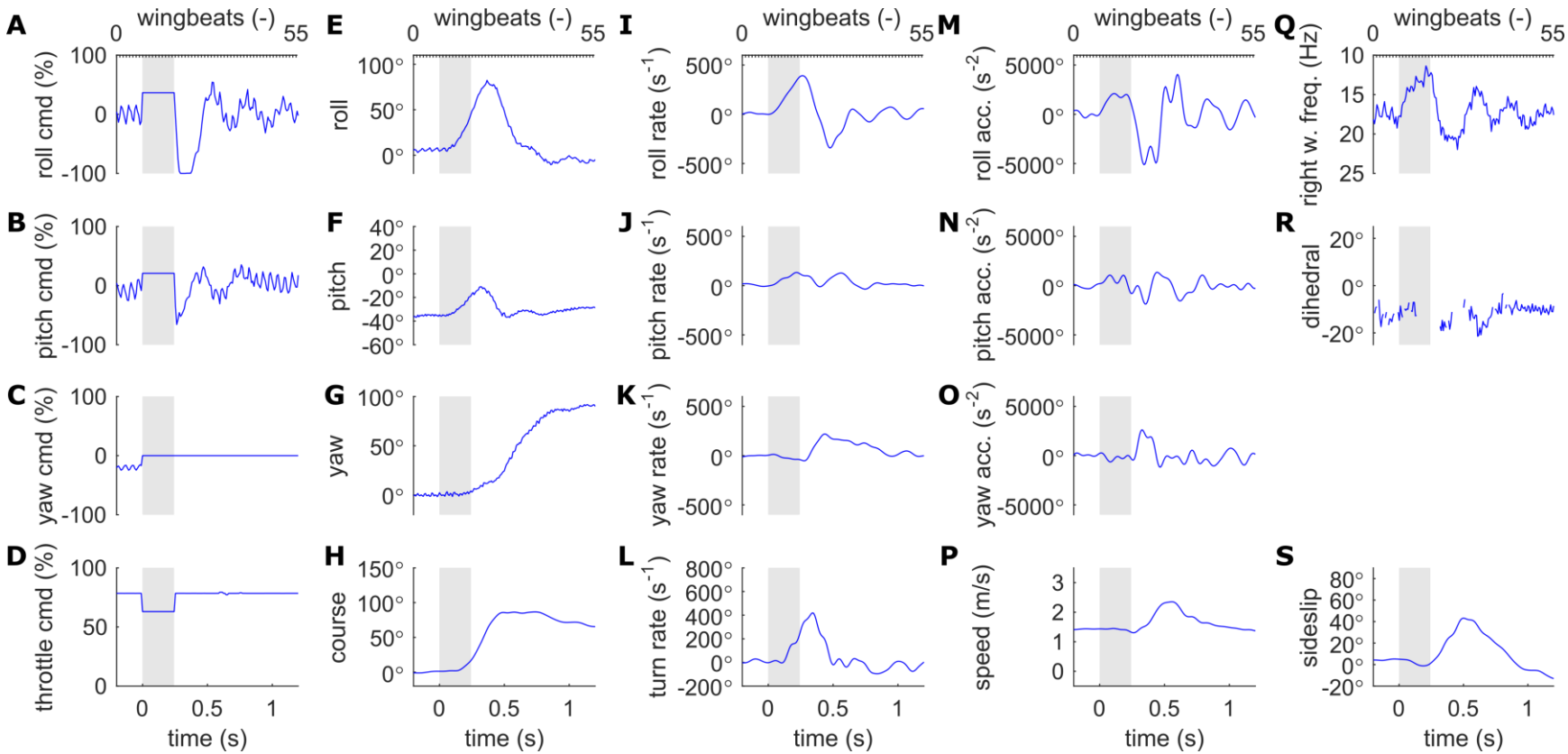

Fig. S15.

Flight data for the maneuver in Fig 2A. (**A–D**) The actuator commands: (**A**) roll actuator command, (**B**) pitch actuator command, (**C**) yaw actuator command, (**D**) throttle command. (**E–G**) Body attitude: (**E**) roll, (**F**) pitch, (**G**) yaw. (**H**) Course. (**I–K**) Body angular rates: (**I**) roll rate, (**J**) pitch rate, (**K**) yaw rate. (**L**) Turn rate. (**M–O**) Body angular accelerations: (**M**) roll acceleration, (**N**) pitch acceleration, (**O**) yaw acceleration. (**P**) Body speed; (**Q**) Flapping frequency of the right wing-pair; (**R**) Dihedral angle; (**S**) Sideslip angle. The gray background in all the panels highlights the OL phase of the maneuver.

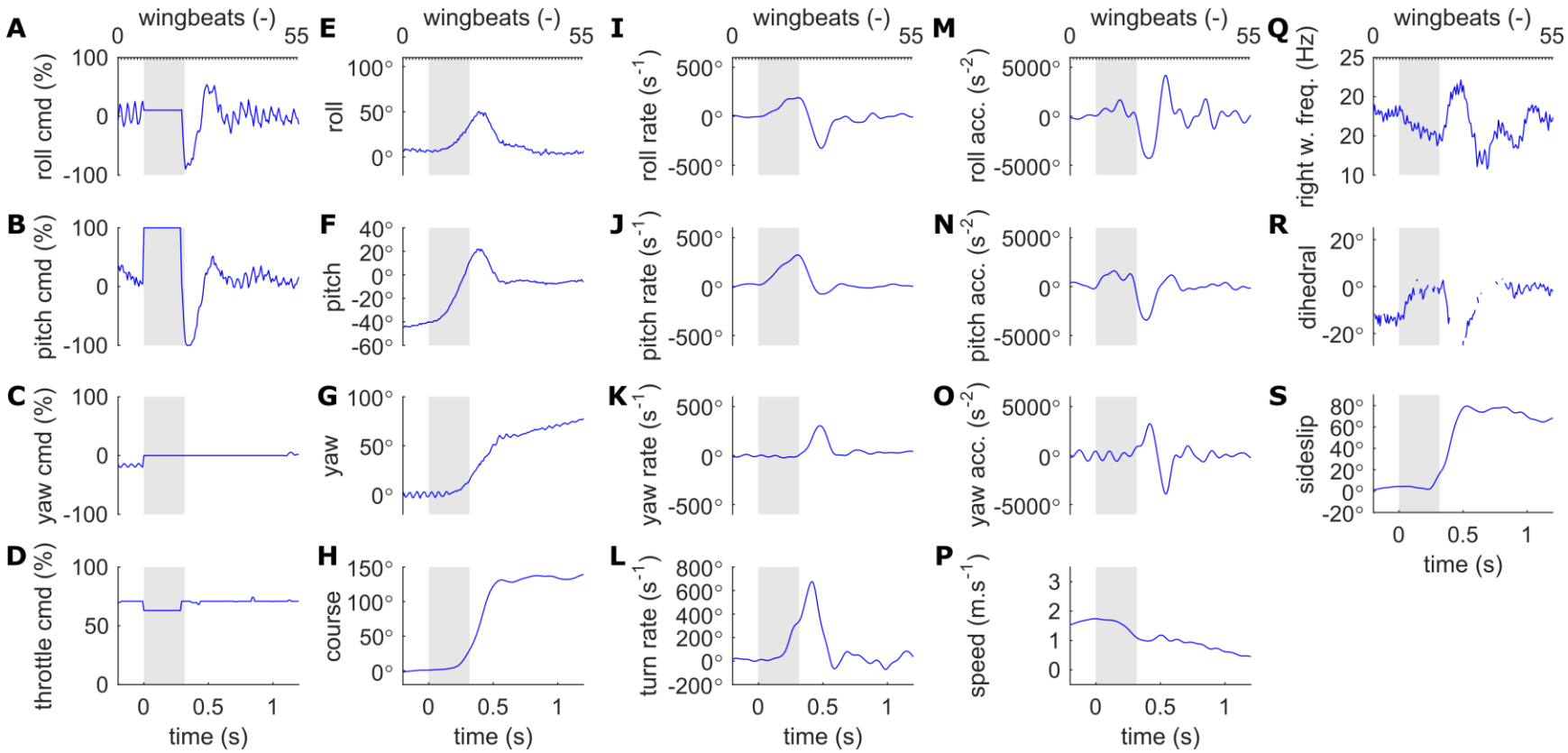

Fig. S16.

Flight data for the maneuver in Fig 2C. (**A–D**) The actuator commands: (**A**) roll actuator command, (**B**) pitch actuator command, (**C**) yaw actuator command, (**D**) throttle command. (**E–G**) Body attitude: (**E**) roll, (**F**) pitch, (**G**) yaw. (**H**) Course. (**I–K**) Body angular rates: (**I**) roll rate, (**J**) pitch rate, (**K**) yaw rate. (**L**) Turn rate. (**M–O**) Body angular accelerations: (**M**) roll acceleration, (**N**) pitch acceleration, (**O**) yaw acceleration. (**P**) Body speed; (**Q**) Flapping frequency of the right wing-pair; (**R**) Dihedral angle; (**S**) Sideslip angle. The gray background in all the panels highlights the OL phase of the maneuver.

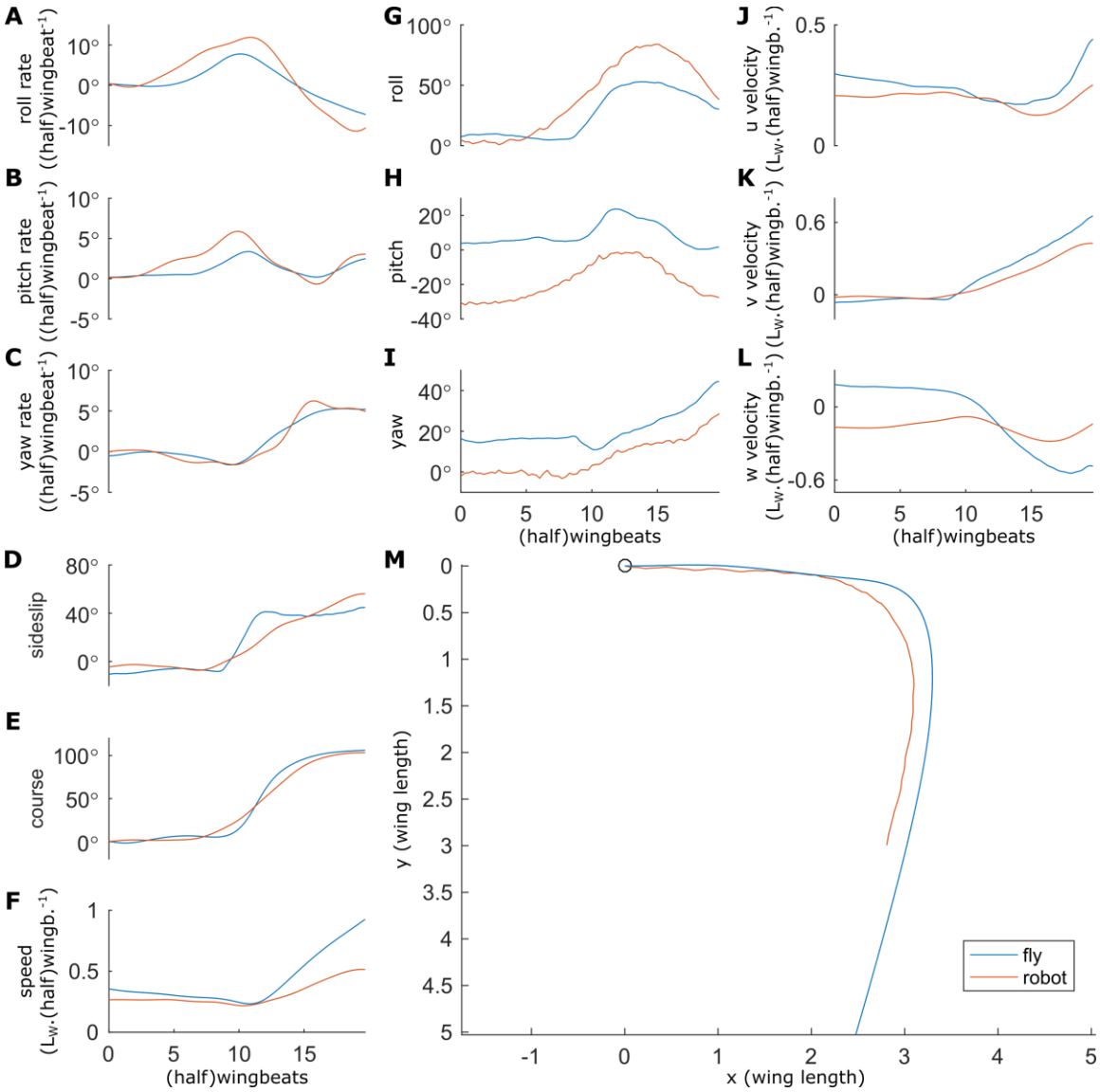

Fig. S17.

Comparison of the scaled flight data for the robot maneuver in Fig. 2A (red lines) and the fruit fly maneuver in Fig. 2B (blue lines). The lengths and time were scaled by wing length and wingbeat cycle time (fly) or half of wingbeat cycle time (robot), respectively. (**A–C**) Body angular rates: (**A**) roll rate, (**B**) pitch rate, (**C**) yaw rate. (**D**) Sideslip angle; (**E**) Course; (**F**) Speed. (**G–I**) Body attitude: (**G**) roll, (**H**) pitch, (**I**) yaw. (**J–L**) Body velocities: (**J**) velocity in body *x*-direction; (**K**) velocity in body *y*-direction, (**L**) velocity in body *z*-direction. (**M**) Flight trajectory (top view). The circle represents the position at the start of the maneuver.

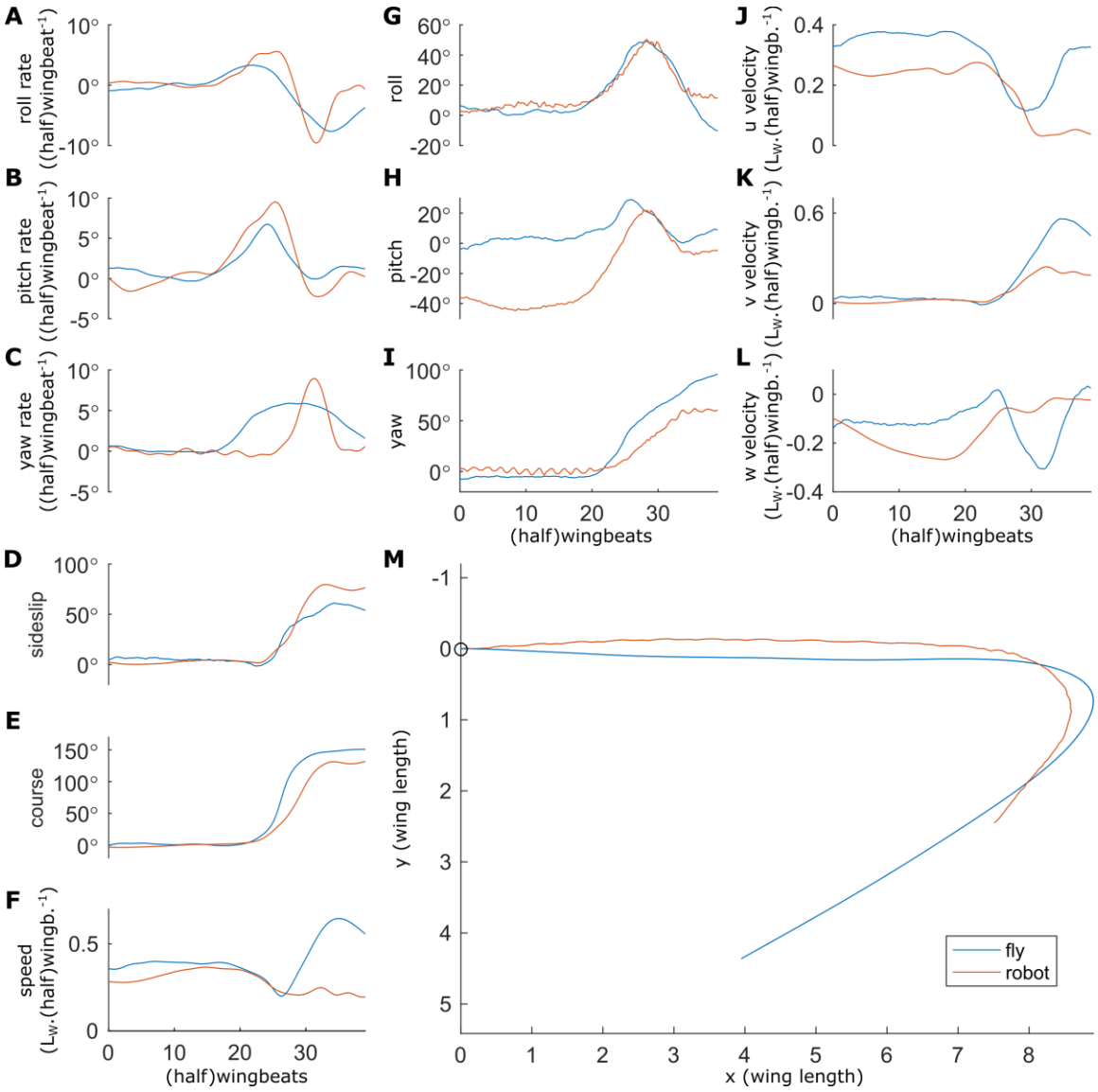

Fig. S18.

Comparison of the scaled flight data for the robot maneuver in Fig. 2C (red lines) and the fruit fly maneuver in Fig. 2D (blue lines). The lengths and time were scaled by wing length and wingbeat cycle time (fly) or half of wingbeat cycle time (robot), respectively. (**A–C**) Body angular rates: (**A**) roll rate, (**B**) pitch rate, (**C**) yaw rate. (**D**) Sideslip angle; (**E**) Course; (**F**) Speed. (**G–I**) Body attitude: (**G**) roll, (**H**) pitch, (**I**) yaw. (**J–L**) Body velocities: (**J**) velocity in body *x*-direction; (**K**) velocity in body *y*-direction, (**L**) velocity in body *z*-direction. (**M**) Flight trajectory (top view). The circle represents the position at the start of the maneuver.

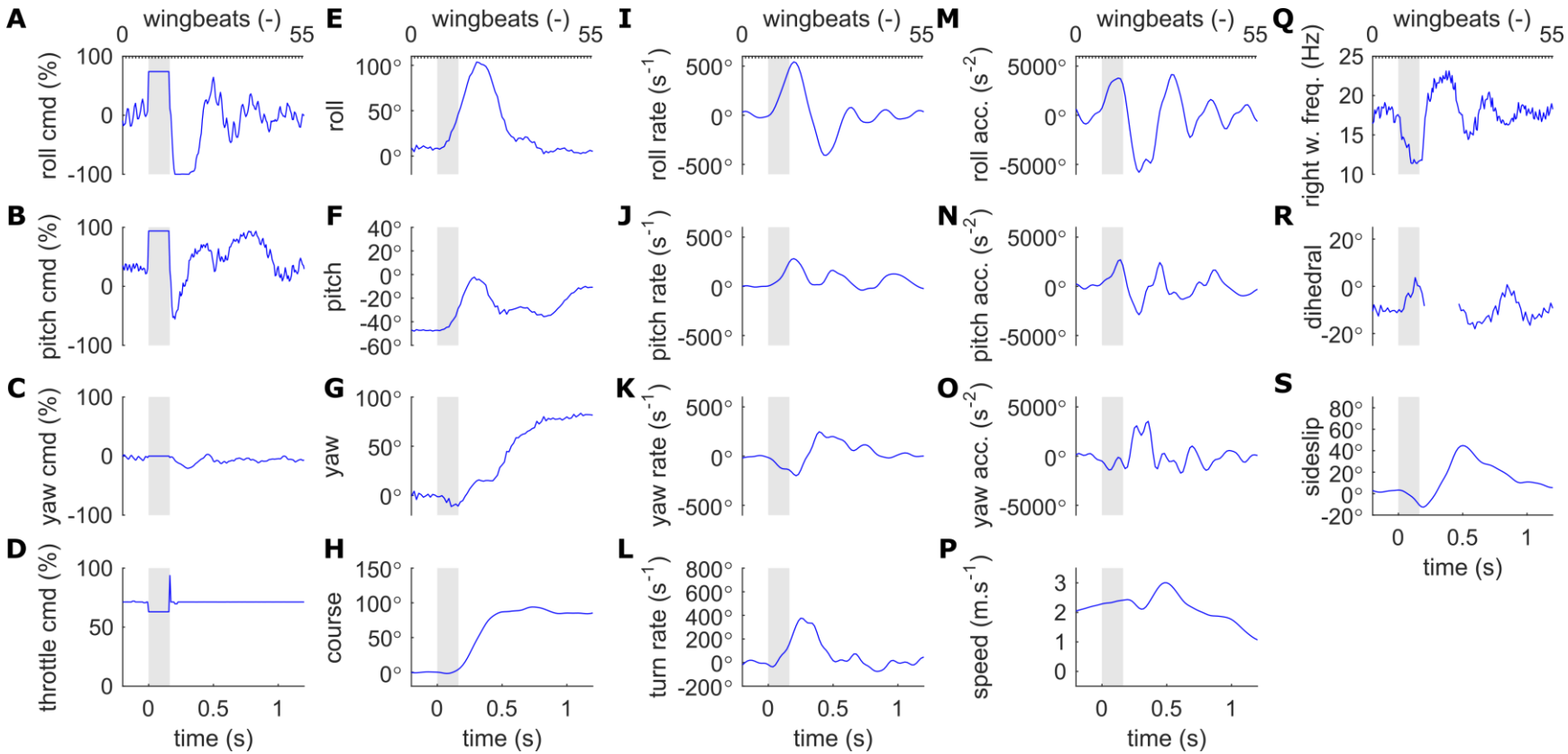

Fig. S19.

Flight data for the maneuver in Fig 2E. (**A–D**) The actuator commands: (**A**) roll actuator command, (**B**) pitch actuator command, (**C**) yaw actuator command, (**D**) throttle command. (**E–G**) Body attitude: (**E**) roll, (**F**) pitch, (**G**) yaw. (**H**) Course. (**I–K**) Body angular rates: (**I**) roll rate, (**J**) pitch rate, (**K**) yaw rate. (**L**) Turn rate. (**M–O**) Body angular accelerations: (**M**) roll acceleration, (**N**) pitch acceleration, (**O**) yaw acceleration. (**P**) Body speed; (**Q**) Flapping frequency of the right wing-pair; (**R**) Dihedral angle; (**S**) Sideslip angle. The gray background in all the panels highlights the OL phase of the maneuver.

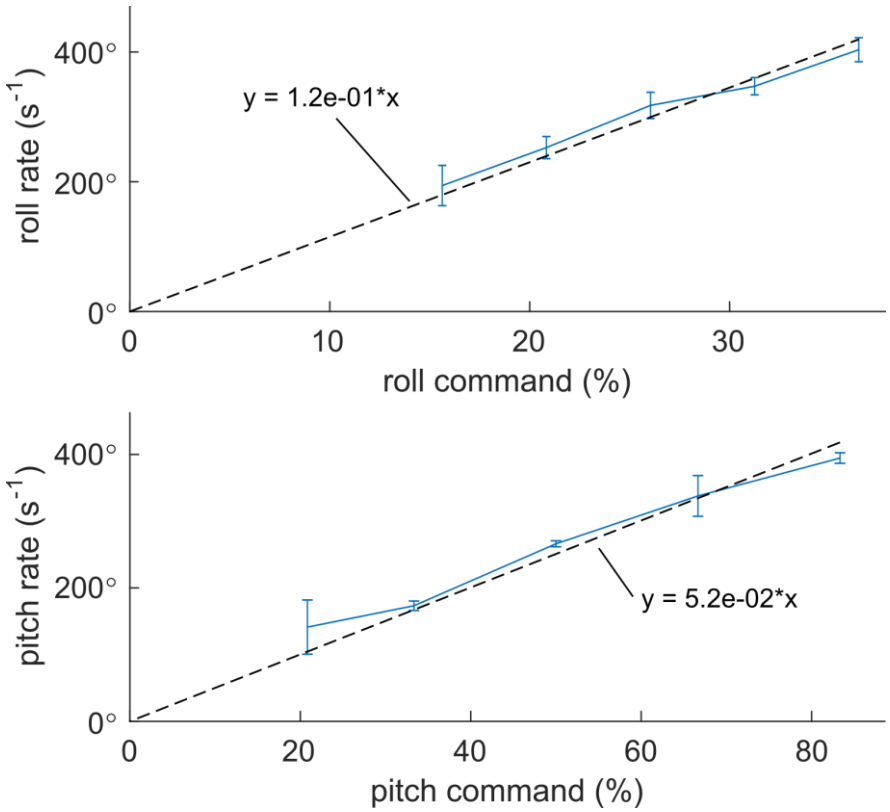

Fig. S20.

The angular rates reached at the end of the OL phase of the maneuvers in Fig. 3 remain approximately linear functions of the commands. (**A**) Roll rate *p* against roll command; (**B**) Pitch rate *q* against pitch command. (**A**, **B**) Blue line shows the average of all the trials, and the errorbars show ±s.d., black dashed line represents a linear fit.

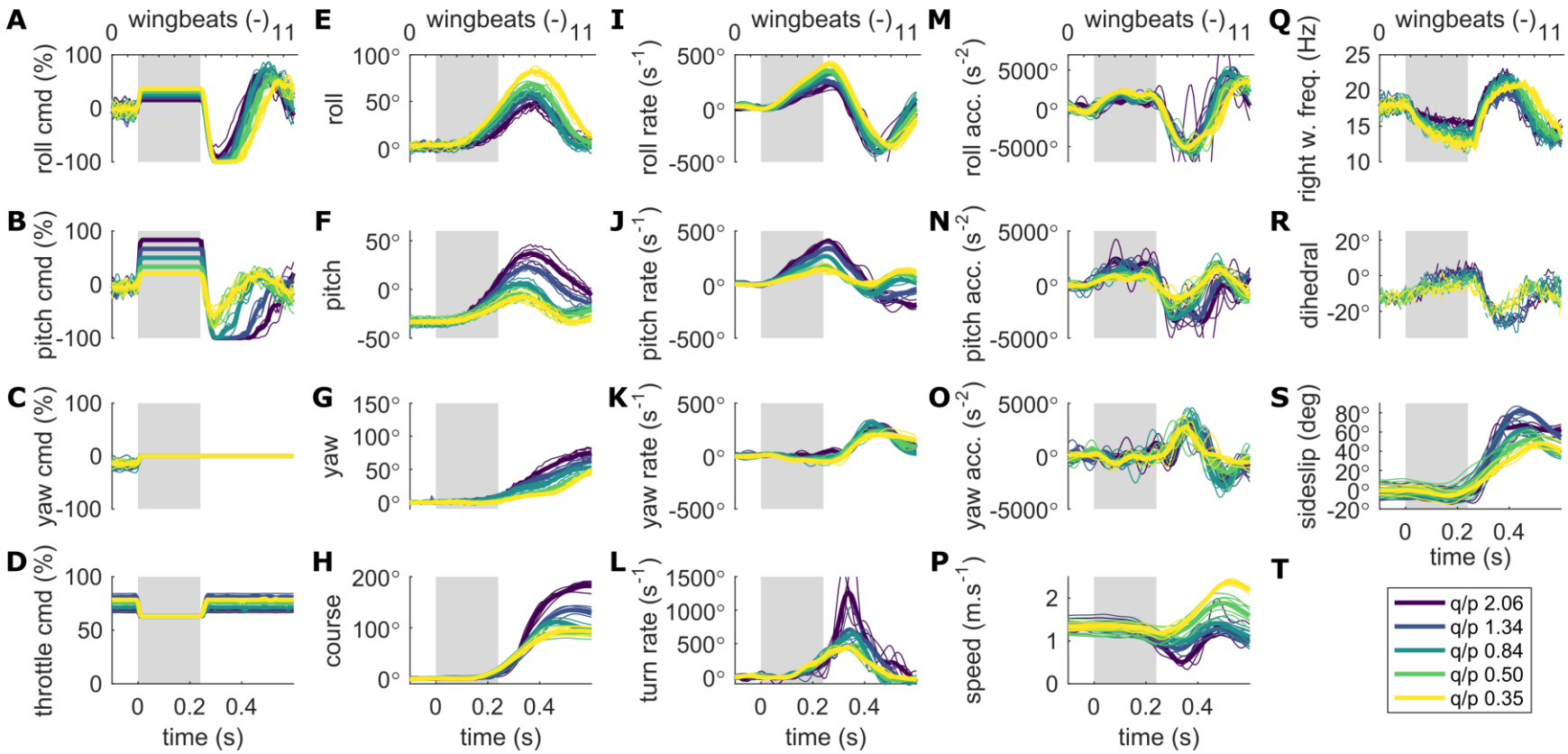

Fig. S21.

Flight data for the maneuvers in Fig 3. (**A–D**) The actuator commands: (**A**) roll actuator command, (**B**) pitch actuator command, (**C**) yaw actuator command, (**D**) throttle command. (**E–G**) Body attitude: (**E**) roll, (**F**) pitch, (**G**) yaw. (**H**) Course. (**I–K**) Body angular rates: (**I**) roll rate, (**J**) pitch rate, (**K**) yaw rate. (**L**) Turn rate. (**M–O**) Body angular accelerations: (**M**) roll acceleration, (**N**) pitch acceleration, (**O**) yaw acceleration. (**P**) Body velocity. (**Q**) Flapping frequency of the right wing-pair. (**R**) Dihedral angle. (**S**) Sideslip angle. The data is color coded according to the legend in panel (**T**), thin lines are individual trials, thick lines the averages. The gray background in all the panels highlights the OL phase of the maneuver.

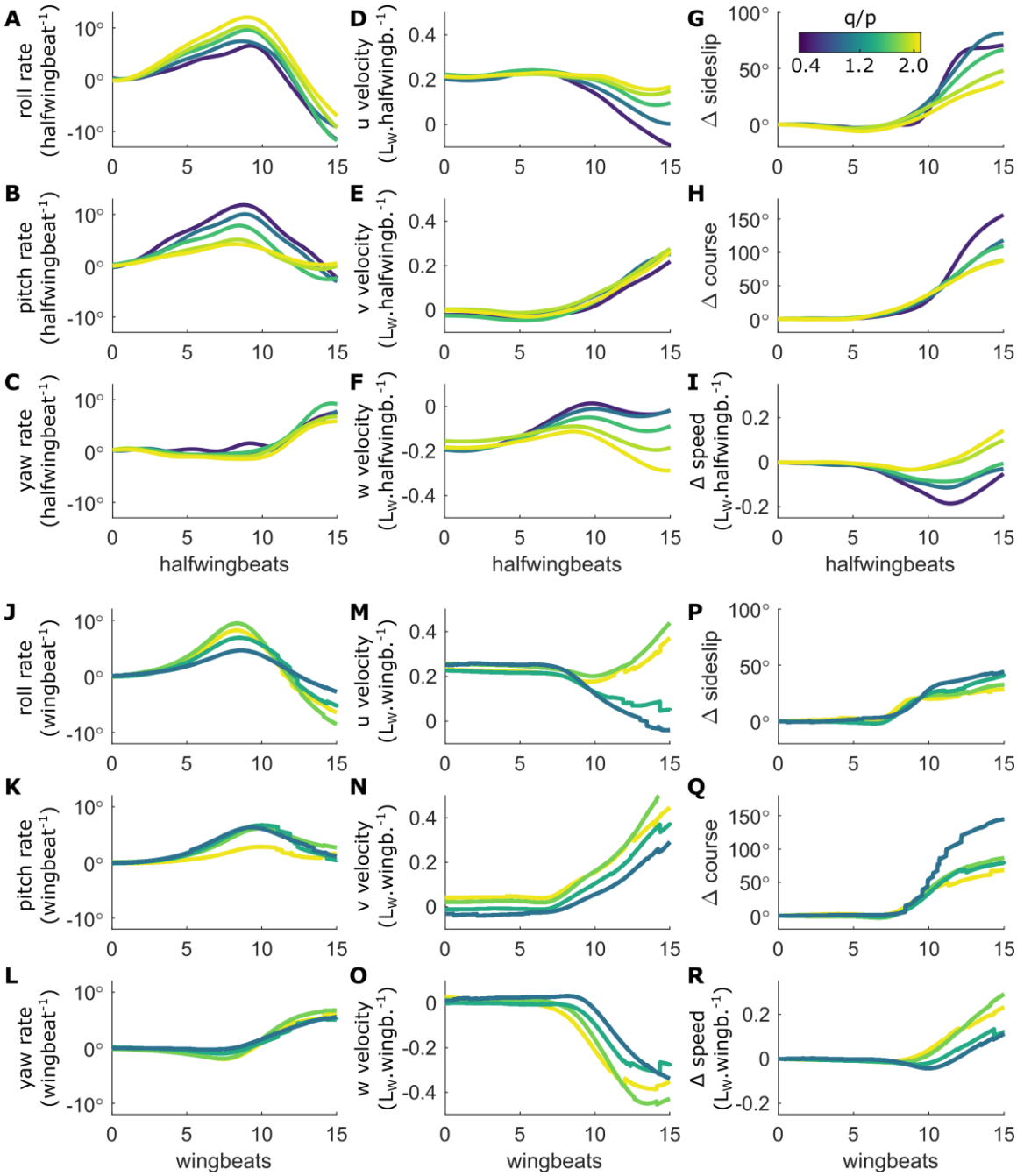

Fig. S22.

Comparison of rapid banked turns performed by the robot (**A–I**) and by fruit flies (**J–R**) for various *q*/*p*, color coded according to the colorbar in panel (G). Each line represents an average of multiple maneuvers (Fig. S21, (*3*)). The lengths and time were scaled by wing length and wingbeat cycle time (fly) or half of wingbeat cycle time (robot), respectively. (**A–C**, **J–L**) Body angular rates: (**A**, **J**) roll rate, (**B**, **K**) pitch rate, (**C**, **L**) yaw rate. (**D–F**, **M–O**) Body velocities: (**D**, **M**) velocity in body *x*-direction; (**E**, **N**) velocity in body *y*-direction, (**F**, **O**) velocity in body *z*-direction. (**G**, **P**) Sideslip change; (**E**, **N**) Course change; (**F**, **O**) Speed change.

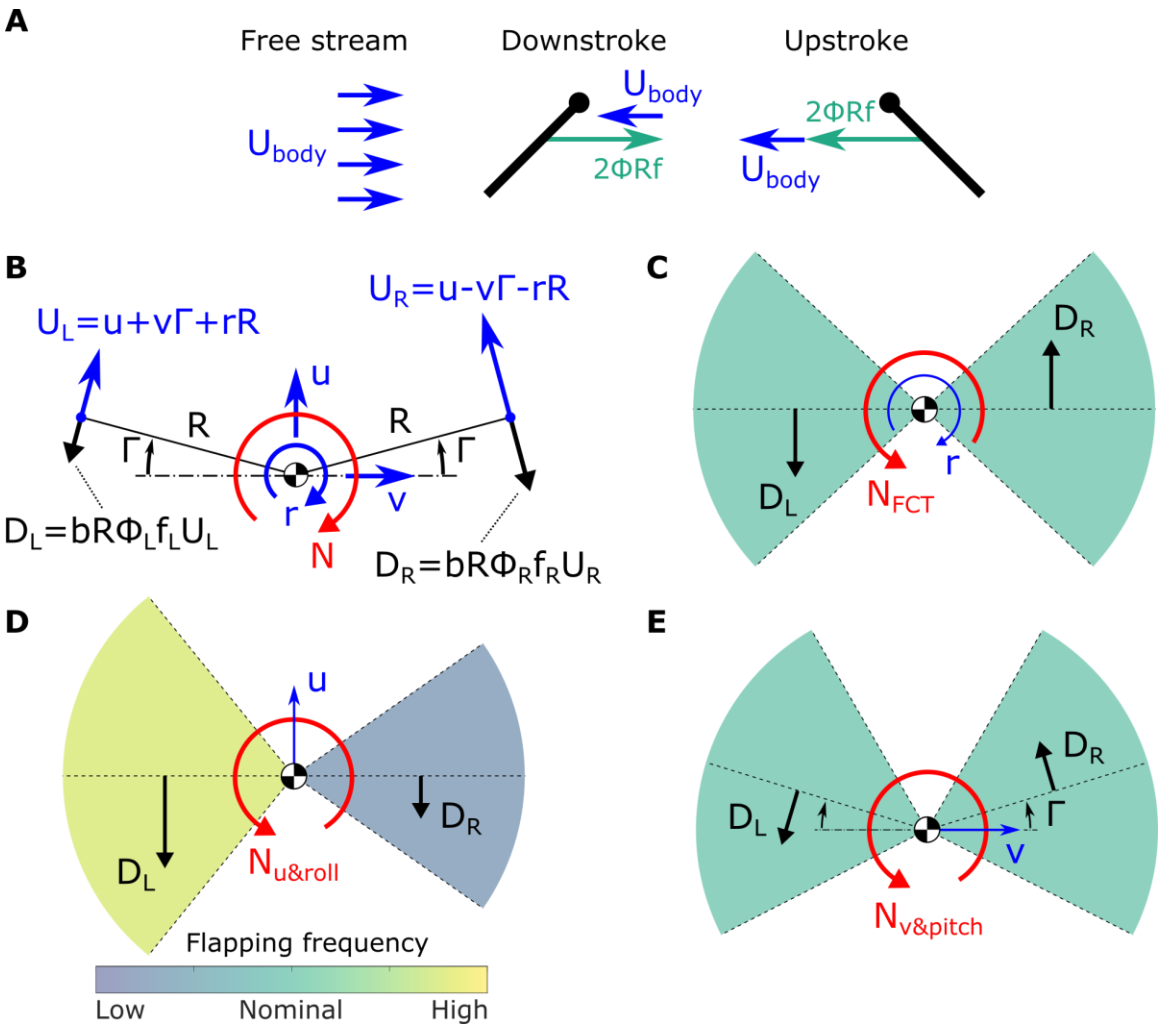

Fig. S23.

A drag-based model explaining the high yaw accelerations during the recovery phase of the rapid banked turn. (**A**) A 2D wing section moving in the direction of the free stream during the downstroke and opposite to its direction during the upstroke. (**B**) Top view diagram showing the effect of the damping forces due to body motion in the horizontal plane on the yaw dynamics. The resultant torque consists of three components: (**C**) flapping-counter-torque; (**D**) torque due to coupling of forward velocity with roll torque generation mechanism via flapping frequency and/or amplitude difference; (**E**) torque due to coupling of sideways velocity with pitch torque generation mechanism via dihedral (i.e. mean stroke) angle.

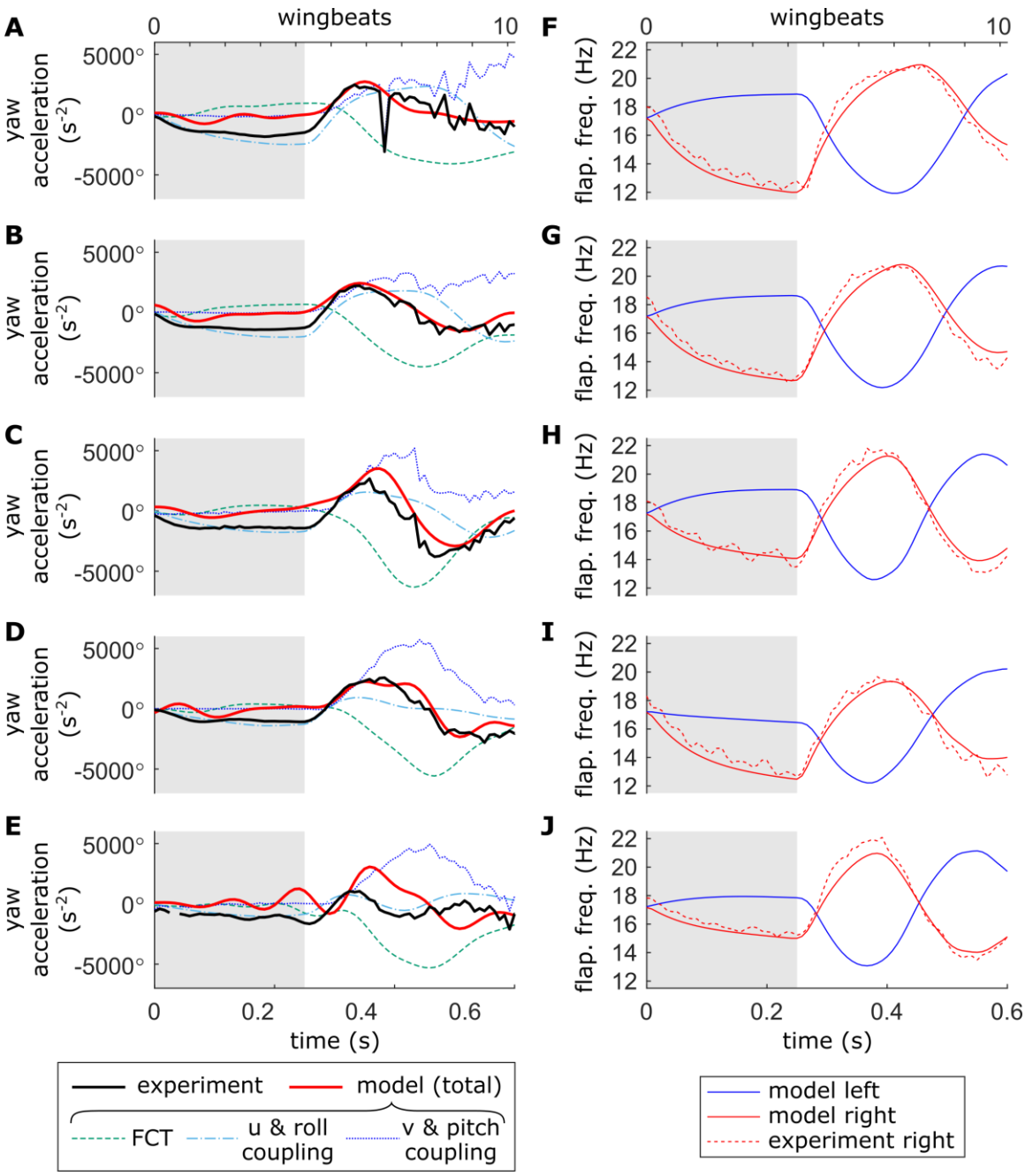

Fig. S24.

The yaw accelerations observed during the banked turn can be explained by coupling of pitch and torque generation mechanisms with horizontal body motion. (**A**–**E**) Yaw accelerations predicted by the model and measured. Individual model components are color coded according to the legend at the bottom of panel E. (**F**–**J**) Flapping frequencies estimated by the model and the measured right wing flapping frequency. (**A**, **F**) Turns with q/p = 0.35. (**B**, **G**) Turns with q/p = 0.50. (**C**, **H**) Turns with q/p = 0.84. (**D**, **I**) Turns with q/p = 1.34. (**F**, **J**) Turns with q/p = 2.06.

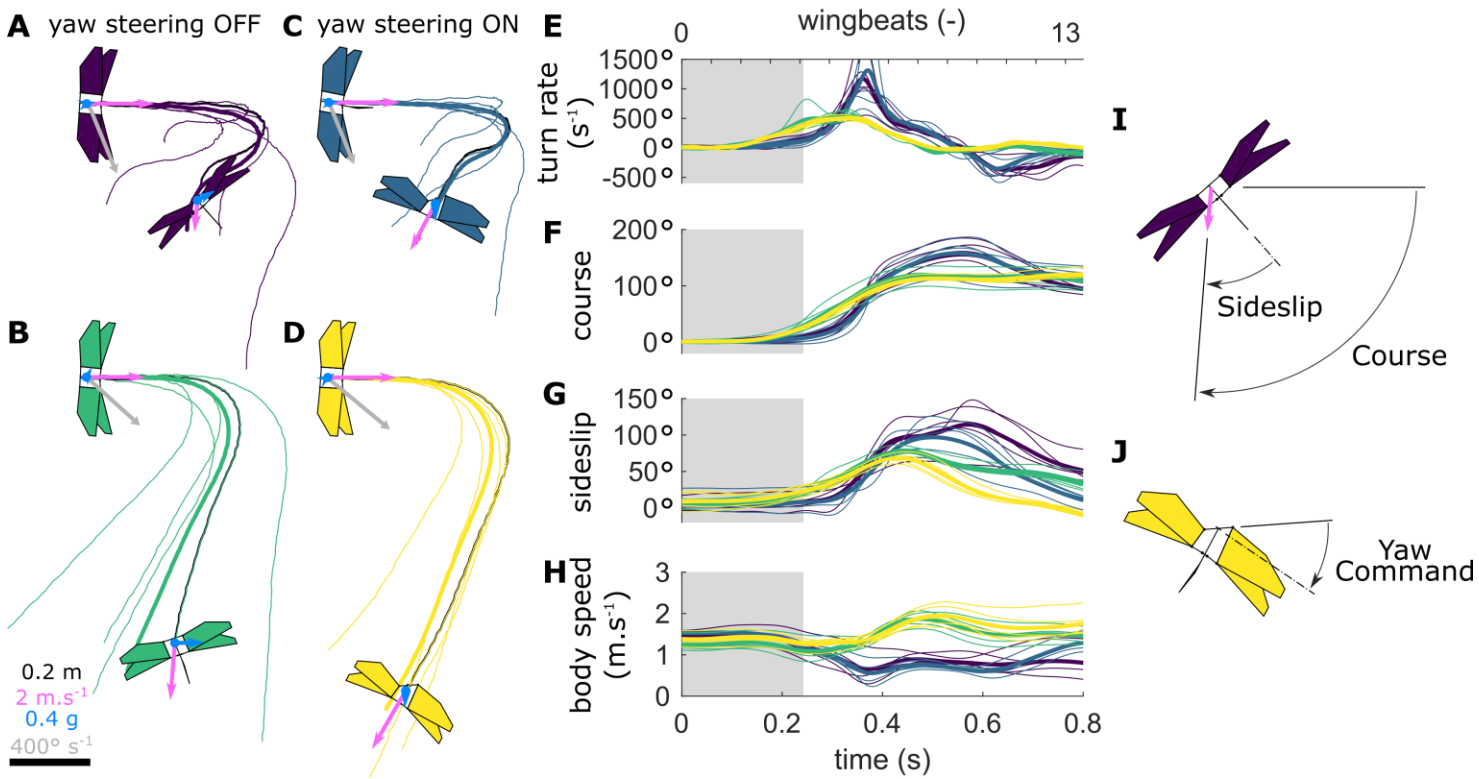

Fig. S25.

The sideslip experienced at the end of the turn can be eliminated by active yaw steering without a noteworthy influence on the rest of the maneuver dynamics. Thin lines are individual trials, and thick lines series averages. (**A**, **B**) Tests without active yaw steering. (**C**, **D**) Tests with maximum yaw torque production during the recovery phase. (**A**–**D**) Top view of trajectories; for one trial, the robot is displayed at the beginning of the OL phase (t = 0 s) and at t = 0.8 s. The gray arrows represent the angular rates vector at the end of the OL phase, and the magenta and blue arrows show flight speed and acceleration, respectively, see scale bar in (B). (**E**–**H**) Time histories of maneuver kinematics, whereby line colors match those in (A–D); the OL phase is highlighted by a gray background, and is immediately followed by the recovery phase. (**E**) Turn rate (rate of change of course); (**F**) Course; (**G**) Sideslip; (**H**) Body speed (velocity vector magnitude); (**I**) Definition of course and sideslip. (**J**) Representation of yaw command.

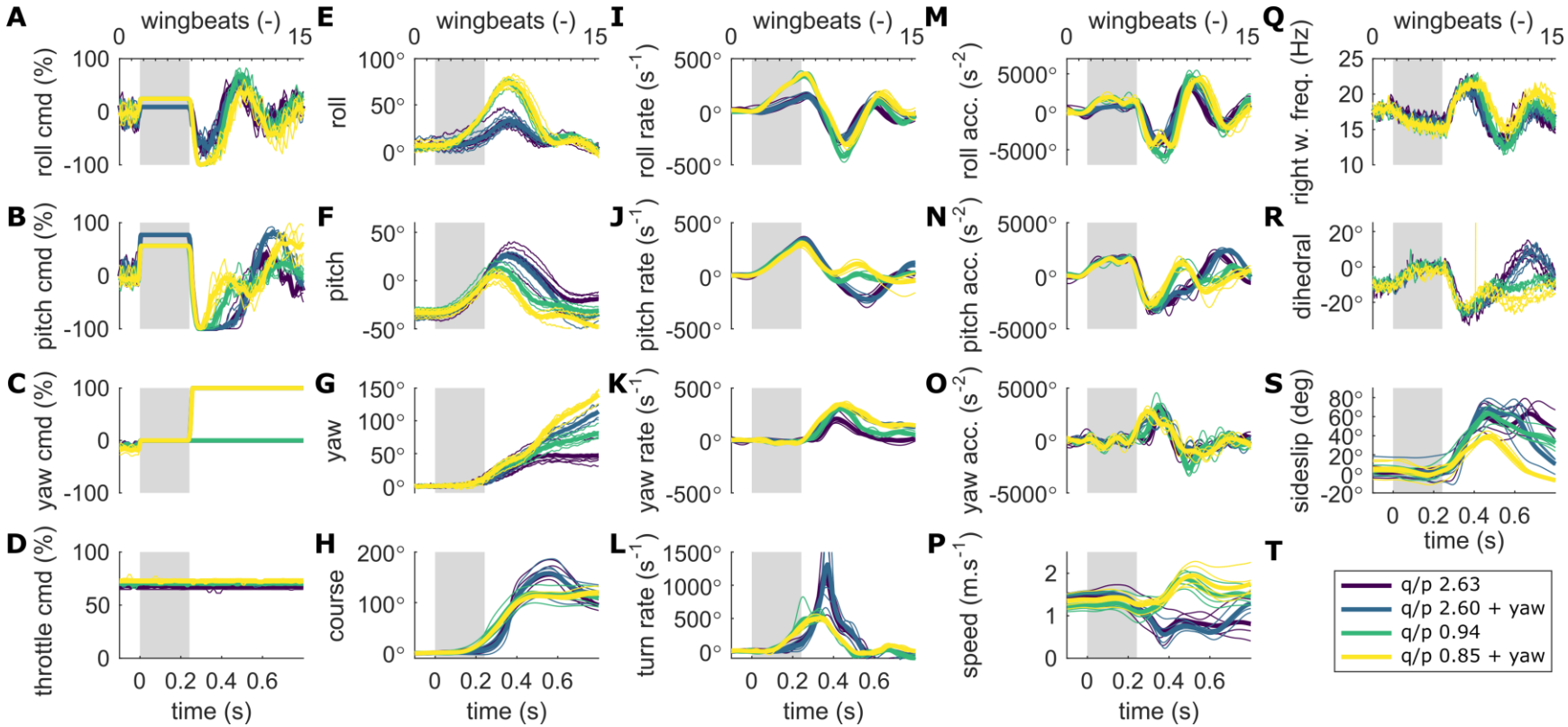

Fig. S26.

Flight data for the maneuvers in Fig S25. (**A–D**) The actuator commands: (**A**) roll actuator command, (**B**) pitch actuator command, (**C**) yaw actuator command, (**D**) throttle command. (**E–G**) Body attitude: (**E**) roll, (**F**) pitch, (**G**) yaw. (**H**) Course. (**I–K**) Body angular rates: (**I**) roll rate, (**J**) pitch rate, (**K**) yaw rate. (**L**) Turn rate. (**M–O**) Body angular accelerations: (**M**) roll acceleration, (**N**) pitch acceleration, (**O**) yaw acceleration. (**P**) Body velocity. (**Q**) Flapping frequency of the right wing-pair. (**R**) Dihedral angle. (**S**) Sideslip angle. The data is color coded according to the legend in panel (**T**), thin lines are individual trials, thick lines the averages. The gray background in all the panels highlights the OL phase of the maneuver.

Supplementary Tables

Table S1.

List of robot components and their weight.

Component	Number	Unit Weight
Lisa/S autopilot	1	2.12 g
DelTang Rx31 receiver	1	0.27 g
MX-3A ESC	2	0.38 g
HK-5330 servo actuator (removed casing)	2	1.63 g
microSD card 2GB	1	0.24 g
Hyperion CX G^3 25C 180 mAH LiPo battery	1	4.70 g
BL DC motor	2	1.43 g
Flapping mechanism	2	2.38 g
Wing-pair (including leading and root edges)	2	1.19 g
Control mechanisms (total)	1	1.35 g
Robot structure (total)	1	2.50 g
Wiring, glue & other (total)	1	3.07 g
Tracking markers + structure (total)	1	1.61 g
TOTAL WEIGHT (no payload)	**28.24 g**	
TOTAL WEIGHT (with tracking markers)	**29.85 g**	

Table S2.

The mass and inertial properties of the robot. The values in front of the brackets represent the inertial properties for the nominal dihedral angle (Γ_0 = 0°), the two values inside the brackets represent the Γ_{min} = -17° and Γ_{max} = +17° cases, respectively. The CoM position [x_{CoM}, y_{CoM}, z_{CoM}] is given with respect to the intersection of the leading-edge plane and the yaw axis in Fig. 1H.

Parameter	Without tracking markers Γ_0 (Γ_{min}, Γ_{max})	With tracking markers Γ_0 (Γ_{min}, Γ_{max})
Weight (g)	28.24	29.85
I_{xx} (kg.m^2)	9.20 e-5 (9.07 e-5, 8.93 e-5)	1.02 e-4 (1.01 e-4, 9.96 e-5)
I_{yy} (kg.m^2)	6.73 e-5 (6.83 e-5, 6.89 e-5)	7.87 e-5 (7.95 e-5, 8.05 e-5)
I_{zz} (kg.m^2)	3.18 e-5 (3.13 e-5, 3.02 e-5)	3.54 e-5 (3.48 e-5, 3.38 e-5)
I_{xy} (kg.m^2)	2.11 e-10 (-1.44 e-9, 1.75 e-9)	-2.78 e-7 (-2.60 e-7, 2.94 e-7)
I_{yz} (kg.m^2)	1.50 e-8	-3.97 e-7
I_{zx} (kg.m^2)	-9.92 e-6 (-5.93 e-6, 1.35 e-5)	-9.15 e-6 (-5.67 e-6, -1.22 e-5)
x_{CoM} (mm)	0.56 (-3.84, 4.66)	-0.03 (-4.17, 3.82)
y_{CoM} (mm)	-0.01	0.14
z_{CoM} (mm)	55.0	51.2

Table S3.

Parameters of the control loops in Fig. S3.

Parameter	Value
OM (roll & pitch)	800º s^{-1}
Zeta (roll & pitch)	0.85
ω_{max} (roll & pitch)	400º s^{-1}
α_{max} (roll & pitch)	8000º s^{-1}
LP filter cutoff (roll & pitch)	15 Hz
P_{roll}	0.625 rad^{-1}
D_{roll}	0.160 $s.rad^{-1}$
P_{pitch}	1.625 rad^{-1}
D_{pitch}	0.210 $s.rad^{-1}$
OM (yaw)	500º s^{-1}
Zeta (yaw)	0.85
ω_{max} (yaw)	90º s^{-1}
α_{max} (yaw)	900º s^{-1}
LP filter cutoff (yaw)	10 Hz
P_{yaw}	0.833 rad^{-1}
FF_{yaw}	0.011 $s.rad^{-1}$

Table S4.

Summary of all the experiments, including the tracking quality.

Exp. #	Figure(s)	Marker Quality *mean* *(standard deviation)*	Description	Camera setup
1	Fig. S13	2.2E-03 (3.2E-04)	Roll flips	10 x Prime 17W @ 360 fps
2	Fig. S14	2.4E-03 (5.7E-04)	Pitch flips	10 x Prime 17W @ 360 fps
4	Figs. S9, S10	1,0E-03 (4,5E-04)	Pitch & roll step	10 x Prime 17W @ 360 fps
13	Fig. 2E–I	8.7E-04 (2.8E-04)	Rapid banked turn (q/p = 0.54)	24 x Flex 13 @ 56 fps
36	Fig. 2C	1.3E-03 (6.9E-04)	Rapid banked turn (q/p = 1.67)	10 x Prime 17W @ 120 fps
54	Figs. 3, S20–S22	1.8E-03 (1.1E-03)*	Rapid banked turns (q/p = 1.34)	10 x Prime 17W @ 120 fps
55	Figs. 3, S20–S22	1.9E-03 (9.9E-04)	Rapid banked turns (q/p = 0.5)	10 x Prime 17W @ 120 fps
56	Figs. 2A, 3, S20–S22	2.0E-03 (1.3E-03)	Rapid banked turns (q/p = 0.35)	10 x Prime 17W @ 120 fps
57	Figs. 3, S20– S22	3.1E-03 (4.7E-03)	Rapid banked turns (q/p = 2.06)	10 x Prime 17W @ 120 fps
58	Figs. 3, S20– S22	1.9E-03 (2.3E-03)	Rapid banked turns (q/p = 0.84)	10 x Prime 17W @ 120 fps
76	Figs. S25, S26	7.6E-04 (3.3E-04)	Rapid banked turns (q/p = 2.63)	12 x Prime 17W @ 360 fps
77	Figs. S25, S26	7.9E-04 (3.5E-04)	Rapid banked turns with yaw (q/p = 2.6)	12 x Prime 17W @ 360 fps
78	Figs. S25, S26	8.2E-04 (5.2E-04)	Rapid banked turns (q/p = 0.94)	12 x Prime 17W @ 360 fps
79	Figs. S25, S26	1.6E-03 (3.6E-03)	Rapid banked turns with yaw (q/p = 0.84)	12 x Prime 17W @ 360 fps
94	Fig. S11	1.3E-03 (5.3E-04)	Repetition maneuvers (pitch steps 0° to 60°)	12 x Prime 17W @ 200 fps
95	Fig. S12	1.3E-03 (4.5E-04)	Repetition maneuvers (roll steps 0° to 15°)	12 x Prime 17W @ 200 fps
98	Fig. S12	1.4E-03 (6.3E-04)	Repetition maneuvers (roll steps 0° to 60°)	12 x Prime 17W @ 200 fps
99	Fig. S12	1.5E-03 (1.1E-03)	Repetition maneuvers (roll steps 0° to 45°)	12 x Prime 17W @ 200 fps
100	Fig. S12	1.3E-03 (4.3E-04)	Repetition maneuvers (roll step 0° to 30°)	12 x Prime 17W @ 200 fps
101	Fig. S11	1.3E-03 (7.8E-04)	Repetition maneuvers (pitch steps 0° to 30°)_	12 x Prime 17W @ 200 fps
102	Fig. S11	1.1E-03 (4.5E-04)	Repetition maneuvers (pitch steps 0° to 15°)	12 x Prime 17W @ 200 fps
103	Fig. S11	1.2E-03 (8.7E-04)	Repetition maneuvers (pitch steps 0° to 45°)	12 x Prime 17W @ 200 fps
104	Fig. S11	1.2E-03 (1.0E-03)	Repetition maneuvers (pitch steps 0° to 75°)	12 x Prime 17W @ 200 fps

Table S5.

Correlation coefficients and normalized root-mean-squared errors between the robot and the fruit fly scaled data in Fig. S17 and Fig. S18.

Variable	Maneuvers in Fig. 2A,B and Fig. S17		Maneuvers in Fig. 2C,D and Fig. S18	
	Pearson's correlation coefficient	Normalized root-mean-squared error	Pearson's correlation coefficient	Normalized root-mean-squared error
x position	0.99	0.07	0.92	0.13
y position	0.99	0.20	1.00	0.27
z position	0.01	0.85	-0.74	4.90
roll rate	0.97	0.30	0.77	0.26
pitch rate	0.83	0.25	0.89	0.20
yaw rate	0.97	0.11	0.58	0.29
u velocity	0.66	0.26	0.61	0.50
v velocity	0.99	0.21	0.95	0.57
w velocity	0.75	0.89	-0.10	0.48
course	0.99	0.06	0.98	0.16
sideslip	0.92	0.16	0.98	0.13
speed	1.00	0.30	-0.34	0.51
roll	0.95	0.24	0.94	0.13
pitch	0.88	0.87	0.75	0.73
yaw	0.90	0.50	0.98	0.27

Table S6.

Correlation coefficients between the measured yaw accelerations and the yaw accelerations predicted by the translation-induced coupled yaw torque for the data presented in Fig. S24.

Pitch-to-roll rate ratio q/p	Pearson's correlation coefficient c_r
0.35	0.75
0.50	0.89
0.84	0.91
1.34	0.90
2.06	0.21

Supplementary Movie Legends

Movie S1.

High-speed video recording (1000 fps) of the robot prototype in hovering flight, whereby the autopilot maintains stable flight of the inherently unstable flight platform. The movie shows the maneuver twice, first in real-time and then slowed down 33 times.

Movie S2.

High-speed video recording (1000 fps) capturing the robot prototype during the pitch maneuver described in the section 'Rapid transitions from hover to fast forward/sideways flight and back' (*18*). During the maneuver, the robot transitioned from hover to forward flight by rapidly pitching forward, and then pitched back again to return to a hover condition. The movie shows the maneuver twice, first in real-time and then slowed down 33 times.

Movie S3.

High-speed video recording (1000 fps) capturing the robot prototype during the roll maneuver described in the section 'Rapid transitions from hover to fast forward/sideways flight and back' (*18*). During the maneuver, the robot transitioned from hover to sideways flight by rapidly rolling sideways, and then roll back again to return to a hover condition. The movie shows the maneuver twice, first in real-time and then slowed down 33 times.

Movie S4.

High-speed video recording (240 fps) capturing the robot prototype rapidly accelerating from hover to fast forward flight, as part of the reproducibility test experiments (*18*). During the maneuver, the robot transitioned from hover to a set forward body pitch angle. The movie shows the maneuver three times: twice in real-time, and once slowed down 16 times.

Movie S5.

High-speed video recording (240 fps) capturing the robot prototype making 360° roll flip maneuver, as described in the section 'Rapid 360º roll and pitch up flips: barrel rolls and loopings' (*18*). The movie shows the maneuver three times, twice in real-time and once slowed down 16 times.

Movie S6.

High-speed video recording (240 fps) capturing the robot prototype making 360° pitch flip maneuver, as described in the section 'Rapid 360º roll and pitch up flips: barrel rolls and loopings' (*18*). The movie shows the maneuver three times, twice in real-time and once slowed down 16 times.

Movie S7.

High-speed video recording (240 fps) of the robot prototype making a fly inspired rapid banked turn, as described in the section 'Rapid banked turns inspired by the evasive maneuvers of fruit flies' (*18*). The movie shows the maneuver three times, twice in real-time and once slowed down 16 times.

Movie S8.

Animation of the comparison between the fly-inspired rapid banked turn (Fig. 2A, movie slowed down 16 times) and the original maneuver by the fruit fly (Fig. 2B, movie slowed down 150 times) (*3*). The difference in size between the robot and fruit fly is indicated by the scale bars in the video, representing the wing span of the robot and fly. The wing motion of the robot is for illustration purposes only; the flapping frequency was estimated using the on-board recorded commands.

Movie S9.

Animation of the comparison between the fly-inspired rapid banked turn (Fig. 2C, movie slowed down 16 times) and the original maneuver by the fruit fly (Fig. 2D, movie slowed down 150 times) (*3*). The difference in size between the robot and fruit fly is indicated by the scale bars in the video, representing the wing span of the robot and fly. The wing motion of the robot is for illustration purposes only; the flapping frequency was estimated using the on-board recorded commands.

Movie S10.

Animation of the fly-inspired rapid banked turn of Fig. 2E (q/p = 0.54), as described in the section 'Rapid banked turns inspired by the evasive maneuvers of fruit flies' (*18*). The animation was upsampled to 120 fps and replayed at 20 fps, and thus movie playback is slowed down 6 times. The three-dimensional trajectory is shown by the magenta dotted line, and the trajectory projections are shown as black dotted lines. Wings are color-coded with percentage motor command, see color bar. The flight phase, time t, speed U, and roll, pitch and yaw angle throughout the maneuver are provided in the top left of the movies. The wing motion of the robot is for illustration purposes only; the flapping frequency was estimated using the on-board recorded commands.